POSTER: Video Fingerprinting in Tor

Mohammad Saidur Rahman
Center for Cybersecurity
Rochester Institute of Technology
saidur.rahman@mail.rit.edu

Nate Mathews
Center for Cybersecurity
Rochester Institute of Technology
nate.mathews@mail.rit.edu

Matthew Wright
Center for Cybersecurity
Rochester Institute of Technology
matthew.wright@rit.edu

ABSTRACT

Over 8 million users rely on the Tor network each day to protect their anonymity online. Unfortunately, Tor has been shown to be vulnerable to the website fingerprinting attack, which allows an attacker to deduce the website a user is visiting based on patterns in their traffic. The state-of-the-art attacks leverage deep learning to achieve high classification accuracy using raw packet information. Work thus far, however, has examined only one type of media delivered over the Tor network: web pages, and mostly just home pages of sites. In this work, we instead investigate the fingerprintability of video content served over Tor. We collected a large new dataset of network traces for 50 YouTube videos of similar length. Our preliminary experiments utilizing a convolutional neural network model proposed in prior works has yielded promising classification results, achieving up to 55% accuracy. This shows the potential to unmask the individual videos that users are viewing over Tor, creating further privacy challenges to consider when defending against website fingerprinting attacks.

CCS CONCEPTS

• **Security and privacy → Pseudonymity, anonymity and untraceability**; **Privacy-preserving protocols**; **Network security**; **Security protocols**; **Privacy protections**; • **Networks → Network privacy and anonymity**; • **Computing methodologies → Neural networks**; **Deep belief networks**; *Machine learning algorithms*;

KEYWORDS

Anonymity System; Privacy; Attack; Video Fingerprinting; Deep Learning;

ACM Reference Format:
Mohammad Saidur Rahman, Nate Mathews, and Matthew Wright. 2019. POSTER: Video Fingerprinting in Tor. In *2019 ACM SIGSAC Conference on Computer and Communications Security (CCS '19), November 11–15, 2019, London, United Kingdom.* ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/3319535.3363273

1 INTRODUCTION

Tor is a widely used anonymity system with over 8 million users each day [8]. Tor protects users' privacy by passing user traffic via encrypted circuits flowing through three nodes: entry, middle, and exit (see Figure 1). The design of Tor circuits hinders an attacker from associating the client's identity with the websites she visits, as no single point along the circuit is able to know both the client's identity and traffic's destination. Tor is, however, vulnerable to a class of traffic analysis attack known as website fingerprinting (WF) [4, 9–11, 13, 16]. A WF attack enables a local passive eavesdropper to deduce the website that a Tor client visits. The attacker collects encrypted network traffic flowing between the client and the entry node, where the eavesdropping could occur at the victim's ISP, her compromised home router, or over her home WiFi connection. Then the attacker extracts different features from the traffic such as traffic transmission time, bursts of traffic, and other packet statistics. The attacker can train a a machine learning (ML) or deep learning (DL) classifier on these features so it will reliably recognize sites of interest. Recent research shows that DL-based classifiers that operate on raw packet information achieve the best results, with less than 2% error in a closed-world setting [4, 16].

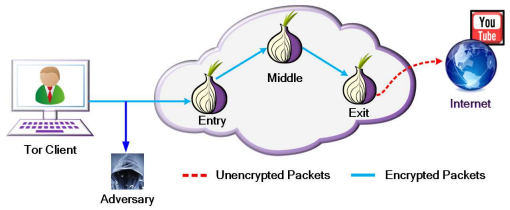

Figure 1: Video fingerprinting attack model.

Prior work in WF, however, examines only the loading of a web page, and most datasets for Tor use only the home page of each site of interest. Web pages are not the only type of content that Internet users commonly consume. Video streaming in particular has grown tremendously. Sandvine reports that video streaming is responsible for 57% of global downstream traffic, with 15% and 11% claimed by Netflix and YouTube, respectively [14]. Further, even though Tor is slower than regular browsing, it is often fast enough for video streaming, even in high definition. WF attacks on Tor, as currently tested, would only reveal that a user is visiting a video hosting site, such as YouTube. With video fingerprinting, however, the attacker could learn more specifically what the user is watching when they visit YouTube, making it potentially much more privacy invasive. Videos are available on numerous controversial topics, such as politics, race, religion, conspiracy theories, sexual orientation, and much more. As such, it is interesting to investigate how resistant Tor is to the fingerprinting of video content. To evaluate this, we collected a new Tor video traffic dataset and perform video fingerprinting (VF) experiments in this study.

