Dynamic Traffic Feature Camouflaging via Generative Adversarial Networks

Jie Li, Lu Zhou, Huaxin Li, Lu Yan, Haojin Zhu
Department of Computer Science and Engineering, Shanghai Jiao Tong University

***Abstract*—Traffic analysis attacks, including website fingerprinting and protocol fingerprinting, are widely adopted by Internet censorship to block a specific type of traffic. To mitigate these attacks, some advanced approaches such as traffic morphing and protocol tunneling techniques have been proposed. However, the existing traffic morphing/protocol tunneling techniques suffer from showing a strong traffic pattern or can be uncovered with a low false positive. Further, they mainly rely on learning the pattern for specific traffic, which makes it highly possible to be identified due to a lack of dynamics. In this paper, we propose a dynamic traffic camouflaging technique, coined FlowGAN, to dynamically morph traffic feature as another "normal" network flow to bypass Internet censorship. The core idea of FlowGAN is to automatically learn the features of the "normal" network flow, and dynamically morph the on-going traffic flows based on the learned features by the adoption of the recently proposed Generative Adversarial Networks (GAN) model. To measure the indistinguishability of the target traffic and the morphed traffic, we introduce a novel concept of ϵ-indistinguishability. We evaluate the proposed method on a dataset involving 10,000 real-world flows, and experimental results show that the effectiveness and the efficiency of FlowGAN regarding ϵ-indistinguishability, AUC, and latency. To the best of our knowledge, our work is the first one to adopt GAN for automatic traffic generation and censor circumvention.**

I. Introduction

Internet censorship circumvention is regarded as one of the key applications of anonymous communication techniques [1]. Since the network protocols and connection endpoints adopted by circumvention systems are quite different from those of other Internet services, it is easy for ISP-level censorship to perform the traffic analysis attacks and block them at a low cost without compromising any other network services. The typical traffic analysis attacks include *website fingerprinting* (WF) attacks [2], [3], [4], [5], which aim to infer the web activity of a client (e.g., which web page is visited) even when the client is using the anonymous tool, and *protocol fingerprinting* (PF) attacks [6], which analyze the traffic pattern to measure the similarity between an unknown flow and a specific protocol. In general, traffic analysis attacks are modeled as the classification problem, where the attackers extract features from traffic flows they captured and employ classifiers such as k-Nearest Neighbors (kNN) or deep neural networks (DNN) to fingerprint these flows.

To mitigate traffic analysis attacks, different kinds of defense techniques have been proposed in recent years. A common approach of preventing leaks is to obfuscate the encrypted traffic by changing the statistical features of the traffic, such as the packet size and packet timing information [3], [7], [8]. More advanced approaches include Traffic Morphing, which can optimally morph one class of traffic to look like another class [9], and the protocol tunneling technique, which leverages the popular encrypted online services (e.g., living streaming) to transmit censored content [10]. However, the existing traffic morphing/protocol tunneling techniques suffer from showing a strong traffic pattern or can be uncovered with a low false positive rate [2], [5]. Further, the existing solutions heavily rely on the learning of the traffic pattern of a specific kind of flow, which makes it highly possible to be recognized and blocked by the censors. A good example is CovertCast, which uses Youtube live streaming to transmit the covert channel. However, Youtube has been blocked by more than 10 countries/areas according to [11], which makes the covert channel fail to work either. Therefore, a more dynamic traffic camouflaging technique is highly expected.

In this study, we propose a novel traffic camouflaging technique, coined FlowGAN, which allows a proxy to dynamically morph on-going traffic flows as some other flows. As shown in Fig. 1, by giving a source (or censored) flow and a target (or permitted) flow, the core idea of FlowGAN is to automatically extract the traffic features of the target flow, and morph the source flow to the target flow based on the extracted traffic features. The generated flow should be indistinguishable from the target flow, which can provide the privacy guarantee and censorship circumvention. All of these can be achieved by the adoption of the recently proposed Generative Adversarial Nets (GAN) model. GAN's discriminator tries to classify if the output flow is censored or the permitted, while simultaneously training a generative model to bypass the discriminator's detection. Since GANs learn the pattern that adapts to the data, they can be applied to a multitude of tasks (or multiple target permitted flows) that traditionally would require very different kinds of loss functions. Compared with any previous works, the proposed approach shows its superiority in terms of dynamical traffic camouflaging, which is expected to morph a censored flow to multiple target flows. This is expected to prevent the censors from recognizing and blocking any specific type of traffic. To validate the effectiveness of FlowGan, we have designed a prototype. The evaluation is performed on a dataset involving 10,000 traffic flows to *baidu.com*. We give a formal definition on indistinguishability of two traffic flows and evaluate the effectiveness and efficiency of FlowGan, in terms of a series of metrics including Indistinguishability,

