POSTER: Traffic Splitting to Counter Website Fingerprinting

Wladimir De la Cadena
University of Luxembourg
wladimir.delacadena@uni.lu

Asya Mitseva
University of Luxembourg
asya.mitseva@uni.lu

Jan Pennekamp
RWTH Aachen University
jan.pk@comsys.rwth-aachen.de

Jens Hiller
RWTH Aachen University
hiller@comsys.rwth-aachen.de

Fabian Lanze
Huf Secure Mobile GmbH
fabian@lanze.net

Thomas Engel
University of Luxembourg
thomas.engel@uni.lu

Klaus Wehrle
RWTH Aachen University
wehrle@comsys.rwth-aachen.de

Andriy Panchenko
BTU Cottbus
andriy.panchenko@b-tu.de

ABSTRACT

Website fingerprinting (WFP) is a special type of traffic analysis, which aims to infer the websites visited by a user. Recent studies have shown that WFP targeting Tor users is notably more effective than previously expected. Concurrently, state-of-the-art defenses have been proven to be less effective. In response, we present a novel WFP defense that splits traffic over multiple entry nodes to limit the data a single malicious entry can use. Here, we explore several traffic-splitting strategies to distribute user traffic. We establish that our *weighted random* strategy dramatically reduces the accuracy from nearly 95% to less than 35% for *four* state-of-the-art WFP attacks without adding any artificial delays or dummy traffic.

1 INTRODUCTION

In the age of mass surveillance, users rely on different anonymization techniques to ensure freedom of speech and to reduce their overall tracking on the Internet and in the IoT [9]. The Tor network [1]—currently the most popular low-latency anonymization network—promises to hide the identities (i.e., IP addresses) of users while communicating on the Internet. To accomplish this goal, user traffic is encrypted in multiple layers and encapsulated in fixed-size packets, called *cells*. These cells are transmitted through a virtual tunnel, i.e., *circuit*, over three nodes, called *onion relays* (ORs). The ORs are known as *entry*, *middle*, and *exit* depending on their position on the path to the destination. Primarily, Tor promises to offer user anonymity in the presence of a local passive adversary, e.g., a malicious entry OR. However, Tor is not able to conceal the size, direction, and timing of transmitted cells. An adversary can passively exploit this side-channel leakage and apply *website fingerprinting*—a special type of traffic analysis attack—to identify the content (i.e., the visited website) of anonymous user connections without breaking the encryption [7, 12, 15, 16].

Website Fingerprinting (WFP) usually corresponds to a supervised machine learning (ML) problem, in which the adversary first defines a set of websites he wants to detect and collects traces of multiple page loads for each of them. Next, he extracts patterns, i.e., *fingerprints*, for each website and applies ML to train a *classifier* that differentiates them. Finally, the adversary uses the classifier to identify the visited website corresponding to an unknown trace of a real user. Recent studies have concluded that WFP is more successful than previously expected [7, 15, 16] and available countermeasures (cf. Section 2) are less effective than previously assumed [15].

Our Contributions. We propose to limit WFP attacks with a novel user-controlled traffic-splitting countermeasure, which distributes traffic between the user and the middle OR over multiple entry ORs to limit the information available to an attacker who controls a subset of all used entry ORs. This design requires users to select an *effective traffic-splitting strategy*. In this paper, we explore several traffic-splitting strategies, which can serve as candidates for adoption in our multipathing architecture [14]. In particular, we analyze the efficiency of these strategies against modern WFP attacks by conducting a simulative evaluation with four state-of-the-art fingerprinting classifiers.

2 RELATED WORK

Research has presented countermeasures to protect users against WFP attacks. To remove website-specific patterns, several defenses add dummy traffic to generate a continuous data flow: BuFLO [5] sends multiple cells in bursts—with fixed size and time between bursts—at the cost of high bandwidth and latency overhead. To reduce this overhead, CS-BuFLO [2] and Tamaraw [3] cluster websites of similar size in a group and pad the number of transfered bytes for each page to the maximum in the corresponding group. However, these defenses introduce relatively high overhead in bandwidth and time and are not applicable in practice [15]. WTF-PAD [10] probabilistically fills gaps in the traffic with dummy packets to hide website-specific bursts. Walkie Talkie (WT) [17] modifies the browser to communicate in a half-duplex mode. Instead of sending packets at arbitrary times, it buffers and pads traffic in one direction to transmit it in bursts. WTF-PAD and WT have been considered for adoption into Tor due to their relatively moderate overhead [6, 11], but recent work proved them to be less effective than previously assumed [15]. Hence, no suitable candidate for adoption exists.