Several prior works [5, 7, 12, 15] have examined the fingerprintability of video streams under typical browsing conditions (HTTPS).

CCS '19, November 11–15, 2019, London, United Kingdom

ACM ISBN 978-1-4503-6747-9/19/11.
https://doi.org/10.1145/3319535.3363273

To the best of our knowledge, however, we are the first to study video stream identification when protected by a privacy-enhancing technology such as Tor.

In this preliminary work, we explore the performance of conventional WF attacks when applied to the VF domain. For this, we focus on the most recent WF attacks which leverage DL models to classify using the raw packet sequence of traffic instances. For our experiments, we examine data representations that contain only packet direction information (as seen in [3, 13, 16]) and representations that include additional packet timing (as seen in [4, 11]). We have performed several experiments in which length of the feature vector (eg. packet sequence length) is varied. In our experiments, we achieve up to 54% (direction only) and 55% (direction and timing) when tested against a closed-world dataset containing 50 different videos. While these results are less than those reported for website fingerprinting, we note that they are for models and data representations that are not tailored to the VF problem. Further, even at these accuracy levels, they represent a potential privacy threat to Tor users that is not being considered in the discussion of developing WF defenses.

2 EXPERIMENTAL SETTING

2.1 Data Collection

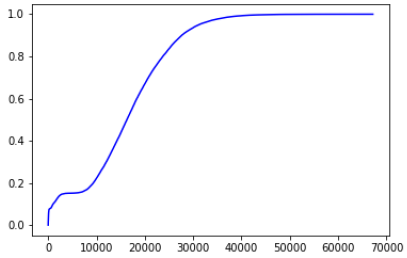

Figure 2: CDF of the number of packets in each instance of the dataset.

To perform our evaluation, we needed to collect a new dataset, as prior Tor fingerprinting works have exclusively examined web page traffic. For our new dataset, we collected traffic generated by loading popular music videos from the video sharing behemoth youtube.com. We selected YouTube for our study due to its online popularity, which makes it a likely target for attackers. To perform our video-over-Tor traffic crawl, we modified the `tor-browser-crawler` [1] to take YouTube video IDs in place of website URLs. We use *Selenium* [2] and the YouTube Web API to detect when a video has finished playing, ending the video capture instance.

A distinctive challenge in regards to fingerprinting YouTube videos is the presence of video advertisements. For any monetized video, YouTube may play non-skippable advertisements of up to 20 seconds in length or skippable advertisements of up to 3 minutes in length. These advertisements may appear at the start, end, or during playback of a video. To best emulate real user behavior, we simulate a user clicking the skip advertisement button when it becomes available for applicable advertisements.

Table 1: Closed world dataset

#of Videos	#of Instances	Total
50	640	32000

We collected approximately 50,000 total instances for 50 popular music videos using version 8.0.2 of the Tor Browser Bundle. We sampled our video list using trending videos from YouTube's *Music* category during the month of June. We restricted our video list to only videos that are approximately 3 minutes in length so that the videos were not trivially distinguishable by their load times alone. We allow YouTube to automatically select the best resolution for the stream during playback as this default is likely to be the behavior that most users choose.

Due to regional restrictions, many video loading attempts were blocked. To address this issue, we filtered out captures containing fewer than 3000 packets. We used manual inspection of blocked video traces and the trace length CDF graph (see Figure 2) to select this threshold. After filtering, we reduced the number of instances per class to the smallest instance count for any class. As stated in Table 1, we found this number to be 640 instances.

