Connecting the Dots in the Sky: Website Fingerprinting in Low Earth Orbit Satellite Internet

Prabhjot Singh
University of Waterloo
prabhjot.singh@uwaterloo.ca

Diogo Barradas
University of Waterloo
diogo.barradas@uwaterloo.ca

Tariq Elahi
University of Edinburgh
t.elahi@ed.ac.uk

Noura Limam
University of Waterloo
noura.limam@uwaterloo.ca

***Abstract*—Despite the implementation of encrypted channels, such as those offered by anonymity networks like Tor, network adversaries have demonstrated the ability to compromise users' browsing privacy through website fingerprinting attacks. This paper studies the susceptibility of Tor users to website fingerprinting when data is exchanged over low Earth orbit (LEO) satellite Internet links. Specifically, we design an experimental testbed that incorporates a Starlink satellite Internet connection, allowing us to collect a dataset for evaluating the success of website fingerprinting attacks in satellite environments compared to conventional fiber connections. Our findings suggest that Tor traffic transmitted via Starlink is as vulnerable to fingerprinting attacks as traffic over fiber links, despite the distinct networking characteristics of Starlink connections in contrast to fiber.**

I. Introduction

To ensure that Internet users can communicate securely in the face of network adversaries with the capabilities to intercept their communications, encryption protocols such as TLS [31] were devised to prevent adversaries from eavesdropping or manipulating exchanged messages. However, while encryption obscures the content of communications, network adversaries may still discern privacy-sensitive information about users (e.g., insights into a user's health status or financial situation [42]), by simply tracking the sequence of websites a user visits over time. The main reason why these attacks are possible is because widespread encryption protocols such as TLS are unable to hide communication metadata, such as the source and destination IPs of a given data exchange or the times at which these data exchanges take place.

To shield themselves from the above risks, savvy Internet users typically resort to privacy-enhancing technologies, such as the Tor anonymity network [8], to conceal the identity of the websites they access through the Internet. Specifically, Tor makes use of a technique known as onion routing to shroud the destination IP address of a user's communication by routing the user's traffic through multiple Internet nodes (or relays, usually three) that comprise a Tor circuit.

Even though Tor provides an enhanced level of privacy to its users, studies on *website fingerprinting* [11], [28] have shown that network eavesdroppers can still overcome Tor's safeguards. Put briefly, an attacker can build a database of website fingerprints, i.e., a set of signatures drawn from the characteristics of the traffic observed when accessing a given website over Tor, and then attempt to match the traffic patterns generated by a Tor user with a fingerprint in this database.

Despite the risks posed by website fingerprinting attacks, their accuracy is known to be sensitive to the underlying conditions of the network segments under analysis [7], [14] (such as the available bandwidth, jitter, or packet drop rates), since these conditions can lead to modifications on the overall shape of the traffic patterns generated when accessing websites [13]. Thus, in the past, researchers have wondered whether (and to what extent) the risks of website fingerprinting attacks would transfer from traditional fiber connections to networking mediums with different transmission characteristics, such as wireless LTE/4G networks [17], [34]. Interestingly, these studies have shown that attackers were still able to accurately fingerprint users' Tor traffic in such settings.

Today, we observe an increasing prevalence of satellite Internet solutions, powered by the launch of LEO (low Earth orbit) satellite constellations such as Starlink [39] and OneWeb [22]. These solutions have largely facilitated the provisioning of Internet access to users residing in remote regions, and continue to be enhanced through the launch of more capable satellites and upgrades to routing algorithms within the constellations themselves [2], [52]. While promising connectivity speeds similar to fiber networks, LEO satellite Internet makes use of wireless mediums which are known to be prone to several sources of interference [16], [20], [29], [48]. It remains unclear, however, what implications these recent satellite networking environments may have on the privacy of users, especially when considering adversaries with the ability to eavesdrop and analyze the metadata of Internet connections that are partly or entirely established via satellite links [26].

In this paper, we aim to shed light on whether LEO satellite Internet users are more vulnerable to website fingerprinting attacks than users using traditional fiber. To this end, we set up an experimental testbed including both a fiber and Starlink connection, and use them to collect a dataset of synchronized website accesses over Tor. We leverage state-of-the-art website fingerprinting attacks over our collected traces to understand whether network adversaries able to inspect the ground links between users and the first hop of both kinds of connections (like a snooping satellite ISP) can identify which websites are being accessed by users. Lastly, we evaluate the security benefits and performance trade-offs of website fingerprinting defenses when applied to fiber and satellite Internet links.

Workshop on Security of Space and Satellite Systems (SpaceSec) 2024
1 March 2024, San Diego, CA, USA
ISBN 979-8-9894372-1-4
https://dx.doi.org/10.14722/spacesec.2024.23xxx
www.ndss-symposium.org
Pre-print version (manuscript accepted for publication at the workshop)

Our findings suggest that Tor traffic exchanged over Starlink Internet links is equally vulnerable to website fingerprinting attacks as Tor traffic exchanged over traditional fiber links. We hypothesize that, despite the different connectivity characteristics of the ground-satellite link that connects our measurement node to the Tor network, most of the interference experienced in this link is absorbed by the network effects (e.g., added latency, jitter, etc.) inherent to Tor circuits.

Contributions. We deliver the following main contributions:

- We implement a testbed that includes a Starlink satellite dish, and use it to collect a novel dataset of Tor traffic over LEO satellite links. We open-source this dataset and our data collection code to foster further research [37].
- We perform a comparative study over the traffic characteristics observed in connections established with and without Tor, both via Starlink and traditional fiber links.
- We explore the success of state-of-the-art website fingerprinting attacks over satellite links, and analyze the suitability of existing website fingerprinting defenses to be deployed on LEO satellite-based Internet links.

II. Related Work

Website fingerprinting. A website fingerprinting attack links a user to the websites they visit, thus defeating the privacy property Tor aims to provide. A website fingerprinting attack conducted over Tor usually requires the adversary to be located between the user and the entry node of the Tor circuit. The adversary has the ability to eavesdrop on communications but not modify, delete, or add packets. In preparation for an attack, the adversary repeatedly accesses a pool of websites it wishes to monitor, collecting the network traces generated upon each of these accesses. Then, the adversary extracts a set of attributes that characterize these traces (e.g., based on the timing, volume, and direction of traffic) to build a database of website *fingerprints*. Once this database is populated, the adversary employs machine learning techniques to create a model for predicting which website a given fingerprint corresponds to. Finally, to launch an attack, the adversary waits for the target user to access a website via the encrypted tunnel, extracts a fingerprint from the resulting traffic, and uses the trained model to identify which website the user has visited.

Attacks. The first wave of website fingerprinting attacks made extensive use of manually engineered features to train classical machine learning classifiers. Well-known instances include the *k*-NN [46], CUMUL [24], and *k*-fingerprinting [11] (*k*-FP) attacks. In turn, recent developments have adopted deep learning to actively automate the process of feature extraction from less pre-processed representations of traffic. The AWF [32] and DF [38] attacks used packets' directional information as input to deep neural networks, while *Tik-Tok* [28] and *Var-CNN* [1] extended these notions by incorporating packet timing information alongside directional data. RF [35] makes use of a traffic aggregation matrix which also uses packet direction and timing, but splits traces into fix-length time slots.