Like our work, Henri [8] proposes splitting traffic exchanged between the user and the entry OR over *two* different, unrelated

CCS '19, November 11–15, 2019, London, United Kingdom
© 2019 Copyright held by the owner/author(s).
ACM ISBN 978-1-4503-6747-9/19/11.
https://doi.org/10.1145/3319535.3363249

network connections (e.g., using several ISPs via DSL, Wi-Fi, or satellite or cellular networks). Thus, it fails to provide any protection against malicious ORs. Although Henri evaluates different splitting strategies for traffic distribution, he does not analyze the influence of the number of used network connections on the accuracy of WFP attacks—one of the main contributions in our work (cf. Section 3). Moreover, the author does not investigate the efficiency of his splitting strategies against the most robust WFP attack [15].

3 OUR TRAFFIC SPLITTING DEFENSE

The main goal of our defense is to provide an efficient traffic splitting strategy against WFP attacks. To find such a strategy, we analyze the influence of (i) the *number of distinct entry ORs* used to establish multiple paths between the user and the middle OR, and (ii) the *percentage* and *diversity* of traffic observable at each entry OR. We particularly aim for a splitting strategy that does not produce (repeatable) patterns, which again could be exploited by an attacker.

Number of Used Entry ORs. To apply our defense, we first need to determine the number m of entry ORs utilized by the Tor user for a given page load. While a large m decreases the amount of information available to each entry OR, it also increases the likelihood of selecting a malicious entry OR for one circuit [4, 14]. Thus, we analyze how m influences the user's protection against WFP attacks and propose a trade-off between the circuit establishment overhead and the probability of picking a malicious entry OR.

Distributing Traffic over Circuits. Having selected the number of entry ORs, we need to define how to distribute the traffic. In this paper, we focus on four different strategies for traffic distribution and their effectiveness against WFP attacks: (i) *Round robin*—our most basic strategy—shifts to the next circuit for each Tor cell (cf. Figure 1), while (ii) *random* splitting randomly selects a circuit for each Tor cell. We compare these to (iii) traffic splitting *by direction*, i.e., using one circuit for incoming and another circuit for outgoing traffic. Finally, (iv) to increase the diversity of the traffic distribution for repeated page loads of the same website, we evaluate a *weighted random* scheme. Specifically, for each page load, we create a vector $\overrightarrow{w}$ consisting of m probabilities, which, in turn, are computed from a m-dimensional Dirichlet distribution. We use these probabilities to weight the selection of an entry OR for each cell transmitted between the user and the middle OR.

4 EXPERIMENTAL SETUP

To allow for verifiable results, we next present our evaluation setup.

Dataset. As in previous related work [15], we rely on a dataset consisting of the index pages of the 100 most popular websites[1]. For fetching, we relied on an existing approach [12] that operates Tor Browser 7.5.6 to collect 100 traces for each website. We further reconstructed the corresponding Tor cells exchanged for each page load by applying a previously-used data extraction method [12]. For our evaluation, we focus only on Tor cells, since the different layers for data extraction (e.g., TCP packets, TLS records, or cells) only have a marginal influence on the classification results [12]. Hence, our results are comparable for other extraction formats.

Simulation of Traffic Splitting. We developed a simulator that artificially splits each page load trace from our dataset based on the selected strategy. We refer to the order of all cells that are assigned to the same circuit after splitting as a *subtrace*. In total, the simulator generates m subtraces for each page load. To obtain realistic results, our simulator further takes the latency of the different circuits between the user and the middle OR into account when employing our multipathing approach [14]. We measured the round-trip time (RTT) of several circuits consisting of the same middle and exit ORs but different entry ORs in the real Tor network. As in previous work [13], we measured the RTTs by sending a *relay connect* cell to a dummy destination, e.g., *localhost*, through each established circuit. As exit ORs forbid packets to *localhost*, these communication attempts trigger error messages that are sent to the measuring user. In total, we gathered RTTs for 4,073 successfully established circuits, which we integrated into our simulator.

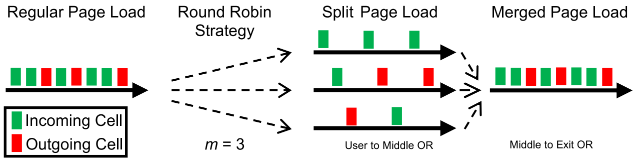

Figure 1: Splitting distributes traffic over multiple circuits.