2.2 Data Representation

We used two data representations in our experiments. The first data representation, which is adopted in most DL-based WF attacks [3, 4, 13, 16], represent traces as a sequence of packet directions as a 1-D vector in which +1 and -1 denote the outgoing and incoming packets, respectively. The size of each packet is ignored, as Tor transmits data in fixed-length cells, making this information is uninteresting. Our second data representation captures both direction and timing information, following the work of Rahman et al. [11]. In particular, they propose a new data representation called Tik-Tok, in which each packet in the trace is represented by its direction (+1, -1) multiplied by its timestamp.

2.3 Model Selection

For the preliminary investigation, we adopted the convolutional neural network model used in the Deep Fingerprinting (DF) attack [16] as our base, and we have tuned this model for VF. The basic DF model contains four blocks, each of which contains two convolutional layers with batch normalization, a max pooling layer, and a drop-out layer. For our tuned model, we added one additional block. We use the same dropout rate for the blocks as the original model (0.1) to reduce overfitting. We set the dropout rate for each of the two fully-connected layers to 0.7, and we adopt rectified linear unit (ReLU) as the activation function for all convolutional and FC layers. In total, our tuned model has 13 layers when the classification layer is included.

3 EVALUATION

We ran five trials for both the direction and Tik-Tok data representations, in which we varied the length of the traces when fed into the classifier. The results of these experiments are shown in Table 2.

Table 2: Closed-World: Attack accuracy with different length of features.

Length of Features	Traffic Information	
	Direction [16]	**Tik-Tok** [11]
5000	28.60	34.10
10000	47.80	47.90
20000	52.20	52.60
30000	53.70	54.10
40000	54.30	54.70

In all of our experiments, we see slightly better accuracy with the Tik-Tok data representation. We achieve our highest accuracy when using a 40,000-length input vector. This is not surprising, since 99% of traces have packet sequence lengths below this threshold, so there is no information loss for most traces. We notice, however, that the accuracy gain over 20,000 and 30,000 length vectors is relatively minimal, despite capturing the full length of only approximately 75% of traces at 20,000 packets. This may indicate that either the later portions of a trace are less valuable for classification, or it may be that classification of larger traces performs poorly, so reducing the information available in their traces makes little difference in the overall accuracy.

All things considered, the performance of the DF model on our VF dataset is interesting, but likely inadequate for attacks in a realistic setting. This is most likely due to the differences between typical WF traffic and VF traffic. In a standard WF scenario, the tested webpages are mostly static within a single dataset. This favors fingerprintability, as the traffic patterns contained by a page are more easily identifiable. On the other hand, video streaming traffic is dynamic due to the use of Dynamic Adaptive Streaming over HTTP (DASH) [6]. The DASH protocol works by dividing a video into small time segments. These time segments are served to the client with variable encoding bitrates based on the available bandwidth on the connection at the time of sending. This results in some trace samples for the same video that have many more packets than others. Because the DF attack does not perform any additional data processing on the raw packet sequences, this variance causes a major issue, as it results in sequences that appear significantly different within the same class. This type of behavior does not occur in the WF domain, as repeated visits to the same site reliably result in similar length packet sequences. It is thus unlikely that our current representation of the data will be adequate for the VF attack domain.

4 CONCLUSION & FUTURE WORK

In this work, we investigated the fingerprintability of video traffic protected by Tor. We collected a large dataset of traffic traces from 50 YouTube videos using the Tor Browser Bundle, YouTube API, and Selenium. In our experiments, we considered two types of data representation: packet direction only and Tik-Tok (direction and time). We adopted the Deep Fingerprinting attack, which performs very well against web pages, and tuned it for our dataset. Despite not addressing the significant differences between video data and web pages, we still manage to get nearly 55% accuracy. This shows the potential for stronger attacks to more seriously unmask users' video viewing habits despite the use of Tor.

In future investigations, we will develop different processing techniques to normalize the appearance of video traffic captured at different bitrates, as well as different deep learning models to handle this data more effectively. If those attempts to improve the attack prove successful, we will need to capture an additional dataset to evaluate the performance of VF in the more realistic open-world setting. We note that our dataset uses videos with a limited range of viewing lengths. Because significantly longer and shorter videos are easily removed from contention as candidate classes, large portions of the open world could be culled in practice, which is an interesting feature of the VF problem. Furthermore, it will be interesting to determine the effectiveness of existing padding defenses for WF in the VF domain. We note that video length is a very distinguishing characteristic that is hard to hide without paying large bandwidth overheads for padding, so we expect that this will be a major problem for these defenses to contend with.

