Website Fingerprinting Attacks with Advanced Features on Tor Networks

Donghoon Kim
Department of Computer Science
Arkansas State University
Jonesboro, USA
dhkim@astate.edu

Andrew Booth
Department of Computer Science
Arkansas State University
Jonesboro, USA
andrew.booth@smail.astate.edu

Olusegun Akinyemi
Department of Computer Science
Arkansas State University
Jonesboro, USA
olusegun.akinyemi@smail.astate.edu

Euijin Choo
Department of Computer Science
University of Alberta
Edmonton, Canada
euijin@ualberta.ca

Doosung Hwang
Department of Software Science
Dankook University
Youngin, South Korea
dshwang@dankook.ac.kr

2024 Cyber Awareness and Research Symposium (CARS) | 979-8-3503-8641-7/24/$31.00 ©2024 IEEE | DOI: 10.1109/CARS61786.2024.10778686

***Abstract*—The Tor network has been discovered to be susceptible to website fingerprinting (WF) attacks. Previous research has primarily been conducted in controlled experimental environments, leading to debates about the feasibility of WF attacks in real-world environments. Recent advancements in feature engineering and machine learning models have aimed to bridge this gap by exploring real-world scenarios. Nonetheless, designing innovative features for WF attacks on Tor networks remains a significant challenge. To address these challenges, this research explores real-world environments using a novel method that extracts network traffic data by filtering out common traffic based on onion service protocols and advanced features that optimize various advantages of different features. The results suggest that WF attacks in real-world environments become more feasible with the proposed method and carefully crafted features. This study contributes to our understanding of the feasibility of WF attacks on Tor networks in real-world environments and can help identify potential security enhancements on Tor networks.**

***Index Terms*—Tor network, onion service, website fingerprinting attack, vulnerability, machine learning**

I. Introduction

Tor (The Onion Router) is a network and software tool that enables users to browse the Internet anonymously, protecting their privacy and security. Tor browser, built on Firefox, can access both general websites (non-hidden) and onion services (hidden) [1], [2]. General websites, such as "www.google.com," are accessible through standard web browsers like Chrome, Safari, and Mozilla Firefox. In contrast, onion services are exclusively accessible via the Tor network using the Tor Browser. Instead of traditional DNS name resolution, they use unique addresses with ".onion" extensions, resulting in unique URLs like "vnwf42cmztzqfojlrehvyd45bts.onion." When accessing onion services, the browser does not communicate directly with the service. Instead, it uses a chosen "rendezvous point" to exchange information [3]. This process is crucial for protecting user privacy and identity, and for understanding Tor network vulnerabilities [4]–[6].

Previous research has demonstrated that website fingerprinting (WF) attacks using machine learning can successfully extract information to identify onion services by analyzing large amounts of network traffic on Tor network [1], [2], [7]. However, questions have been raised about whether such WF attacks are actually feasible in the real world because of the presumptions made in the planning and assessment of these attacks. Concretely, they often have experimented without assumptions including real traffic variability and diverse network conditions [8], [9]. This shift in research direction recognizes the limitations of existing features and machine learning models developed for WF attacks and promotes research that enhances privacy and security on the Tor network. This is crucial, as certain activities or misconfigurations can still compromise user anonymity in real-world scenarios on the Tor network.