Defenses. Website fingerprinting defenses aim to prevent attacks by concealing the true shape of website traces. Some defenses, such as CS-BuFLO [4] and Tamaraw [5], obfuscate packet timing and burst characteristics by keeping a consistent rate of packet transmission. However, these defenses increase latency and bandwidth consumption, limiting their practicality. Conversely, adaptive and randomized padding defenses, exemplified by WTF-PAD [15] and FRONT [9], proactively introduce chaff to obfuscate the timing of the true packets belonging to a connection, homogenizing the access to various websites. While we use the aforementioned padding-centric defenses in this work, Mathews et al. [19] outline a more exhaustive examination of website fingerprinting defenses.

Website fingerprinting in wireless networks. Wireless networks, and mobile networks, in particular, have previously been analyzed in the context of website fingerprinting. Rupprecht et al. [17] provided an analysis of potential attacks that can be targeted towards LTE (Long-Term Evolution), a prevalent mobile communication standard aimed at enhancing data transmission rates and connectivity beyond the capabilities of preceding 3G networks. By placing their focus on LTE's data link layer, Rupprecht et al. were able to find protocol flaws that allow an eavesdropper to access a mobile's device communication metadata, thus allowing for successful website fingerprinting attacks on encrypted LTE traffic.

LEO satellite connections' characteristics. Emerging LEO constellations have the objective of offering broadband internet services with decreased latency. Satellite constellation networks are comprised of ground stations and orbiting satellites, enabling worldwide communication. LEO satellites are positioned at varying altitudes, typically ranging from around 180 km to 2 000 km above the Earth's surface. This close proximity provides the advantage of reduced communication delays and the potential for enhanced data throughput [44].

The recent and ongoing deployment of Starlink, a prominent LEO satellite constellation developed by SpaceX, has given birth to a flurry of measurement studies focused on analyzing the performance and operational effectiveness of Starlink's satellite Internet service [16], [18], [20], [29], [48], [52]. For instance, Ma et. al [18] show that the throughput and latency experienced by Starlink users are highly dynamic, especially when compared to conventional terrestrial networks. These results have also been corroborated by other studies [16] which show satellite Internet performance to vary by geography – clients located in the USA were found to experience 2.3x higher communication latency than those in the UK, together with lower network throughput – and also prone to unusually high packet loss rates. In addition, Starlink's performance is also influenced by environmental factors such as terrain, rain, clouds, and temperature [16], [52].

From the above, one can see that satellite communication links can be substantially more unstable than traditional fiber links. Our work aims to shed light on whether website fingerprinting attacks can succeed in such a setting.

III. Methodology

This section details the methodology of our study. In Section III-A, we describe our threat model and assumptions, while Section III-B presents our experimental testbed and details the process we followed for collecting and pre-processing websites' network traces. Then, Section III-C describes the website fingerprinting attacks and defenses we consider, and Section III-D describes the metrics we use for measuring the success of attacks and the performance of defenses.

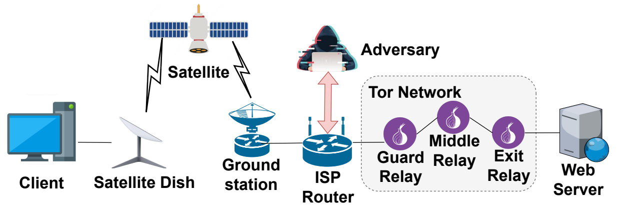


Figure 1: Threat model overview.

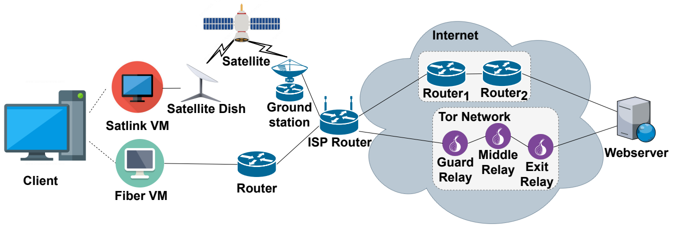


Figure 2: Data collection testbed.

A. Assumptions and threat model

We follow the typical threat model for website fingerprinting attacks, albeit with one important change to the location and mode of operation of the adversary. This change is motivated by the real-world challenges in acquiring traffic metadata from Starlink signals.

More concretely, Starlink satellites beam data using sophisticated signal encryption schemes that allow only the target satellite dish receiver to be able to decode the information being sent/received to/from the satellite [49]. In other words, even if the adversary places a Starlink satellite dish within the same geographical data transmission cell where the target user satellite dish sits in, the adversary would be unable to access the raw IP packet stream (or other well-formatted data comprising Starlink's data-link layer) that is directed at the target user. This prevents a typical website fingerprinting adversary from inspecting users' communications.

We note that in other cases, such as website fingerprinting attacks launched over LTE/4G networks [17], [34], the adversary is first required to tap into the radio signals exchanged between a user's equipment (e.g., a smartphone) and the LTE base station. This capability, which can be obtained through the use of LTE software stacks implemented in software-defined radios (e.g., srsRAN [23]) in tandem with sniffer analysis frameworks (e.g., OWL [3]), allows the adversary to access and decode transmissions ranging from the physical layer up to the data-link layer, and then derive user-specific traffic metadata. However, to the best of our knowledge, such capabilities are not publicly available for Starlink satellite links, despite current advances in the reverse-engineering of Starlink downlink signals [12], [21]. For this reason, we introduce a variation of the website fingerprinting adversary model.

An ISP-based website fingerprinting adversary. While strong signal encryption may prevent most third parties from inspecting the traffic of satellite Internet users, ISPs operating the satellite networking service itself might be interested in launching website fingerprinting attacks. In this setting, it is possible that, despite allowing users to leverage privacy-preserving communication protocols such as those provided by Tor, snooping ISPs might wish to identify which content is being accessed by their users, e.g., towards preventing the access to websites used for streaming pirated DRM-protected content [33]. For instance, according to Starlink's fair use policy [41], the company reserves the right to take additional network management measures as necessary to comply with applicable laws, including the analysis of traffic patterns.

Figure 1 illustrates this scenario, placing the eavesdropping adversary at the satellite provider's infrastructure, with full access to the IP traffic exchanged by the satellite Internet users and their destinations.

Attack setting and other assumptions. Our study focuses on website fingerprinting attacks within the *closed-world* scenario, where the target user is assumed to visit a website from a predetermined set of websites monitored by the adversary. We assume that access to any monitored website in our closed-world setting is equally probable. We also assume that the attacker can separate the traces associated with the loading of different websites and determine which defense is in use by a Tor user. Thus, we consider the use of defenses like GLUE [9] to be outside the scope of our study.

B. Dataset collection and pre-processing

We collect a novel dataset of website accesses via Tor that contain two different sets of traces: those collected when the client uses a simple terrestrial fiber network, and those collected when the client uses a satellite link. Having both of these sets allows us to build a baseline of the effectiveness of website fingerprinting attacks on a typical Internet connection and further allows for direct comparisons with the effectiveness of the same attacks once deployed over Starlink traces.

Apart from Tor traces, we also collect additional website traces over direct connections to each website using plain Firefox. We collect these traces to characterize the overheads of using Tor instead of Firefox on both satellite and fiber connections (see Section IV). Later in our evaluation, we also use this data to compare the performance of a classifier when fingerprinting plain traffic vs. Tor traffic.