Evaluation Setup. For our evaluation, we considered four state-of-the-art WFP attacks: k-NN [16], CUMUL [12], k-FP [7], and DF [15]. For details, we refer the reader to the original papers. For all experiments, we conducted a 10-fold cross-validation and calculated the *accuracy* of all attacks against our defense in a *closed-world* scenario (i.e., the attacker knows the set of all visited websites).

5 EVALUATION

Next, we present our results proving the effectiveness of our splitting strategies against the state-of-the-art WFP attacks. For all experiments, we assume that the adversary is aware of the applied splitting strategy and, thus, trains his classifier on traces which have been generated with the same strategy. In Table 1, we detail the accuracy of each classifier in a scenario without defense and against our evaluated strategies for varying numbers of entry ORs.

Number of Used Entry ORs. Before we discuss the efficiency of each traffic-splitting strategy in detail, we first analyze how the number of the used entry ORs influences accuracy of WFP attacks. Independent of the chosen strategy, we observe that all WFP attacks become less effective when the user utilizes a larger constant number of entry ORs to fetch a website. Our experiments confirm our initial intuition that a partial traffic pattern at a single entry OR is not sufficient to mount a successful WFP attack. We further notice no significant decrease of the classification accuracy for $m \geq 4$ regardless of the splitting strategy. Therefore, we consider $m = 4$ as a good choice and believe this choice neither significantly increases circuit establishment times (current versions of Tor already build three circuits preemptively [1]) nor dramatically increases the probability of selecting a malicious entry OR [4]. Moreover, a variable number of entry ORs for different page loads further reduces the classification accuracy (columns "$[\![2, 5]\!]$"). Here, a single malicious entry OR is challenged by the uncertainty of the applied splitting strategy as website-specific patterns are less deterministic.

Efficiency of Different Distributions. To find the most suitable splitting method, we next survey the efficiency of each strategy.

[1] According to Alexa (https://www.alexa.com/) as reported in September 2018.

Table 1: Accuracy (in %) of state-of-the-art WFP attacks in scenarios without defense and against our splitting strategies.

Strategy	Undefended	Round Robin					Random					By Direction		Weighted Random				
m	1	2	3	4	5	$[\![2, 5]\!]$	2	3	4	5	$[\![2, 5]\!]$	In	Out	2	3	4	5	$[\![2, 5]\!]$
k-NN	94.92	82.49	75.78	72.58	68.82	54.54	76.11	67.29	61.24	54.23	39.10	29.41	26.59	4.83	3.73	3.69	3.37	**3.31**
CUMUL	94.94	92.06	90.27	87.64	85.61	73.53	89.11	84.93	80.63	76.85	63.59	31.23	25.61	51.59	41.55	34.83	**31.13**	35.77
k-FP	92.09	88.45	86.46	83.87	81.94	69.26	85.59	81.28	78.21	75.57	69.26	60.11	51.55	45.65	38.31	35.66	**34.09**	34.23
DF	94.50	94.38	94.41	92.41	90.56	80.48	91.75	90.02	90.41	82.36	71.07	18.00	25.25	46.09	39.25	34.66	**30.27**	34.41

Round Robin & Random. Overall, we notice a slow decrease in accuracy with the round robin strategy as the number of used entry ORs increases. We observe a similar trend with a steeper decline for the random strategy. Nonetheless, the accuracies of CUMUL and DF still remain comparably high, corresponding to the correct identification of most page loads. A reason is that both strategies produce subtraces of similar size for different page load traces belonging to the same website when applied in a setting with a constant m. Moreover, round robin cannot completely hide the total size of a given website—one of the most important features for WFP attacks [7]—even when observing only a fraction of the page load. Then again, both round robin and random strategies introduce traffic diversity in a setting with a variable number of entry ORs, resulting in accuracy drops of more than 10%. Unfortunately, this drop is insufficient for a practical deployment.

By Direction. A simple scheme, which only splits the traffic by direction, already delivers a significant decrease in accuracy for all WFP attacks. Even though the number of transferred cells per direction remains unaltered, most classifiers only recognize a third of the page loads. This drop might be caused by the classifiers' inability to retrieve information about the relationship between incoming and outgoing cells. However, despite the absence of this characteristic, k-FP profits from other features that use available information on timing and data rate per direction, contributing to a comparably high classification rate for this attack.