This study aims to explore the feasibility of WF attacks in real-world environments utilizing advanced technologies. To accomplish these goals, the study employs a two-part approach. *Firstly*, it aims to enhance the efficacy of WF attacks by employing a novel method we have designed by analyzing the onion service protocols. *Secondly*, we advance various features by balancing the strengths and weaknesses of previous features for real-time WF attacks. We have designed a novel method to enhance WF attacks by removing initial packets that typically contain commonly occurring packets generated by the Tor protocol for accessing onion services. Additionally, we have discovered that previous features used in the literature can be advanced. Specifically, Panchenko *et al.*'s CUMUL [2], which uses packet quantity and number, and Kim *et al.*'s 125 features [10], which leverage different packet information, both have their respective strengths and limitations. The CUMUL features exhibit good accuracy in identifying general websites with large-scale datasets of webpages. However, it may not be suitable for tracking changes in websites over time due to the limitation of its monotonous features, which primarily include the number and size of packets [2], [11]. Kim *et al.*'s 125 features, based on 103 features [11] demonstrated strong adaptability to changes in websites over time [10], [11]. However, they had limitations compared to CUMUL in identifying specific characteristics, primarily due to variations in packet number and size influenced by webpage design [10]. Since onion services and Tor browsers meet at rendezvous points rather than at the physical locations of onion service servers, features such as time intervals related to location and the first 30 packets, which are included in the 125 features, may not be critical components for WF attacks. Thus, we advance these features to effectively handle changes over time and perform WF attacks by carefully crafting integrated features that consider their individual strengths and weaknesses. The contributions of this study are as follows:

- Our framework involves the collection of a comprehensive dataset comprising genuine general websites and onion services. This dataset serves as the foundation for our analytical investigations, providing real-world examples crucial for assessing the effectiveness of our methods.
- We propose innovative methodologies that involve the design and implementation of intricate features. These features are meticulously crafted by analyzing the characteristics of onion service protocol. By processing raw network data through these features, we aim to significantly enhance the precision and efficiency of WF attacks, thereby advancing the state-of-the-art in this field.
- This empirical approach allows us to validate the feasibility and effectiveness of our proposed methods under realistic conditions, highlighting their potential applicability in safeguarding network security.

This structure of this paper is as follows. Section II provides a review of relevant research, focusing on various features and real-world WF attacks. Section III provides background information on onion services protocols, discusses the key features employed in this study for WF attacks, presents a threat model, and outlines the data collection process. Section IV details two experiments conducted based on our two approaches and analyzes their outcomes. Finally, Section V summarizes the findings and conclusions of this study.

II. Related Work

Many researchers demonstrated that WF attacks on the Tor networks are successful in identifying both general websites and onion services with high performance metrics and high volume of dataset [1], various features [2], under closed [2], [12]–[15] and open [16]–[18] world models. Kwon *et al.* [1] proposed novel attack models leveraging the vulnerabilities of the Tor network's onion service by analyzing incoming and outgoing cells. Yan and Jasleen [19] conducted a comprehensive feature analysis across eight different communication scenarios and compared their findings to the work of Kwon *et al.*'s [1]. They categorized features into five levels (packet, burst, TCP, Port, and IP address) and enumerated the most informative ones for each scenario. Lashkari *et al.* [12] utilized 23 time-related features to classify Tor and non-Tor data, as well as to identify 9 different applications. Panchenko *et al.* [2] investigated the practical limitations of website fingerprinting on an Internet-wide scale, analyzing over 300,000 webpages using CUMUL features, which leverage the cumulative sum of packet sizes and counts.

As the performance of WF attacks has improved, researchers have expanded their studies from controlled environments to real-world environments [8]. Wang [20] examined the effectiveness of WF attacks in the real open world. They found that the existing attacks have limited precision when the base rate is low. To address this limitation, they proposed three novel optimized classifiers that can handle low base rate scenarios. Deng *et al.* [21] conducted WF attacks to identify websites in multi-tab browsing sessions. Cherubin *et al.* [8] used Triplet Fingerprinting attack [22] in an online setting, demonstrating that an adversary can achieve over 95% accuracy in WF classification when monitoring a small set of 5 popular websites. However, this accuracy quickly drops to below 80% when the number of monitored websites increases to 25. Bahramali *et al.* [9] demonstrated augmenting network traces to accurately reflect variations in network conditions similar to real-world environments. Jansen *et al.* used GTT23 which is "Genuine Tor Traces" measured in 2023 to uncover discrepancies in synthetic WF datasets that misrepresent authentic characteristics [23]. Despite their success in executing WF attacks in real-world environments, it is clear that the pursuit of improved features remains an ongoing necessity.