Experimental testbed. Figure 2 depicts a birds-eye overview of our experimental testbed. It essentially comprises a client machine under our control, which is used to access a set of websites included in our closed-world website list via the Tor network. The client machine executes two virtual machines (VMs), each with 16GB of storage and 2GB of RAM. It is equipped with two network interface cards, and each VM routes its traffic through a different interface. One of these cards is connected to a Starlink dish (satellite Internet connection), while the other is attached to our university's fiber-based network (terrestrial Internet connection). On each VM, we deploy Docker containers that we orchestrate for simultaneously collecting network traces of a given website. Website access is only deemed successful if we correctly retrieve a website's homepage both via fiber and Starlink.

The ability to collect traces for the same website simultaneously over the two different links enables us to collect traces that represent a given website at roughly the same instant of time. This way, we mitigate the effect of concept or data drift [14] by minimizing the chance that our fingerprint database would include significantly different versions of a given website, should, for instance, all fiber traces be collected after all Starlink traces.

Considered websites. In our data collection procedure, we considered the top 125 websites found on the Tranco list [27], as of September 2022 [43]. We manually verified that each of these websites was active by sending a request to the website's homepage and confirming it returned an HTTP 200 response code. We configured our scripts to collect a total of 125 instances of each of the 125 websites. However, the number of websites (and per-website samples) included in our dataset was later trimmed to account for transmission errors detected upon data pre-processing (discussed later in this section).

Collection of website traces. As stated before, we collect traces using the fiber and Starlink connections simultaneously, interleaving Tor- and plain Firefox-based requests. We visit each website via Firefox by automating web browser interactions via `selenium`, and visit each website via Tor by leveraging `tbselenium`, a headless wrapper around the Tor browser. We used the default configuration setup for Tor and, to ensure the freshness of each website visit, we restarted the `tbselenium` Tor driver after clearing its cache upon each visit and forcing the selection of different circuits. If a given visit to a website returns an explicit error to `tbselenium` (e.g., due to network instabilities), we try revisiting the website (up to a maximum of three times) towards receiving a valid response.

Pre-processing of network traces. We only deem a given website access as *valid* if we can confirm the successful access to the website's homepage via Tor (on both interfaces) and Firefox (also on both interfaces). This effectively comprises a batch of four individual trace samples which we add to our dataset. In our pre-processing step, we aim to weed out from the dataset those traces that resulted in timeouts (we consider a request to timeout if one minute has elapsed before the page can be successfully retrieved) or that include errors that prevented the website from being fetched correctly (but that did not trigger explicit errors in `selenium` or `tbselenium`).

After removing traces afflicted by the above issues, we obtained a dataset that includes 80 instances each of 75 different websites (listed in Appendix A) visited over both Starlink and terrestrial fiber, using both Tor and Firefox. We denote this dataset as $TorFirefox\text{-}SatFiber_{4\times75\times80}$, containing a total of $4 \times 75 \times 80 = 24\,000$ samples, and publicly released it [37].

Interestingly, our pre-processing step revealed that we were unable to access specific websites via Tor entirely, despite the per-access refresh of `tbselenium` and Tor circuitry. In line with previous findings, we conjecture that these websites may actively block accesses coming from Tor [50], [51].

C. Attacks and defenses

Website fingerprinting attacks. Our study employs prominent attacks in the literature, including those that leverage manually-engineered features (*k*-FP), as well as those that leverage latent feature spaces learned through deep learning (DF and Tik-Tok). While *k*-FP provides human-interpretable information on which traffic features lead the classifier to issue predictions, the latter attacks have been shown to be more effective in practice.

The *k*-FP attack leverages 150 different summary statistics extracted from network traces. The classifier operates by generating a fingerprint for each website using a modified version of the Random Forest algorithm. Then, the attack uses the k-Nearest Neighbours classifier to predict website accesses. Instead, DF and Tik-Tok are based on deep neural networks that directly extract latent features from input traces passed to the classifier during the training and inference step. The DF attack accepts as input a direction vector representing the direction of packets in the trace, while the Tik-Tok model enriches this trace representation by computing the element-wise product of the direction and timing of packets in a trace. We leverage all attacks with their default hyperparameters.

Website fingerprinting defenses. Our study leverages a set of popular open-source implementations of website fingerprinting defenses, including: a) Tamaraw [5] and CS-BuFLO [4], two defense mechanisms that employ strategies based on fixed-rate packet transmissions to conceal timing patterns and packet burst behavior; b) FRONT [9], a defense which adds a variable number of randomly-padded dummy packets to the start of packet sequences, and; c) WTF-PAD [15], a lightweight adaptive padding defense that inserts dummy packets to conceal the existing time gaps between packets.

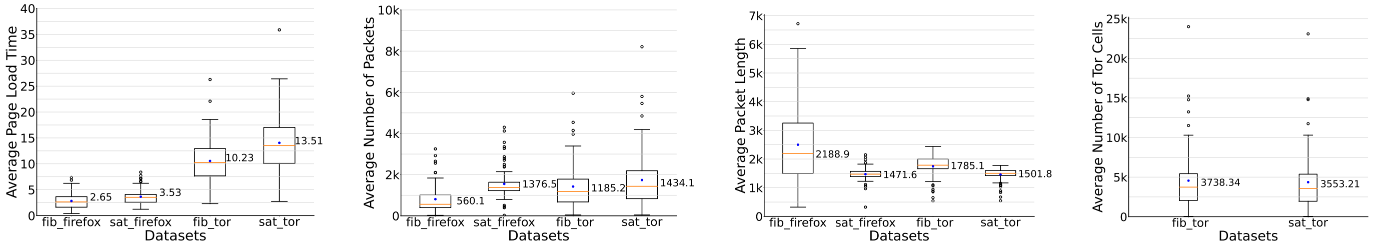
To avoid the repeated collection of traffic traces for evaluating website fingerprinting defenses, these defenses' authors released simulators that can turn undefended Tor traffic traces into their defended versions in an offline manner. Gong et al. [10] have recently compared the simulation and true implementation results for a set of WF defenses and reached the conclusion that simulators can accurately reflect the effectiveness of defenses on live traffic. We utilize the same defense simulators and configurations recently used in the work of Veicht et al. [45], which focused on the security analysis of website fingerprinting defenses. We refer the reader to Appendix B for details on the defenses' parameters.

D. Evaluation procedure and metrics

Evaluation procedure. We make use of 10-fold cross-validation when training and testing our classifiers to minimize the effects of selection bias. In particular, we employ stratified cross-validation to ensure an equal distribution of instances across all the classes comprising our dataset. In each cross-validation fold, we use 80% of the data for training, 10% for the model's validation, and the remaining 10% for testing.

Attack performance metrics. The main metric we pay attention to when analyzing the success of a website fingerprinting attack (whether a defense is being used or not) is *accuracy*. Accuracy is defined as the ratio of correctly predicted instances to the total number of instances in the dataset, and this metric has been extensively used in the website fingerprinting literature for determining the efficacy of both attacks and defenses in the closed-world scenario, carrying a rather intuitive meaning for an adversary – it quantifies the adversary's success in discerning exactly which website a given user is accessing.

Defense performance metrics. Apart from a desirable reduction in attacks' accuracy, website fingerprinting defenses may also be evaluated on the amount of overhead they impose over an undefended Tor network trace. For this reason, in our experiments, we also leverage *bandwidth* and *latency*



(a) Page load times (s).