ACKNOWLEDGMENTS

This material is based upon work supported by the National Science Foundation under Awards No. 1722743, 1816851, and 1433736.

REFERENCES

[1] 2018. Tor Browser Crawler. https://github.com/webfp/tor-browser-crawler.
[2] 2019. Selenium. https://github.com/SeleniumHQ/selenium.
[3] Kota Abe and Shigeki Goto. 2016. Fingerprinting attack on Tor anonymity using deep learning. *Proceedings of the Asia-Pacific Advanced Network* (2016).
[4] Sanjit Bhat, David Lu, Albert Kwon, and Srinivas Devadas. 2019. Var-CNN: A data-efficient website fingerprinting attack based on deep learning. *Proceedings on Privacy Enhancing Technologies* 2019, 4 (2019), 292–310.
[5] Jiaxi Gu, Jiliang Wang, Zhiwen Yu, and Kele Shen. 2018. Walls have ears: Traffic-based side-channel attack in video streaming. In *IEEE INFOCOM*.
[6] ISO 23009-1:2019(en) 2019. *Information technology — Dynamic adaptive streaming over HTTP (DASH)*. Standard.
[7] Ying Li, Yi Huang, Suranga Seneviratne, Kanchana Thilakarathna, Adriel Cheng, Guillaume Jourjon, Daren Webb, and Richard Xu. 2018. Deep Content: Unveiling video streaming content from encrypted WiFi traffic. In *2018 IEEE 17th International Symposium on Network Computing and Applications (NCA)*. 1–8.
[8] Akshaya Mani, T Wilson-Brown, Rob Jansen, Aaron Johnson, and Micah Sherr. 2018. Understanding Tor usage with privacy-preserving measurement. In *Proceedings of the Internet Measurement Conference*. ACM.
[9] Andriy Panchenko, Fabian Lanze, Jan Pennekamp, Thomas Engel, Andreas Zinnen, Martin Henze, and Klaus Wehrle. 2016. Website fingerprinting at Internet scale. In *Proceedings of the 23rd Network and Distributed System Security Symposium (NDSS)*.
[10] Mike Perry. 2013. A critique of website traffic fingerprinting attacks. *Tor project blog*. (2013). https://blog.torproject.org.
[11] Mohammad Saidur Rahman, Payap Sirinam, Nate Matthews, Kantha Girish Gangadhara, and Matthew Wright. 2019. Tik-Tok: The utility of packet timing in website fingerprinting attacks. *arXiv preprint arXiv:1902.06421* (2019).
[12] Andrew Reed and Benjamin Klimkowski. 2016. Leaky streams: Identifying variable bitrate DASH videos streamed over encrypted 802.11n connections. In *IEEE Annual Consumer Communications Networking Conference (CCNC)*. 1107–1112. https://doi.org/10.1109/CCNC.2016.7444944
[13] Vera Rimmer, Davy Preuveneers, Marc Juarez, Tom Van Goethem, and Wouter Joosen. 2018. Automated website fingerprinting through deep learning. In *Proceedings of the 25th Network and Distributed System Security Symposium (NDSS)*.
[14] Sandvine. 2018. The Global Internet Phenomena Report October 2018. https://www.sandvine.com/hubfs/downloads/phenomena/2018-phenomena-report.pdf.
[15] Roei Schuster, Vitaly Shmatikov, and Eran Tromer. 2017. Beauty and the Burst: Remote identification of encrypted video streams. In *Proceedings of the 26th USENIX Conference on Security Symposium*.
[16] Payap Sirinam, Mohsen Imani, Marc Juarez, and Matthew Wright. 2018. Deep Fingerprinting: Undermining website fingerprinting defenses with deep learning. In *Proceedings of the 2018 ACM Conference on Computer and Communications Security (CCS)*. ACM.