III. Methodology

Background. This study employs two main sets of features to evaluate and enhance WF attacks: CUMUL and 125 features. Panchenko *et al.* [2] designed the CUMUL feature. CUMUL has 104 features derived from the total number and size of packets. These features consist of four key elements: the number of incoming and outgoing packets, and the total volume of incoming and outgoing data. Additionally, there are 100 additional features that are calculated as cumulative sums of packets using linear interpolation [2]. The 125 features consist of diverse information, including packet general information, cell sequence length, packet inter-arrival time, burst information, cell ordering, and concentration based on 103 features [10], [11]. Oh *et al.* [11] found that the 103 features outperformed CUMUL when analyzing data from websites that frequently update their content over time. Detailed descriptions for each feature are provided in Table I in Section IV-A.

Threat Model. An attacker can monitor the traffic between the client and the first router (entry guard) in the Tor network [2], as well as between the last router and the destination client, with-

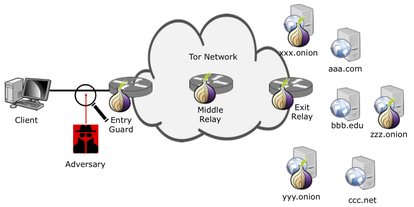

Fig. 1: The Threat Model.

out needing to decrypt the packets. The attackers may be the owner of a Tor router, an internet service provider, or a network administrator. In this study, we consider a scenario where an attacker is monitoring traffic within the same broadcast domain as both the client and the entry guard as the threat model illustrated in Figure 1. Because it highlights the significant risks and vulnerabilities associated with user identity exposure. By focusing on attackers such as Tor router owners, internet service providers, or network administrators, we address a range of potential threats that are particularly relevant in today's digital landscape. This scenario emphasizes the importance of understanding how WF attacks can compromise user anonymity on Tor networks.

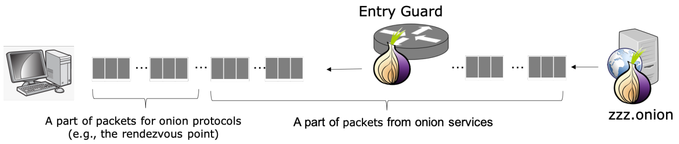
Entry Guard
zzz.onion
A part of packets for onion protocols
(e.g., the rendezvous point)
A part of packets from onion services

Fig. 2: Packet Flow in the Tor Network: From Onion Service to Client via Entry Guard

Approach. Our research approach involves a comprehensive analysis, focusing on the feasibility and effectiveness of WF attacks in real-world environments. Given the success of WF attacks in controlled settings, researchers have become increasingly interested in their performance in real-world scenarios [24]. As a result, research has shifted towards examining WF attacks beyond the limitations of controlled settings [9], [23]. This study aims to explore the vulnerabilities of the Tor network through WF attacks, particularly in real-world environments. To evaluate the feasibility of WF attacks on Tor, we have developed methods aimed at enhancing performance, including reducing computation time and utilizing enhanced features. When accessing onion services with a Tor browser, the protocol employed differs from that used for general websites. Instead of receiving messages directly from the onion service, the Tor browser and the onion service meet at a specific rendezvous point [3]. The onion service protocol consists of several stages that require the exchange of multiple network packets between the Tor browser and the onion service in order to establish the rendezvous point. The overall packets in the Tor browser encompass both Tor protocol packets and onion services packets, as depicted in Figure 2. These network packets are commonly encountered within the onion service protocol at the beginning of connection establishment [9]. In contrast, connecting to a general website involves a basic protocol that typically requires a relatively small number of packets. Given this context, it is reasonable to assume that the removal of common packets associated with connection establishment from the collected network data will have a significant impact on website fingerprinting attacks for onion services.