(b) Number of IP packets.

(c) IP packet lengths.

(d) Number of estimated Tor cells.

Figure 3: Different network-based statistics extracted from the $TorFirefox\text{-}SatFiber_{4\times75\times80}$ dataset.

overheads as efficiency indicators of website fingerprinting defenses. Defenses are typically deemed to be practical if and only if they can substantially reduce an attack's accuracy while having a small impact on latency (i.e., the time to load a website) and bandwidth overhead (i.e., the amount of additional data required to load a website).

Traffic analysis machine. To train and test our models on the network traces we collected for our study, we leverage a server machine with 2 AMD EPYC 7302 16-Core CPUs, 512 GB RAM, and an NVIDIA A100 GPU with 40 GB memory.

IV. Characterizing Fiber and Starlink Traces

This section presents a characterization of the traces included in our dataset. We aim to uncover the major differences between connections established over terrestrial fiber and Starlink, as well as highlight the performance drops expected when using Tor instead of plain Firefox in these different networking environments. We describe our main takeaways below.

Starlink is 33% slower than fiber (on our deployment). Figure 3(a) depicts the page load times observed when loading the 75 websites included in our dataset over plain Firefox and Tor, both for Starlink and fiber connections. We can observe that Starlink-based connections consistently reveal higher times-to-last-byte when compared to fiber connections, representing a total average increase of 33.2% when considering website accesses established over Firefox, and 32% average increase when considering website accesses performed via Tor.

Tor is almost 4× slower than plain Firefox. Figure 3(a) shows that, for fiber connections, Tor accesses are on average 3.86× slower than those via plain Firefox. This difference is even more pronounced when considering Starlink connections, where Tor accesses are on average 3.83× slower than those via plain Firefox.

Starlink connections require more packet exchanges. Figure 3(b) summarizes the number of packets observed when accessing each of the 75 websites via plain Firefox and Tor, both for Starlink and fiber connections. We can see that the median number of packets exchanged when using Firefox more than doubles when using a Starlink connection (1376.47 packets) when compared to the use of fiber (560.11 packets). For Tor, we are also able to observe an increase in exchanged packets when moving from fiber to Starlink setting, but this increase seems less pronounced (∼21% more packets).

Interestingly, website accesses using Firefox via Starlink reveal a smaller inter-quartile range than accesses via a terrestrial fiber connection, indicating a more concentrated distribution around the mean. The opposite is true for accesses over Tor where, albeit less evident, the distribution seems to be more concentrated around the mean for the connections making use of the fiber connection.

Starlink-exchanged packets tend to be smaller. Figure 3(c) depicts the length of IP packets observed when accessing each of the 75 websites over each of our networking configurations. We can see from the figure that the packets composing Tor traffic exhibit a rather concentrated size, with a median size of 1501.79 when Tor data is exchanged via Starlink and a median size of 1 785.11 when exchanged via fiber connections. Interestingly, we observe that the size of plain Firefox packets is also rather concentrated around a mean of 1 471.64, while plain Firefox packets exchanged over fiber connections exhibit a more variable (and typically larger) length, with a median of 2 188.94 and a size of 3 254.31 at the 75th percentile.

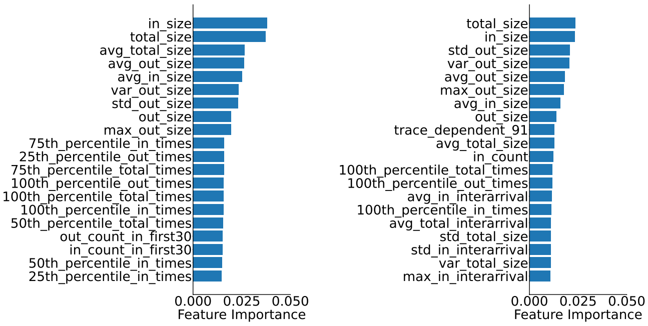
TCP retransmissions are more common in Starlink. Towards understanding the differences in the number of packets observed in our traces (Starlink vs. fiber), we conducted an additional analysis focused on the study of TCP retransmissions. In general, while retransmission requests are fairly rare throughout our traces, they are more common in Starlink connections. For instance, when accessing the website *google-domains.com* via plain Firefox, we observed that 0.02% of packets are retransmitted when using the fiber link, while 0.3% of packets are retransmitted when using Starlink. When accessing the same website using Tor, we find that 0.03% of packets are retransmitted over a fiber connection, while 0.7% of packets are retransmitted over Starlink. This shows that even if the percentage of retransmitted packets is not excessively high in any of the scenarios, there is a disparity of packet retransmissions (up to an order of magnitude) when using the Starlink connection instead of fiber. This alludes to the inherent noise previously found in Starlink internet connections [16].

A comparable number of Tor cells are exchanged over Starlink and fiber connections. In contrast to the average number of IP packets exchanged, the average number of Tor cells transmitted through both fiber and Starlink exhibit a remarkable similarity (see Figure 3(d)). In addition, one can observe that 75% of the traces exhibit a number of Tor cells that is less than or equal to 5 436. As most deep-learning website fingerprinting attacks trim their input vectors to 5 000 cells, we posit the same trimming threshold should also work well for our dataset (we validate this claim in Section V-B).

Summary. Overall, our findings suggest that the use of Starlink imposes a larger relative penalty on plain Firefox connections as compared to Tor connections. We hypothesize that the variable latency and jitter which is introduced (and

Table I: Attack accuracy for Firefox traces (on TCP/IP data).

Dataset	***k*-FP**	***k*-FP (w/ pkt. lengths)**	**DF**	**TikTok**
Firefox w/fiber	0.8557	0.8892	0.8837	0.7945
Firefox w/Starlink	0.4075	0.4285	0.4915	0.4715



(a) Via fiber.

(b) Via Starlink.

Figure 4: Top-20 most important features (Firefox traces).

compounded) by the multiple relays composing Tor circuits may help amortize the performance penalties incurred by clients that use Starlink's up/downlinks to connect to the Internet via Tor. Next, we discuss the findings resulting from a set of experiments with state-of-the-art website fingerprinting attacks and defenses conducted over our dataset.

V. Fingerprinting Fiber and Starlink traces

In this section, we compare the susceptibility of fiber and Starlink connections to different website fingerprinting attacks.

A. *Attacks based on classical machine learning*

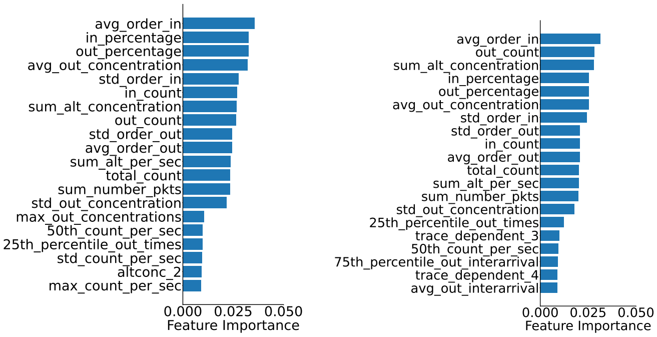
We start by providing the main takeaways of our experiments with *k*-FP on the different sets of traces composing our dataset. Besides assessing the success of this attack on Tor traffic, we also attempt to fingerprint plain Firefox connections. Note that, in practice, the destination of Firefox connections would be trivially disclosed to an adversary (e.g., by looking at the connection's destination IP address). However, we do this as an exercise towards understanding how satellite connections affect traffic features and whether these effects degrade or improve our ability to fingerprint network traffic.