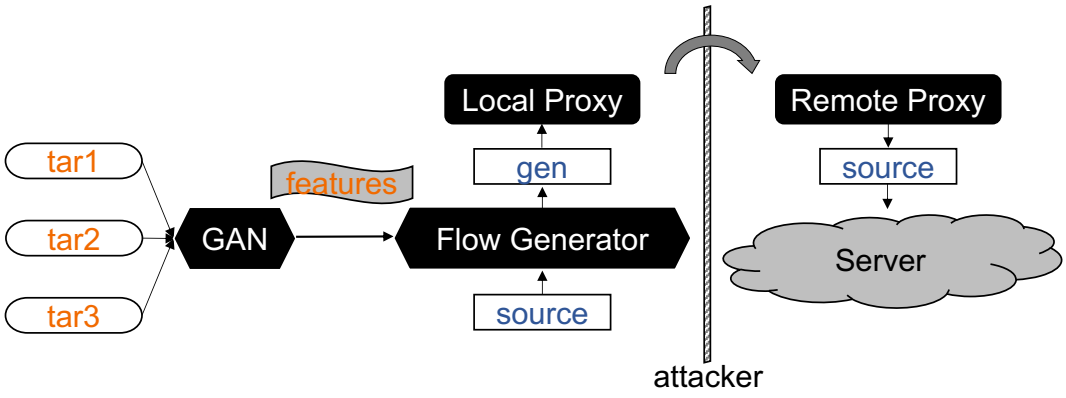

Fig. 1. Overview of FlowGAN

AUC, Latency. The evaluation results well demonstrate that FlowGan outperforms the counterparts and is expected to significantly mitigate existing traffic analysis attacks.

The main contributions of our paper are as follows:

- We proposed a new traffic camouflaging mechanism, FlowGan, to secure anonymous network communications against both website fingerprinting attacks and protocol fingerprinting attacks. FlowGan employs GAN models to generate network flows that are indistinguishable from given traffic flows so that the censor cannot block them.
- We proposed a novel privacy model to evaluate the effectiveness of the proposed work. This new private model measures the indistinguishability between the GAN-generated network flows and the targeted network flows.
- We performed a series of state-of-the-art traffic analysis attacks towards FlowGan based on a real-world dataset. The evaluation results well demonstrated the effectiveness and the efficiency of FlowGAN under a series of state-of-the-art traffic analysis attacks.

To the best of our knowledge, this is the first work which leverages GAN for traffic camouflaging and anonymous communications. The remainder of the paper is structured as follows. Section II surveys the representative and latest related works of traffic analysis attacks and defenses. Section III presents the threat model in our paper. Section IV describes our private model and the evaluation metric. Section V introduces the methodology and implementation of our system. Section VI elaborates experiments and evaluation results. Section VII concludes this paper.

II. Related Works

Traffic analysis can be used to conduct side-channel attack in many scenarios [12], [13], [14], [15]. One of the most common cases is the fingerprinting attack through network traffic analysis, including website fingerprinting and protocol fingerprinting. Recent research works have shown that state-of-art anonymous mechanisms are still vulnerable under these attacks and thus new defense mechanisms are proposed.

A. Traffic Fingerprinting Attacks

Fingerprinting attacks are usually modeled as a classification problem in which attackers extract features from side channel and classify a set of websites/apps [4], [16]. Traffic fingerprinting attacks have been being extensively studied over the past ten years [17]. Dyer et al. [3] provided a first comprehensive study on traffic analysis attacks and defenses in 2012, and their results showed that many of the countermeasures by that time could not resist even simple attacks that leverage general traffic features such as bandwidth and total time.

Cai et al. [18] proposed to compare the similarity of packet traces (e.g., packet orderings) for identifying web pages, and then used Hidden Markov Model (HMM) to fingerprint websites based on the sequence of visited web page under a simulated Tor environment. Wang et al. [5] built a k-Nearest-Neighbor (kNN) classifier using a large set of features. Their classifier has been tested on a large open-world setting and outperformed some previous works. In [19], [20], the authors proposed that side channel data is efficient to uncover enough information. Hayes et al. [21] proposed an attack that employed Random Forest model to produce fingerprints of websites, and launched website fingerprinting attack with a large amount of noisy data as well as Tor environment. In 2016, Panchenko et al. [4] used a SVM classifier to classify the cumulative features (CUMUL) of different flows. Meanwhile, they evaluated fingerprinting at the Internet scale with representative data. However, their experiments showed no existing method can scale for any web page in their datasets.

In the recent two years, approaches that exploited more advanced models, such as deep learning model (supervised model [22], unsupervised model [23]), were studied. Moreover, the fingerprintability [24] was measured and analyzed. Though secure channels and encryption have been widely deployed in network communication, traffic features in side-channel and usage patterns inevitably provide evidence to attackers. Along with larger datasets being evaluated and more powerful attacks being proposed, the implications of traffic fingerprinting attacks are alarming and attract research interests increasingly.