Moreover, for WF attacks to be successful in real-world environments, it is essential to optimize the feature vectors for real-time execution. We examine the feasibility of real-time WF attacks by developing novel features that represent an optimal combination derived from various feature sets including CUMUL [2] and 125 features [10]. Both CUMUL [2] and the 125 features [10] demonstrate reliable performance, each with its own unique set of strengths and weaknesses. CUMUL is effective at identifying static data, but it may not be as effective at identifying dynamic websites because it only takes into account the number and size of packets, rather than other factors. By leveraging the strengths of CUMUL and complementing its limitations with the advantages of 125 features, we aim to develop high-performance features without increasing their number. This approach could enable WF attacks to be viable in real-time scenarios. Research focused on enhancing performance and reducing computation time will be extensively discussed and experimented with in Experiments 1 and 2 in Section IV-A and IV-B.

Data Collection. We use the network traffic collection system we developed to gather data. Data for onion services is collected through `ahmia.fi`. The system offers several options to collect the most relevant data for real-world environments. For instance, it can decide whether to continue collecting traffic from a site or to move on to the next website. Alternatively, it can collect data from sites on a list once and then revisit them multiple times. Since website content frequently changes, the performance of machine learning models may vary depending on when the dataset was trained [11]. Therefore, it is crucial to collect data that best reflects the actual environment.

Moreover, the size of the collected data varies depending on the framework environment and the specific types of onion services and general websites. For this study, the data includes .pcap files, with approximately 19.76 GB for 30 general websites and around 14.47 GB for 30 onion services.

IV. Experiments

A. Experiment 1 (EX1): Removing Initial Packets

We acknowledge that filtering out common packets related to connection establishment from the collected network data will significantly affect WF attacks. To investigate the impact of these common packets on WF attacks, EX1 performs experiments where initial common packets—present in both onion services and general websites—are removed to evaluate the effectiveness of website fingerprinting attacks. Specifically, we examine how this modification influences the effectiveness of the attack, given the presence of common network packets

TABLE I: Extracted Features

Features	Contents in a Feature	Count
104 CUMUL [2]	The number of incoming packets	1
	The number of outgoing packets	1
	The sum of incoming packets	1
	The sum of outgoing packets	1
	Cumulative sum of packets based on linear interpolation	100
125 features [10]	Packet general information	44
	Cell sequence length	4
	Packet Inter arrival time	4
	Burst information	22
	Cell ordering	18
	Concentration	8
140 features	Burst information	9
	Concentration	9
	Packet ordering	18
	CUMUL	104

in both onion services and general websites. EX1 uses two previous works as feature vectors as in Table I. The first work is CUMUL, which uses 104 features solely focused on packet count and volume [2]. The second work is 125 features covering a wider range of network metrics such as packet inter-arrival time, sequencing, and burst duration [10]. The rationale for this will be explained in detail in Section IV-B.

To study the impact of Tor network protocol packets on WF attacks with respect to feature extraction, the first few packets were removed before extracting two sets of features: 125 features [10], and 104 CUMUL [2]. Classification was performed using Decision Tree (DT), Random Forest (RF), Extra Tree (XT), and XGBoost (XGB).