To perform the above comparisons with plain Firefox traffic, we modify the original implementation of the *k*-FP attack in two meaningful ways. First, we allow for features to be directly generated from TCP/IP header information (e.g., IP packet length, time between IP packets, etc.) instead of Tor cells as in the original attack. Second, we create a version of the *k*-FP classifier which takes packet lengths into account as features for building and matching website fingerprints. The rationale for these modifications on *k*-FP hinges on the fact that Tor exchanges data in cells padded to 512B, thus making packet length analysis irrelevant [47]. In contrast, Firefox does not exchange data via cells, thus providing a network data analyst with access to raw TCP/IP packet length information.

***k*-FP is more accurate for plain Firefox traffic over fiber.** In this first experiment, we used raw TCP/IP packet header data to generate features for the *k*-FP attack. Table I depicts the accuracy of the website fingerprinting attacks we consider on Firefox, over both fiber and Starlink connections. We see that the original *k*-FP classifier achieves an accuracy of 85% when fingerprinting websites accessed via *Firefox w/fiber*, but achieves an accuracy of only 40% when fingerprinting websites accessed via *Firefox w/Starlink*. The inclusion of packet lengths in *k*-FP brings only marginal benefits for the attack in both settings, amounting to a ∼3% accuracy increase.

Table II: Attack accuracy using Tor cell data.

Dataset	***k*-FP**	**DF**	**TikTok**
Tor w/fiber	0.7282	0.8738	0.8860
Tor w/Starlink	0.6426	0.8540	0.8682



(a) Via fiber.

(b) Via Starlink.

Figure 5: Top-20 most important features (Tor traces).

Figure 4(a) and Figure 4(b) show the top-20 most important features for the *k*-FP attack when launched over plain Firefox traffic exchanged via fiber and Starlink, respectively. We can observe that the two most important features for classifying website accesses on Firefox via fiber are the sum of all incoming packet sizes and the sum of all packet sizes in the data exchange, whereas the importance of these features is swapped for Firefox via Starlink traffic. We can also see from Figure 4(a) that 9 out of the 20 most important features focus on packet timing information, whereas only 7 timing-related features are within the top 20 most important features for Firefox traffic exchanged over Starlink. Interestingly, while timing features in the former case are mostly related to percentiles, timing features are more related to the average and standard deviation of packet arrivals for the latter.

The above observations bear further work to ensure that our results are not specific to one site (we discuss a multisite study in Section VII), as well as further analysis into what characteristics of *Firefox w/Starlink* traffic can make it less fingerprintable (possibly borrowing some of these findings to develop a novel website fingerprinting defense).

***k*-FP is more accurate for Tor traffic over fiber.** In this second experiment, we analyzed the effectiveness of the original *k*-FP attack on Tor traffic exchanged via fiber and Starlink. Thus, in this case, we extract the attack features based on our estimates of the Tor cells exchanged within these traces. The results in Table II show that the accuracy of *k*-FP in *Tor w/fiber* traffic is close to 73% and around 64% for *Tor w/Starlink*. This discrepancy is consistent with the results observed for Firefox browsing, where the classifier had performed better for fingerprinting websites visited via the fiber connection.

A close look at the top 20 most important features for classifying Tor traffic (Figure 5) reveals that the cumulative average of incoming packets is the most important feature for classifying both kinds of connections. Moreover, 14 out of the top 20 features are shared between both (though not necessarily in the same order). This may be the case due to the similarity observed in Tor cell statistics for both fiber and Starlink traffic, as observed in Figure 3(d). Nevertheless, the remaining features in the top 20 exhibit some variations (e.g., the inclusion of the avg. inter-arrival time of outgoing packets in Starlink traffic) which may also be explained by the noise inherent to Starlink connections.

B. Attacks based on deep learning

We now focus on comparing the effectiveness of the DF and Tik-Tok attacks on the network traces we collected.

The success of deep learning attacks is comparable to *k*-FP on plain Firefox traffic. The results in Table II show that the DF and Tik-Tok deep learning attacks achieve a comparable accuracy to the classical machine learning attack *k*-FP when fingerprinting plain Firefox traffic. More closely, we see that the accuracy of DF is comparable to the accuracy obtained by *k*-FP when considering packet lengths. These results suggest that the application of deep learning attacks brings only marginal improvements, if any, for the classification of Firefox traces – for instance, Tik-Tok achieves an accuracy of only 79%, which is around 15% below the accuracy obtained by *k*-FP without considering packet size information.

Deep learning attacks can successfully fingerprint Tor traffic via fiber and Starlink. The accuracy results reported in Table II reveal that the DF and Tik-Tok attacks achieve a similar performance when applied to Tor traffic regardless of whether the traces were collected via fiber or Starlink connections. Interestingly, we also observe that the DF classifier can achieve roughly the same accuracy for plain Firefox traffic collected over fiber (Table I) and Tor traffic, suggesting that users have little benefits when using Tor for shielding their browsing behaviors against website fingerprinting attacks.

Models trained on Starlink data are more robust. Table III presents the accuracy of the Tik-Tok attack when swapping the shares of the dataset used for training and testing the classifier. We can see that using Tor traces collected on the fiber connection to train an attack that aims to fingerprint Tor traffic exchanged via Starlink results in an accuracy decrease of about 4% when compared to the use of Starlink training data. In turn, using Tor traces collected on the Starlink connection to train an attack that aims to fingerprint Tor traffic exchanged via fiber results in an accuracy decrease of only 2% when compared to the use of fiber training data. The above results suggest that an adversary who trains the Tik-Tok attack on traces obtained via Starlink can obtain a relatively high accuracy when fingerprinting both Starlink and fiber traffic. A potential explanation for this fact is that the noise inherent to Starlink may contribute to an increased per-class trace diversity and an overall enhancement of the model's robustness.

Table III: Tik-Tok's accuracy when exchanging the sets of data used for training and, respectively, testing the classifier.

Training Data	Testing Data	Accuracy
Fiber	Fiber	0.8860
Starlink	Starlink	0.8682
Fiber	Starlink	0.8168
Starlink	Fiber	0.8627

Impact of trace length on fingerprinting accuracy. As mentioned in Section III-C, DF and Tik-Tok automatically extract latent features from a trace's direction or direction+timing representation, respectively. In the original attacks, the number of cells considered in each trace (n) is set at 5 000 (samples with lengths below 5 000 are appended with zeros, and those with lengths greater than 5 000 are truncated). Our previous analysis in Section IV – Figure 3(d) revealed that $n = 5\,000$ roughly corresponded to the 75th percentile of cells across all Tor traces (both fiber and Starlink). Our experiments in Appendix C suggest that $n = 5\,000$ is also adequate for classifying Tor traffic in both of our networking environments.

VI. Defending Fiber and Starlink Traces

This section presents the main takeaways of our experiments when assessing the effectiveness and efficiency of existing website fingerprinting defenses when deployed over Tor connections established via fiber and Starlink. As mentioned in Section III-C, we use a set of defense simulators that convert our undefended Tor cell traces into their defended versions.