B. Defenses for Fingerprinting Attacks

Various defenses were proposed to counter fingerprinting attacks. Since many of the attacks extracted traffic features to discriminate different websites, a typical category of countermeasures is to conceal the traffic features. Padding and creating dummy traffic are common ways to modify traffic features such as packet size and time interval. For instances, HTTPOS [7] can hide packet length by strategically sending HTTP request, while TrafficMorphing [9], WTF-PAD [25], BuFLO [3], CS-BuFLO [26], and Tamaraw [8] padding traffic packets in different strategies to hide the actual time interval and length. However, many of them have been proved to be ineffective under newly proposed attacks or cause a large overhead [2], [5].

With the urgent requirements for defending fingerprinting attacks, more effective and efficient countermeasures that consider the open-world scenario and overhead were proposed in recent years. To consider the practical environment, Cherubin et al. [27] proposed a server-side defense a lightweight client-side defense implemented as a browser add-on to defend WF attacks at the application level. Walkie-Talkie [2] modified the browser communication from usual full-duplex mode to in half-duplex mode so that the traffic is sent in moldable interleaving bursts and the molding is computationally cheap. Cui et al. [28] used variable realistic network traffic as noise based on the motivation that visiting two websites at once mitigates website fingerprinting attacks. Our solution considers a practical scenario that censors that identify suspicious traffic patterns [29] may block the morphed traffic causing the countermeasure failed. So our goal is to generate traffic that is hard to be fingerprinted and can circumvent Internet censorship.

III. Threat Model

By following the similar threat model as shown in [30], in this study, we consider the attacker model as a state-level adversary, which attempts to classify the censored traffic from normal traffic by leveraging a series of website fingerprint techniques. The censored traffic is expected to take advantage of protocol tunneling techniques to evade the detection of the adversary by using the encrypted applications as the carriers for covert channels. The adversary is not assumed to demand the application providers to decipher and disclose the original content, which enables the easy identification and filtering of the covert data. However, the censorship system can be deployed by domestic ISPs with cooperation with the adversary, which allows it to monitor, store and inspect all traffic flows crossing its borders [30].

The adversary is assumed to perform the various traffic analysis towards some suspicious flow and block the traffic containing the covert channels. In practice, morphing the censored traffic to a specific kind of popular traffic (e.g., livestreaming services such as youtube) is quite risky since this popular traffic may be unavailable in some regions. For example, youtube has been blocked by more than 10 countries including Iran, Afghanistan, Armenia, Brazil, Finland, Germany [11]. Therefore, a more dynamic traffic obfuscation solution which can dynamically transform the censored traffic to other popular applications is highly desirable.

TABLE I
Notation

Symbol	Description
Adv	Adversary
Chl	Challenger
Tar	Traffic from morphing target
Gen	Generated traffic
P	Adversary's prior knowledge
H_P	Adv's classifier trained with P.
$P_T = P \setminus Tar$	Adv's prior knowledge about T
$P_G = P \setminus Gen$	Adv's prior knowledge about G
$C_1 \ _R T$	C_1 is a random subset of Tar
$C_0 \ _R G$	C_0 is a random subset of Gen

Challenger C		**Adversary A**
Tar *initialization*		
$Gen \quad GAN(Tar)$		$Hp \quad P$
$S_0 \quad Tar, S_1 \quad Gen$		
$b \ \$ \ f0, 1g$		
$c \quad S_b$	$\xrightarrow{c}$	$\hat{b} \quad A(c)$
$PrivK = 1 \ ( ) \ \hat{b} = b$		

Fig. 2. Traffic Indistinguishability Game

IV. Traffic Indistinguishability Definition

In this section, we formalize traffic indistinguishability under the traffic analysis attack in our problem. We consider the scenario that an attacker attempts to infer if an encrypted traffic flow contains the covert channel via performing the traffic analysis and block a specific traffic flow based on the inference results. Our goal is to generate a traffic flow which is indistinguishable from another "normal" network flow.

A. Notation

The notations in this paper are summarized in Table I. We denote the set of target traffic flows as $Tar = fT_1; T_2; :::; T_n g$ and the set of generated traffic flows which are generated based on the target traffic as $Gen = fG_1; G_2; :::; G_n g$, where T_i and G_j represent a traffic flow of Tar and Gen, respectively. For an adversary (Adv), we denote his/her prior knowledge as $P = P_T \ [\ P_G$ where P_T and P_G are prior knowledge about Tar and Gen, respectively. P_T and P_G are supposed to be known by the adversary and used to train the classifier to classify Tar and Gen.

B. Traffic Indistinguishability Game

We model our problem by a traffic indistinguishability game involving an adversary Adv and a challenger Chl. Traffic indistinguishability game can be described as follows (Fig. 2):