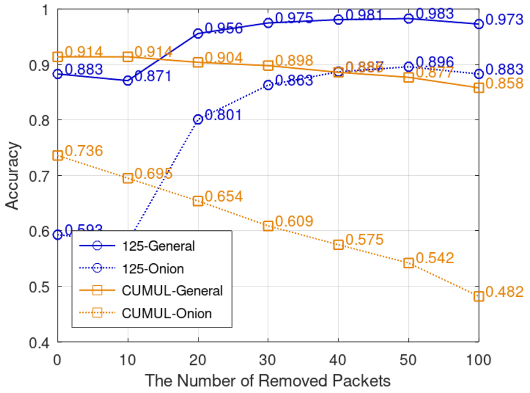

Fig. 3: EX1: Removing Initial Packets on RF

The accuracy results are visualized in Figure 3, where `125-general` and `125-onion` is for 125 features, and `104-general` and `104-onion` is for 104 CUMUL. In the figure, the X-axis represents the number of removed initial packets, and the Y-axis shows the accuracy. When no packets are removed (indicated as 0 on the X-axis), 104 CUMUL performs slightly better than the 125 features approach. For 30 general websites, 104 CUMUL achieves an accuracy of 91.4%, compared to 88.3% for the 125 features approach on RT. The removal of packets can lead to two conflicting trends in website fingerprinting attacks. The accuracy of the 125 feature increases, while the accuracy of the 104 CUMUL decreases. For the 125 features of general websites, the accuracy improves with the removal of more packets, reaching a maximum of 98.3% after 50 packets are removed. Similarly, for the 125 features of onion services, the accuracy improves with the removal of more packets, reaching a maximum of 89.6% (from 59.3%) after 50 packets are removed. Conversely, when packets are removed in CUMUL, accuracy tends to decrease; For general websites, the accuracy is 91.4% without any packets removed, and drops to 87.7% when 50 packets are removed. For onion services, the accuracy is 73.6% without removing, and the accuracy is 54.2% when 50 packets are removed. The two opposing trends observed can be explained by the fact that the 125 features consist of various contents beyond the number of packets. Even after removing common packets for onion protocols, the remaining elements may still retain the distinguishing characteristics of the website, highlighting other features. However, for CUMUL, the number and amount of packets are crucial for its accuracy. Removing packets can significantly alter this information, weakening the CUMUL feature's distinguishing characteristics.

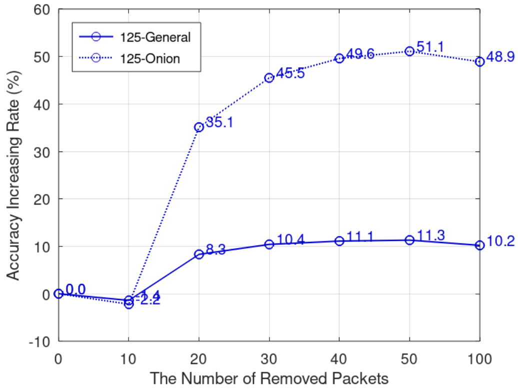

Fig. 4: EX1: Increasing Rates using Accuracy on RF

The analysis presented in Figure 4 illustrates how the accuracy improvement rates vary. In the figure, the X-axis shows the number of removed packets, and the Y-axis represents the rate of accuracy improvement. For 125 features, there was an 11.3% enhancement in accuracy observed on the general website, rising from 88.3% to 98.3%. On the onion services, the accuracy exhibited a more substantial increase of 51.1%, climbing from 59.3% to 89.6%. These results highlight a more clear improvement rate in the case of the onion services using 125 features. This is likely due to the onion services' more complex protocol, which generates a large number of common packets. By removing a significant portion of these common initial packets, the accuracy of the onion services was able to improve significantly for 125 features. On the other hand, in the case of CUMUL, removing packets is detrimental because it disrupts feature stability. The removal alters the number or quantity of packets, making it challenging

TABLE II: EX2. CUMUL Feature Manipulation with 4+ Interpolation Features (Training Time in Seconds).