A. Evaluating the effectiveness of defenses

Table IV lists the accuracy of Tik-Tok on defended Tor traffic. Overall, we can see that the accuracy obtained by the attack for Starlink traces is less than that observed for fiber traces. While this was also true for non-defended traffic (see Table II), constant-rate defenses such as Tamaraw and CS-BuFLO achieve similar accuracy reductions for both fiber and Starlink traces, bringing the attack's accuracy down to approximately 10% and 16%, respectively. This is expected, as both defenses heavily shape the timing and sizes of packets sent to the network to obfuscate traffic patterns.

While other defenses can also moderately decrease the Tik-Tok attack's accuracy, we can observe that the application of these defenses results in disparate effectiveness when applied to fiber and Starlink traces. For instance, we can see that the FRONT_T1 and FRONT_T2 defense variants reduce the attack's accuracy for an extra 12% and 11% when deployed on Starlink traces. While less pronounced, this trend can also be observed for the WTF-PAD defense, where its application to Starlink traces leads to an accuracy reduction of about 5%. These results suggest that the incorporation of dummy traffic, although generally effective on fiber, has a comparatively greater impact on the ability of traffic classifiers to accurately fingerprint Tor connections established over Starlink.

B. Evaluating the overhead imposed by defenses

After gauging the effectiveness of defenses over the Tor traces included in our dataset, we now turn our attention to the comparison of the overheads imposed by these defenses when applied to fiber and Starlink traces. When reporting our results, we present the bandwidth and latency overheads imposed by each defense as the median value of the bandwidth and latency values observed among the defended traces.

Table IV: Tik-Tok accuracy against Tor with different defenses.

Defense	Fiber Traces	Starlink Traces
Tamaraw	0.1087	0.1008
CS-BuFLO	0.1655	0.1540
FRONT_T1	0.5910	0.4700
FRONT_T2	0.5462	0.4358
WTF-PAD	0.8360	0.7880
No defense	0.8860	0.8682

Table V: Defenses' latency and bandwidth overheads.

Defense	Latency Overhead (×)		Bandwidth Overhead (×)	
	Fiber	Starlink	Fiber	Starlink
Tamaraw	5.83	4.28	1.60	1.65
CS-BuFLO	30.54	21.50	1.48	1.47
FRONT_T1	1.00	1.00	1.24	1.31
FRONT_T2	1.00	1.00	1.34	1.46
WTF-PAD	1.00	1.00	1.18	1.21

Defended Starlink traces impose a smaller latency overhead. Table V shows the latency overhead of the considered defenses when applied to fiber and Starlink traces. Overall, one can observe that the latency overhead tends to be the same (or less pronounced) when applied to Starlink traces than when the same defense is applied to fiber traces. Note that the latency overhead is effectively zero on both kinds of traces for adaptive and random padding defenses like FRONT variants and WTF-PAD, since these defenses largely aim to avoid the introduction of communication delays. However, considering Tamaraw and CS-BuFLO, defended Starlink traces impose a latency overhead that is about 1.36 and 1.42 times smaller than that imposed on fiber traces, respectively.

Defended Starlink traces impose a larger bandwidth overhead. Table V also shows the bandwidth overhead of the defenses. For instance, it shows that Tamaraw, the most bandwidth-inefficient defense, imposes an overhead of 1.6 times that of a Tor undefended trace over fiber. We can also see from the table that Starlink traces impose an equivalent or slightly larger bandwidth overhead than that of fiber traces, for all the considered defenses. This increase in overhead is particularly noticeable for the FRONT_T2 defense, where the bandwidth overhead is about 11% larger when the defense is applied to Starlink traces rather than fiber traces.

While the tested defenses allow for a reduction in attack accuracy on Starlink connections (see Table IV), the above analysis reveals that the defenses lead to a small increase in bandwidth usage when compared to their counterpart deployments over fiber. Still, this additional overhead can pose a concern for satellite Internet users (in particular, Starlink users [41]) whose satellite ISPs may apply data caps or limit the amount of high-speed data exchanged by customers towards balancing the supply and demand of traffic [16].

VII. Limitations and Future Work

This section discusses the limitations of our study and points to several directions for future work.

Geo-distributed Starlink testbed. Our evaluation setup considered a single node connected to Starlink. Recent studies [16], [52] have shown that the performance of Starlink client nodes may vary across different continents (or even countries) due to the configuration of the satellite constellation and the number of active subscribers in specific regions. Future work includes the deployment of additional Starlink data collection nodes in different points of the globe, towards understanding whether our findings generalize across locations.

Considering the influence of weather. The weather experienced by our Starlink node during the data collection period was characterized by a clear sky. However, past studies on LEO satellite performance have reported that different weather conditions might affect satellite Internet performance (e.g., imposing additional jitter and latency) [16], [52]. An interesting direction for future work includes the collection of website access traces under different weather conditions (e.g., clouds, rain, snow, etc.) towards assessing whether these conditions result in significant differences in the ability of an adversary to perform accurate website fingerprinting.

Browsing Tor via satellite hopping. Our study is limited to investigating the effectiveness of website fingerprinting attacks over the traffic of users who are connected to the Internet via Starlink. However, it does not consider a potential scenario where some or all of the Tor nodes comprising a circuit are also connected via Starlink up/downlinks. Creating a testbed where multiple legs of a Tor circuit are connected via Starlink (e.g., enabled through Starlink Business fixed sites [40] which provides public IPs) is an interesting direction for future work.

Lack of open-world experiments. Our study focused on website fingerprinting in the closed-world setting. We aim to extend our study to also consider the open-world setting.

Considering performance enhancing proxies (PEPs). PEPs [30] are often deployed on satellite links to improve TCP's performance on satellite links [20]. QPEP [25] wraps users' TCP traffic within a QUIC-based encrypted tunnel to improve performance while protecting users' traffic against eavesdropping. However, QPEP does not actively attempt to shape traffic patterns, and QUIC traffic has been found to be vulnerable to fingerprinting [36]. Studying the resistance against website fingerprinting provided by QPEP deployments over Starlink is an interesting direction for future work.

VIII. Conclusions

In this paper, we evaluated the effectiveness of website fingerprinting attacks on Tor connections established via Starlink, a prominent LEO satellite constellation providing satellite Internet services. Through a synchronous collection of web browsing traces over traditional fiber and Starlink, we characterized Tor accesses over both kinds of links, and compared the effectiveness of website fingerprinting attacks when applied to these different networking settings. Our findings suggest that undefended Tor traffic is equally fingerprintable over Starlink and fiber and that defenses, while effective, may be further parameterized to trade-off security with network efficiency.

Acknowledgments

This work was supported in part by NSERC under grant DGECR-2023-00037, by the NRC-Waterloo Collaboration Center under project reference number 090755, and by the EPSRC New Investigator Award (ref: EP/V011294/1). We also thank Aravindh Raman for helpful discussions about this work.

References

[1] S. Bhat, D. Lu, A. Kwon, and S. Devadas, "Var-cnn: A data-efficient website fingerprinting attack based on deep learning," *Proceedings on Privacy Enhancing Technologies*, vol. 2019, no. 4, pp. 292–310, 2019.

[2] V. Bhosale, A. Saeed, K. Bhardwaj, and A. Gavrilovska, "A characterization of route variability in leo satellite networks," in *Proceedings of the International Conference on Passive and Active Network Measurement*. Springer, 2023, pp. 313–342.