Weighted Random. Finally, when applying a weighted random circuit selection, we observe a significant decrease in the accuracy compared to the other strategies. All evaluated WFP attacks achieve less than 35% accuracy. For the worst-performing classifier, k-NN, the rate of reliably-detectable page loads drops below 4%. We believe that this significant decrease is caused by the diversity in total size among the different subtraces of a single website. A notable observation is that a variable number of entry ORs (column "$[\![2, 5]\!]$") does not improve the defense, since the strategy already introduces sufficient diversity by design.

We conclude that a splitting strategy should generate subtraces with highly diverse characteristics to serve as an effective defense.

Bandwidth and Latency Overhead. Since our defense operates without any dummy traffic, we assume that it will not introduce significant overheads in bandwidth and time. However, our multipathing method [14] requires the integration of new types of Tor cells for circuit establishment and operation (sharing the chosen splitting strategy with the middle OR). Our defense may also require some extra time to build m circuits as well as to buffer and sort out-of-order cells. Based on previous analysis of multipath approaches [4], we believe that these additions are acceptable, especially since they support a promising defense.

6 CONCLUSION

We analyzed the effectiveness of different splitting mechanisms to counter WFP and concluded that a simple distribution of traffic across several entry ORs is insufficient to reduce the accuracy of state-of-the-art classifiers. Our evaluation reveals that a weighted random strategy decreases the accuracy of CUMUL, k-FP, and DF to less than 35% (and k-NN below 4%) in a closed-world scenario.

For future work, we plan to implement our defense, along with traffic splitting strategies, into Tor and evaluate its performance in real-world settings. We also want to extend our experiments to an open-world setting and consider the integration of traffic padding to reduce the fingerprinting accuracies even further without introducing noticeable overhead. Finally, we need to evaluate our defense against adversaries who control multiple entry ORs.

ACKNOWLEDGMENTS

We would like to thank Daniel Forster for his input in fruitful discussions during his internship at the University of Luxembourg. Parts of this work have been funded by the Luxembourg National Research Fund (FNR) within the CORE Junior Track project PETIT.

REFERENCES

[1] 2019. Tor Protocol Specification. https://gitweb.torproject.org/torspec.git/tree/tor-spec.txt. (Accessed: August 2019).

[2] Xiang Cai et al. 2014. CS-BuFLO: A Congestion Sensitive Website Fingerprinting Defense. In *WPES*.

[3] Xiang Cai et al. 2014. A Systematic Approach to Developing and Evaluating Website Fingerprinting Defenses. In *ACM CCS*.

[4] Wladimir De la Cadena et al. 2019. Analysis of Multi-path Onion Routing-Based Anonymization Networks. In *IFIP WG 11.3 DBSec*.

[5] Kevin Dyer et al. 2012. Peek-a-Boo, I Still See You: Why Efficient Traffic Analysis Countermeasures Fail. In *IEEE S&P*.

[6] Ian Goldberg. 2019. Network-Based Website Fingerprinting. https://tools.ietf.org/html/draft-wood-privsec-wfattacks-00. (Accessed: August 2019).

[7] Jamie Hayes and George Danezis. 2016. k-fingerprinting: A Robust Scalable Website Fingerprinting Technique. In *USENIX Security*.

[8] Sébastien Christophe Henri. 2018. Improving Throughput, Latency and Privacy with Hybrid Networks and Multipath Routing. PhD Thesis.

[9] Jens Hiller et al. 2019. Tailoring Onion Routing to the Internet of Things: Security and Privacy in Untrusted Environments. In *IEEE ICNP*.

[10] Marc Juarez et al. 2016. Toward an Efficient Website Fingerprinting Defense. In *ESORICS*.

[11] Nick Mathewson. 2019. New Release: Tor 0.4.0.5. https://blog.torproject.org/new-release-tor-0405. (Accessed: August 2019).

[12] Andriy Panchenko et al. 2016. Website Fingerprinting at Internet Scale. In *NDSS*.

[13] Andriy Panchenko and Johannes Renner. 2009. Path Selection Metrics for Performance-Improved Onion Routing. In *IEEE SAINT*.

[14] Jan Pennekamp et al. 2019. Multipathing Traffic to Reduce Entry Node Exposure in Onion Routing. In *IEEE ICNP*.

[15] Payap Sirinam et al. 2018. Deep Fingerprinting: Undermining Website Fingerprinting Defenses with Deep Learning. In *ACM CCS*.

[16] Tao Wang et al. 2014. Effective Attacks and Provable Defenses for Website Fingerprinting. In *USENIX Security*.

[17] Tao Wang and Ian Goldberg. 2017. Walkie-Talkie: An Efficient Defense Against Passive Website Fingerprinting Attacks. In *USENIX Security*.