Model	Category	Metrics	ALL (4+100)	First 10 (4+6)	First 20 (4+16)	First 30 (4+26)	First 40 (4+36)
RF	30 general	Accuracy	0.914	0.905	0.905	0.905	0.905
		Precision	0.916	0.909	0.908	0.907	0.908
		Recall	0.914	0.905	0.905	0.905	0.905
		F1-score	0.913	0.905	0.905	0.904	0.905
		Training time	8.757	0.857	1.745	2.597	3.457
	30 onion	Accuracy	0.736	0.687	0.684	0.677	0.674
		Precision	0.738	0.686	0.685	0.676	0.671
		Recall	0.736	0.687	0.684	0.677	0.674
		F1-score	0.734	0.683	0.681	0.673	0.669
		Training time	10.783	1.018	2.101	3.168	4.217
XGB	30 general	Accuracy	0.923	0.914	0.907	0.900	0.893
		Precision	0.926	0.917	0.910	0.904	0.896
		Recall	0.923	0.914	0.907	0.900	0.893
		F1-score	0.922	0.913	0.907	0.900	0.893
		Training time	26.356	26.041	26.598	27.032	27.590
	30 onion	Accuracy	0.740	0.739	0.737	0.736	0.738
		Precision	0.743	0.742	0.737	0.738	0.741
		Recall	0.740	0.738	0.736	0.734	0.737
		F1-score	0.739	0.737	0.734	0.733	0.736
		Training time	33.676	33.585	33.536	34.044	33.582

TABLE III: EX2. Multi classification.

Model	Metrics	Feature vectors			
		81	90	140	CUMUL
RF	Accuracy	0.927	0.924	0.924	0.885
	Precision	0.934	0.930	0.930	0.886
	Recall	0.928	0.922	0.924	0.886
	F1-score	0.928	0.922	0.924	0.886
	Time	0.317	0.283	0.344	1.074
XGB	Accuracy	0.920	0.917	0.917	0.893
	Precision	0.920	0.920	0.920	0.900
	Recall	0.920	0.920	0.920	0.890
	F1-score	0.920	0.920	0.920	0.890
	Time	2.472	2.635	3.595	4.852

to maintain consistent features. We recognize that determining the exact number of common initial packets involved in the onion services' protocol remains a major research task, which we leave as future work.

B. Experiment 2 (EX2): Advanced features

To enable real-time WF attacks, we must minimize preprocessing and training times. We evaluated several feature sets, including CUMUL [2] and 125 features [10], to balance their advantages and drawbacks. By reducing equidistant points and optimally integrating CUMUL with 125 features, we achieve real-time performance.

To evaluate CUMUL with a subset of features, we conducted an experiment. Results using a portion of CUMUL features are shown in Table II. `All (104)` refers to the initial 4 features for packet numbers and 100 features from linear interpolation, totaling all CUMUL features. `First 10` means the initial 4 features for the overall number and quantity of packets and the initial 6 features from linear interpolation features. Including all the features of CUMUL yields optimal performance. However, even when using only some of its features, the performance does not degrade significantly. Specifically, general websites fall within a margin of approximately 2%, while onion services remain within a margin of up to 7% of RF.

To leverage the advantages of CUMUL including our findings in Table II while addressing its limitations, this study has developed a reduced set of features for real-time fingerprinting attacks by optimally combining CUMUL features [2] with the 125 features [10]. The new feature vectors, comprising 81, 90, and 140 features, are created by selecting common features and combining feature vectors from both previous works. There are 40 common features, which consist of 36 features selected from a pool of 125 based on the feature importance calculated by a random forest classifier, and 4 basic features extracted from CUMUL. The remaining features are obtained through linear interpolation from CUMUL. In the case of the 81-feature set, 36 features were selected from the pool of 125 features based on feature importance, while four basic features were extracted from CUMUL, and 41 features were obtained from linear interpolation features in CUMUL. The 90-feature set included 50 interpolation features from CUMUL, while the 140-feature set contained all 104 CUMUL features, along with the 36 features from the 125 features.

In this experiment, we perform multi-class classification across 10 distinct onion services. Results with three different sets of features, including 81 features, 90 features, and 140 features and CUMUL, are presented in Table III. The feature vectors were assessed using performance metrics such as Accuracy, Precision, Recall, and F1-score, along with training and testing time. However, due to page constraints, the experimental results only included accuracy and time.