[3] N. Bui and J. Widmer, "Owl: A reliable online watcher for lte control channel measurements," in *Proceedings of the 5th Workshop on All Things Cellular: Operations, Applications and Challenges*, 2016, pp. 25–30.

[4] X. Cai, R. Nithyanand, and R. Johnson, "Cs-buflo: A congestion sensitive website fingerprinting defense," in *Proceedings of the 13th Workshop on Privacy in the Electronic Society*, 2014, pp. 121–130.

[5] X. Cai, R. Nithyanand, T. Wang, R. Johnson, and I. Goldberg, "A systematic approach to developing and evaluating website fingerprinting defenses," in *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, 2014, pp. 227–238.

[6] G. Cherubin, "Bayes, not naïve: Security bounds on website fingerprinting defenses," *Proceedings on Privacy Enhancing Technologies*, vol. 4, pp. 135–151, 2017.

[7] G. Cherubin, R. Jansen, and C. Troncoso, "Online website fingerprinting: Evaluating website fingerprinting attacks on tor in the real world," in *Proceedings of the 31st USENIX Security Symposium*, 2022, pp. 753–770.

[8] R. Dingledine, N. Mathewson, P. F. Syverson *et al.*, "Tor: The second-generation onion router." in *Proceedings of the USENIX Security Symposium*, vol. 4, 2004, pp. 303–320.

[9] J. Gong and T. Wang, "Zero-delay lightweight defenses against website fingerprinting," in *Proceedings of the 29th USENIX Security Symposium*, 2020, pp. 717–734.

[10] J. Gong, W. Zhang, C. Zhang, and T. Wang, "Wfdefproxy: Modularly implementing and empirically evaluating website fingerprinting defenses," *arXiv preprint arXiv:2111.12629*, 2021.

[11] J. Hayes and G. Danezis, "k-fingerprinting: A robust scalable website fingerprinting technique." in *Proceedings of the 25th USENIX Security Symposium*, 2016, pp. 1187–1203.

[12] T. E. Humphreys, P. A. Iannucci, Z. M. Komodromos, and A. M. Graff, "Signal structure of the starlink ku-band downlink," *IEEE Transactions on Aerospace and Electronic Systems*, p. 1–16, 2023.

[13] R. Jansen and R. Wails, "Data-explainable website fingerprinting with network simulation," *Proceedings on Privacy Enhancing Technologies*, vol. 4, pp. 559–577, 2023.

[14] M. Juarez, S. Afroz, G. Acar, C. Diaz, and R. Greenstadt, "A critical evaluation of website fingerprinting attacks," in *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, Scottsdale, Arizona, USA, 2014, p. 263–274.

[15] M. Juarez, M. Imani, M. Perry, C. Diaz, and M. Wright, "Toward an efficient website fingerprinting defense," in *Proceedings of the European Symposium on Research in Computer Security*, 2016, pp. 27–46.

[16] M. M. Kassem, A. Raman, D. Perino, and N. Sastry, "A browser-side view of starlink connectivity," in *Proceedings of the 22nd ACM Internet Measurement Conference*, 2022, pp. 151–158.

[17] K. Kohls, D. Rupprecht, T. Holz, and C. Pöpper, "Lost traffic encryption: fingerprinting lte/4g traffic on layer two," in *Proceedings of the 12th Conference on Security and Privacy in Wireless and Mobile Networks*, 2019, pp. 249–260.

[18] S. Ma, Y. C. Chou, H. Zhao, L. Chen, X. Ma, and J. Liu, "Network characteristics of leo satellite constellations: A starlink-based measurement from end users," in *Proceedings of the 2023 IEEE Conference on Computer Communications*, 2023, pp. 1–10.

[19] N. Mathews, J. K. Holland, S. E. Oh, M. S. Rahman, N. Hopper, and M. Wright, "Sok: A critical evaluation of efficient website fingerprinting defenses," in *Proceedings of the 44th IEEE Symposium on Security and Privacy*, 2023, pp. 969–986.

[20] F. Michel, M. Trevisan, D. Giordano, and O. Bonaventure, "A First Look at Starlink Performance," in *Proceedings of the 22nd ACM Internet Measurement Conference*, Nice, France, October 2022, pp. 130–136.

[21] M. Neinavaie and Z. M. Kassas, "Signal mode transition detection in starlink leo satellite downlink signals," in *2023 IEEE/ION Position, Location and Navigation Symposium (PLANS)*, 2023, pp. 360–364.

[22] One Web, https://oneweb.net/, last Accessed: 2024-02-09.

[23] Open source SDR 4G software suite, https://github.com/srsran/srsRAN_4G , last Accessed: 2024-02-09.

[24] A. Panchenko, F. Lanze, J. Pennekamp, T. Engel, A. Zinnen, M. Henze, and K. Wehrle, "Website fingerprinting at internet scale." in *Proceedings of the 23rd Annual Network and Distributed System Security Symposium*, 2016.

[25] J. Pavur, M. Strohmeier, V. Lenders, and I. Martinovic, "QPEP: An Actionable Approach to Secure and Performant Broadband From Geostationary Orbit," in *Proceedings of the 28th Network and Distributed System Security Symposium*, 2021.

[26] J. Pavur and I. Martinovic, "Sok: Building a launchpad for impactful satellite cyber-security research," in *arXiv cs.CR 2010.10872*, 2020.

[27] V. L. Pochat, T. Van Goethem, S. Tajalizadehkhoob, M. Korczyński, and W. Joosen, "Tranco: A research-oriented top sites ranking hardened against manipulation," in *Proceedings of the Network and Distributed Systems Security Symposium*, 2019.

[28] M. S. Rahman, P. Sirinam, N. Mathews, K. G. Gangadhara, and M. Wright, "Tik-tok: The utility of packet timing in website fingerprinting attacks," *Proceedings on Privacy Enhancing Technologies*, vol. 2020, no. 3, 2020.

[29] A. Raman, M. Varvello, H. Chang, N. Sastry, and Y. Zaki, "Dissecting the performance of satellite network operators," *Proc. ACM Netw.*, vol. 1, no. CoNEXT3, nov 2023.

[30] RFC 3135 - Performance Enhancing Proxies Intended to Mitigate Link-Related Degradations, https://www.rfc-editor.org/rfc/rfc3135, last Accessed: 2024-02-09.

[31] RFC 8446 - The Transport Layer Security (TLS) Protocol Version 1.3, https://www.rfc-editor.org/rfc/rfc8446, last Accessed: 2024-02-09.

[32] V. Rimmer, D. Preuveneers, M. Juarez, T. Van Goethem, and W. Joosen, "Automated website fingerprinting through deep learning," in *Proceedings of the 25th Network and Distributed Systems Security Symposium*, 2018.

[33] Rogers Media Inc. v. John Doe 1, 2022 FC 775 (CanLII), https://canlii.ca/t/jpncf, last Accessed: 2024-02-09.

[34] D. Rupprecht, K. Kohls, T. Holz, and C. Pöpper, "Breaking lte on layer two," in *Proceedings of the IEEE Symposium on Security and Privacy*, 2019, pp. 1121–1136.

[35] M. Shen, K. Ji, Z. Gao, Q. Li, L. Zhu, and K. Xu, "Subverting website fingerprinting defenses with robust traffic representation," in *Proceedings of the 32nd USENIX Security Symposium*, 2023, pp. 607–624.