Three different sets of features (about 92%) show higher performance than CUMUL (89%) on RF. The rationale behind this is that CUMUL solely incorporates features related to packet counts, while the three different sets of features of additional features enhance the basic characteristics of CUMUL by providing information about diverse aspects of network traffic such as packet ordering and burst interval time. In RF, three different sets of features show 3 times lower training time than CUMUL, primarily because they employ a reduced number of features. We observed similar performance, i.e., between 92.7% and 92.4%, for three different sets of features with RF. When using 81 features, both RF and XGB showed the highest performance, with RF achieving 92.7% and XGB achieving 92.0%. The main difference between the three sets of features is the required training time. When using XGB, 81 features resulted in the fastest training and testing time, followed by 90 features. The analysis results indicate that a model with equivalent performance can be created using fewer features, while also reducing the training time. Our experimental results demonstrate that the features we designed significantly improve the effectiveness of real-time WF attacks.

V. Conclusion

This study explored the vulnerabilities of the Tor network through WF attacks, particularly utilizing novel methods and advanced features in real-world environments. By leveraging novel methods that manipulate network traffic data based on onion service protocols and advanced features that capitalize on the advantages of different features, we demonstrated that WF attacks are not only feasible but also effective in real-world settings. The findings of this study are significant as they highlight the ongoing security challenges faced by the Tor network. The demonstrated effectiveness in real-world environments emphasizes the urgent need for enhanced security measures. Addressing these vulnerabilities is crucial for maintaining the integrity and privacy of the Tor network. However, the study also encountered certain limitations, such as the variability in real-world data and the challenges of maintaining consistent performance across large datasets. These limitations suggest that while significant progress has been made, there is still much to be done to fully mitigate the risks of WF attacks. Future research should focus on further refining real-time WF attack models and feature vectors to enhance their effectiveness and efficiency in real-world environments, ensuring that the Tor network can better protect its users against these threats.

Acknowledgment

This material is based upon work supported by the National Science Foundation under Award No. OIA-1946391.

References

[1] Albert Kwon, Mashael AlSabah, David Lazar, Marc Dacier, and Srinivas Devadas. Circuit fingerprinting attacks: Passive deanonymization of tor hidden services. In *24th {USENIX} Security Symposium ({USENIX} Security 15)*, pages 287–302, 2015.

[2] Andriy Panchenko, Fabian Lanze, Jan Pennekamp, Thomas Engel, Andreas Zinnen, Martin Henze, and Klaus Wehrle. Website fingerprinting at internet scale. In *NDSS*, 2016.

[3] *How Do Onion Services Work?*, (accessed June 21, 2024). https://community.torproject.org/onion-services/overview/.

[4] Florian Platzer, Marcel Schäfer, and Martin Steinebach. Critical traffic analysis on the tor network. In *Proceedings of the 15th International Conference on Availability, Reliability and Security*, pages 1–10, 2020.

[5] Mohamad Amar Irsyad Mohd Aminuddin, Zarul Fitri Zaaba, Azman Samsudin, Nor Badrul Anuar Juma'at, and Sazali Sukardi. Analysis of the paradigm on tor attack studies. In *2020 8th International Conference on Information Technology and Multimedia (ICIMU)*, pages 126–131. IEEE, 2020.

[6] Yixin Sun, Anne Edmundson, Laurent Vanbever, Oscar Li, Jennifer Rexford, Mung Chiang, and Prateek Mittal. {RAPTOR}: Routing attacks on privacy in tor. In *24th {USENIX} Security Symposium ({USENIX} Security 15)*, pages 271–286, 2015.

[7] Tao Wang. High precision open-world website fingerprinting. In *2020 IEEE Symposium on Security and Privacy (SP)*, pages 152–167, 2020.

[8] Giovanni Cherubin, Rob Jansen, and Carmela Troncoso. Online website fingerprinting: Evaluating website fingerprinting attacks on tor in the real world. In *31st USENIX Security Symposium (USENIX Security 22)*, pages 753–770, 2022.