[36] S. Siby, L. Barman, C. Wood, M. Fayed, N. Sullivan, and C. Troncoso, "Evaluating practical quic website fingerprinting defenses for the masses," *Proceedings on Privacy Enhancing Technologies*, vol. 4, pp. 79–95, 2023.

[37] P. Singh, D. Barradas, T. Elahi, and N. Limam, "Website access traces for Tor/Firefox over Starlink and fiber links," Feb. 2024. [Online]. Available: https://doi.org/10.5281/zenodo.10641853

[38] P. Sirinam, M. Imani, M. Juarez, and M. Wright, "Deep fingerprinting: Undermining website fingerprinting defenses with deep learning," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, 2018, pp. 1928–1943.

[39] Starlink, https://www.starlink.com/, last Accessed: 2024-02-09.

[40] Starlink Business - Fixed Site , https://www.starlink.com/business/fixed-site , last Accessed: 2024-02-09.

[41] Starlink Fair Use Policy , https://www.starlink.com/legal/documents/DOC-1134-82708-70 , last Accessed: 2024-02-09.

[42] Q. Sun, D. R. Simon, Y.-M. Wang, W. Russell, V. N. Padmanabhan, and L. Qiu, "Statistical identification of encrypted web browsing traffic," in *Proceedings of the 23rd IEEE Symposium on Security and Privacy*, Oakland, CA, USA, 2002, pp. 19–30.

[43] Tranco list generated on September 2022, https://tranco-list.eu/list/82GXV, last Accessed: 2024-02-09.

[44] F. Vatalaro, G. Corazza, C. Caini, and C. Ferrarelli, “Analysis of leo, meo, and geo global mobile satellite systems in the presence of interference and fading,” *IEEE Journal on Selected Areas in Communications*, vol. 13, no. 2, pp. 291–300, 1995.

[45] A. Veicht, C. Renggli, and D. Barradas, “Deepse-wf: Unified security estimation for website fingerprinting defenses,” *Proceedings on Privacy Enhancing Technologies*, vol. 2023, no. 2, 2023.

[46] T. Wang, X. Cai, R. Nithyanand, R. Johnson, and I. Goldberg, “Effective attacks and provable defenses for website fingerprinting.” in *Proceedings of the 23rd USENIX Security Symposium*, 2014, pp. 143–157.

[47] T. Wang and I. Goldberg, “Improved website fingerprinting on tor,” in *Proceedings of the 12th ACM workshop on Workshop on privacy in the electronic society*, 2013, pp. 201–212.

[48] Y. Zhang, Q. Wu, Z. Lai, and H. Li, “Enabling Low-latency-capable Satellite-Ground Topology for Emerging LEO Satellite Networks,” in *Proceedings of the 2022 IEEE Conference on Computer Communications*, Virtual Event, May 2022, pp. 1329–1338.

[49] Y. Zhang, S. Zhao, J. He, Y. Zhang, Y. Shen, X. Jiang *et al.*, “A survey of secure communications for satellite internet based on cryptography and physical layer security,” *IET Information Security*, vol. 2023, 2023.

[50] Z. Zhang, T. Vaidya, K. Subramanian, W. Zhou, and M. Sherr, “Ephemeral exit bridges for tor,” in *Proceedings of the 50th Annual IEEE/IFIP International Conference on Dependable Systems and Networks*, 2020, pp. 253–265.

[51] Z. Zhang, W. Zhou, and M. Sherr, “Bypassing tor exit blocking with exit bridge onion services,” in *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*, 2020, p. 3–16.

[52] H. Zhao, H. Fang, F. Wang, and J. Liu, “Realtime multimedia services over starlink: A reality check,” in *Proceedings of the 33rd Workshop on Network and Operating System Support for Digital Audio and Video*, 2023, pp. 43–49.

Appendix

A. *Websites included in our dataset*

Listing 1 includes the websites contained in our dataset (drawn from the September 2022 Tranco list [43]). The majority of these websites are still considered relevant: 75% of the websites we consider are within the top 250 websites in the January 2024 Tranco list (the most up to date list upon this work’s submission for peer review).

B. *Defense configurations*

We incorporated WTF-PAD [15] into our setup, utilizing the implementation provided in the WFES [6] repository. Additionally, we have integrated two versions of FRONT [9] into our experiments, named FRONT_T1 and FRONT_T2. FRONT_T1 is configured with parameters $N_c = N_s = 1700$, $W_{min} = 1$, and $W_{max} = 14$, whereas FRONT_T2 uses different settings, with $N_c = N_s = 2500$. Due to the larger sampling window in FRONT_T2, it is expected to introduce more dummy packets into the trace than FRONT_T1. For CS-BuFLO [4] and Tamaraw [5], we employed another set of parameters specific to these defenses. CS-BuFLO is set with $d = 1$ and a range for $2^{-4} * 1000 \leq \rho \leq 2^3 * 1000$. Tamaraw uses $\rho_{out} = 0.04$, $\rho_{in} = 0.012$ with $L = 50$.

C. *Finding a suitable trace length*

Figure 6 depicts the evolution of the accuracy of the Tik-Tok classifier, according to the length (n) of each trace. We can observe that the trend is rather similar for both our Tor fiber and Starlink traces. The figure also shows that an n smaller than 4 000 prevents the classifier from achieving an accuracy over 86% for both connection types but that $n = 5\,000$ provides an adequate trade-off between the length of input traces and the accuracy obtained by the classifier.

```
0.  adobe.com
1.  amazon.co.jp
2.  amazon.in
3.  apache.org
4.  apple.com
5.  azure.com
6.  bbc.co.uk
7.  bbc.com
8.  bing.com
9.  bit.ly
10. booking.com
11. cdc.gov
12. cnn.com
13. digicert.com
14. dnsmadeeasy.com
15. doubleclick.net
16. dropbox.com
17. ebay.com
18. etsy.com
19. facebook.com
20. fandom.com
21. fastly.net
22. fbcdn.net
23. flickr.com
24. force.com
25. gandi.net
26. github.com
27. github.io
28. google-analytics.com
29. googledomains.com
30. icloud.com
31. instagram.com
32. intuit.com
33. issuu.com
34. linode.com
35. live.com
36. mail.ru
37. microsoft.com
38. mozilla.org
39. msn.com
40. naver.com
41. netflix.com
42. nytimes.com
43. office365.com
44. opera.com
45. oracle.com
46. outlook.com
47. paypal.com
48. pornhub.com
49. reddit.com
50. reuters.com
51. salesforce.com
52. salesforceliveagent.com
53. skype.com
54. soundcloud.com
55. sourceforge.net
56. spotify.com
57. stackoverflow.com
58. t.me
59. telegram.org
60. theguardian.com
61. tiktok.com
62. tumblr.com
63. twitch.tv
64. vimeo.com
65. w3.org
66. weebly.com
67. wellsfargo.com
68. whatsapp.com
69. wikimedia.org
70. wikipedia.org
71. xvideos.com
72. yahoo.co.jp
73. youtube.com
74. zemanta.com
```

Listing 1: List of websites considered in our experiments.

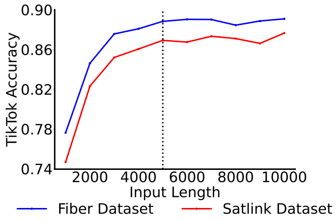


Figure 6: Trade-off between trace length and Tik-Tok accuracy.