[9] Alireza Bahramali, Ardavan Bozorgi, and Amir Houmansadr. Realistic website fingerprinting by augmenting network traces. In *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security*, pages 1035–1049, 2023.

[10] Donghoon Kim, Loc Ho, Young-Ho Kim, Won-gyum Kim, and Doosung Hwang. Poster: A pilot study on real-time fingerprinting for tor onion services. In *The Network and Distributed System Security Symposium (NDSS)*, 2021.

[11] Hyungseok Oh, Donghoon Kim, Won-gyum Kim, and Doosung Hwang. Performance analysis of tor website fingerprinting over time using tree ensemble models. In *2020 International Conference on Computational Science and Computational Intelligence (CSCI 2020)*, 2020.

[12] Arash Habibi Lashkari, Gerard Draper-Gil, Mohammad Saiful Islam Mamun, and Ali A Ghorbani. Characterization of tor traffic using time based features. In *ICISSP*, pages 253–262, 2017.

[13] Vera Rimmer, Davy Preuveneers, Marc Juarez, Tom Van Goethem, and Wouter Joosen. Automated website fingerprinting through deep learning. *arXiv preprint arXiv:1708.06376*, 2017.

[14] Wang, Tao. Website fingerprinting: Attacks and defenses, 2016.

[15] Zhongliu Zhuo, Yang Zhang, Zhi-li Zhang, Xiaosong Zhang, and Jingzhong Zhang. Website fingerprinting attack on anonymity networks based on profile hidden markov model. *Trans. Info. For. Sec.*, 13(5):1081?1095, May 2018.

[16] Marc Juarez, Sadia Afroz, Gunes Acar, Claudia Diaz, and Rachel Greenstadt. A critical evaluation of website fingerprinting attacks. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, pages 263–274, 2014.

[17] Andriy Panchenko, Lukas Niessen, Andreas Zinnen, and Thomas Engel. Website fingerprinting in onion routing based anonymization networks. In *Proceedings of the 10th annual ACM workshop on Privacy in the electronic society*, pages 103–114, 2011.

[18] Antonio Pescape, Antonio Montieri, Giuseppe Aceto, and Domenico Ciuonzo. Anonymity services tor, i2p, jondonym: Classifying in the dark (web). *IEEE Transactions on Dependable and Secure Computing*, 2018.

[19] Junhua Yan and Jasleen Kaur. Feature selection for website fingerprinting. *Proceedings on Privacy Enhancing Technologies*, 2018(4):200–219, 2018.

[20] Tao Wang. High precision open-world website fingerprinting. In *2020 IEEE Symposium on Security and Privacy (SP)*, pages 152–167. IEEE, 2020.

[21] Xinhao Deng, Qilei Yin, Zhuotao Liu, Xiyuan Zhao, Qi Li, Mingwei Xu, Ke Xu, and Jianping Wu. Robust multi-tab website fingerprinting attacks in the wild. In *2023 IEEE Symposium on Security and Privacy (SP)*, pages 1005–1022. IEEE, 2023.

[22] Payap Sirinam, Nate Mathews, Mohammad Saidur Rahman, and Matthew Wright. Triplet fingerprinting: More practical and portable website fingerprinting with n-shot learning. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, pages 1131–1148, 2019.

[23] Rob Jansen, Ryan Wails, and Aaron Johnson. A measurement of genuine tor traces for realistic website fingerprinting. *arXiv preprint arXiv:2404.07892*, 2024.

[24] Mohamad Amar Irsyad Mohd Aminuddin, Zarul Fitri Zaaba, Azman Samsudin, Faiz Zaki, and Nor Badrul Anuar. The rise of website fingerprinting on tor: Analysis on techniques and assumptions. *Journal of Network and Computer Applications*, 212:103582, 2023.