Extended Abstract: Traffic Shaping for Network Protocols: A Modular and Developer-Friendly Framework

Hugo Santos Pereira
Universidade NOVA de Lisboa & NOVA LINCS
Lisbon, Portugal
hg.pereira@campus.fct.unl.pt

Afonso Vilalonga
Universidade NOVA de Lisboa & NOVA LINCS
Lisbon, Portugal
j.vilalonga@campus.fct.unl.pt

Kevin Gallagher
Universidade NOVA de Lisboa & NOVA LINCS
Lisbon, Portugal
k.gallagher@fct.unl.pt

Henrique Domingos
Universidade NOVA de Lisboa & NOVA LINCS
Lisbon, Portugal
hj@fct.unl.pt

Abstract

Censorship-resistant systems and privacy-preserving communication tools are increasingly vulnerable to detection by adversaries using deep packet inspection (DPI) and traffic analysis. While encryption ensures the confidentiality of packet payloads, metadata, such as packet sizes, burst patterns, and timing characteristics, remain exposed and can be exploited to fingerprint and block these tools or deanonymize their endpoints. Both historical evidence of real-world censorship techniques and research-based approaches have demonstrated the vulnerability of these systems to attacks that exploit packet metadata. However, in many of these examples, we observe that typically, the initial seconds of communication between the user and the system's proxy are sufficient to carry out the attacks. In this work, we present the design of a modular framework for shaping the initial seconds of a user-proxy connection aimed at mitigating the above-described vulnerabilities with minimal performance overhead. Central to our framework are two components: a scheduler, which intercepts and shapes packets exchanged between the user and the system's proxy, and a shaper policy, which defines how the scheduler shapes the exchanged traffic. We plan to base our shaping policies on two main approaches: (1) predefined or user-configurable schedules and (2) traffic patterns generated by a generative adversarial network (GAN) designed to mimic realistic behavior. By targeting the initial communication phase, where many classifiers extract highly discriminative features, we hypothesize that we can provide robust protection against DPI and traffic analysis attacks that threaten real-world systems designed to evade censorship or provide user privacy.

Keywords

Censorship Resistance, Traffic Shaping, Traffic Analysis, GANs

1 Introduction & Background

As online censorship and surveillance become increasingly sophisticated and pervasive [6], censorship evasion systems and privacy-preserving communication tools must evolve to counter adversaries relying on DPI and traffic analysis to fingerprint and block them and/or deanonymize the connection endpoints. Some of these attacks, particularly DPI, have been used in real-world scenarios against censorship evasion systems [2]. In contrast, others, such as traffic correlation and website fingerprinting, have mainly been speculated to be feasible for deployment in real-world scenarios [3, 10]. Regardless of their real-world application, both pose a threat because, even when payloads are encrypted and protocols are obfuscated, side-channel metadata, such as packet sizes, inter-arrival times, and burst patterns, can still reveal the nature of a connection [12, 13].

However, many of these real-world attacks have specifically targeted the initial moments of a connection, where traffic features are highly distinctive and information-dense [12, 14]. Additionally, research has also emphasized the significance of early-connection traffic, with recent studies on website fingerprinting demonstrating that classifiers trained solely on the first few seconds of traffic can achieve high accuracy [9, 11].

This work presents the design of a modular and extensible framework for early-session traffic shaping, designed to be adopted by censorship evasion systems, similar to how current systems leverage uTLS [5]. Our framework supports two types of shaping policies: (1) predefined or user-defined (rule-based) schedules and (2) GAN-generated (model-based) traffic patterns to mimic realistic behavior. These approaches will enable developers to simulate realistic patterns that resemble legitimate browsing behavior or other rule-based shaping methodologies with minimal bandwidth and latency overhead. We hypothesize that by enforcing these scheduling strategies only on the early traffic of the connection, while leaving the rest of the connection unchanged, we maintain the system's performance as close as possible to that of the system without our framework. At the same time, making the system resistant to attacks that rely on the initial seconds of the connection to either fingerprint or deanonymize the user or the content being accessed.

The remainder of this extended abstract is organized as follows: In Section 2, we discuss our planned system design, and in Section 3, we present our hypotheses on how the system may help mitigate traffic analysis and fingerprinting attacks.

Free and Open Communications on the Internet 2025(2), 40–42

2 System Design

We present the design of a modular, developer-oriented traffic-shaping framework designed to strengthen censorship circumvention and privacy-preserving communication tools against early-traffic fingerprinting and traffic analysis attacks. The system targets a key vulnerability in such systems: the highly fingerprintable initial seconds of a connection. By shaping traffic during this window, we hypothesize that it can resist the attacks introduced in Section 1 while incurring minimal overhead.

2.1 Architecture Overview

The system operates as a transparent shim between the *Application* and the *Network* stack. At its core is the *Traffic Shaper*, a component that enforces a configurable transmission schedule over packet size, timing, and burst patterns. The shaping logic is transport-agnostic, occurring after the handshake of the underlying protocol (e.g., TCP's 3-way handshake), and can be applied across various protocols, including TCP, UDP, QUIC, and TLS.

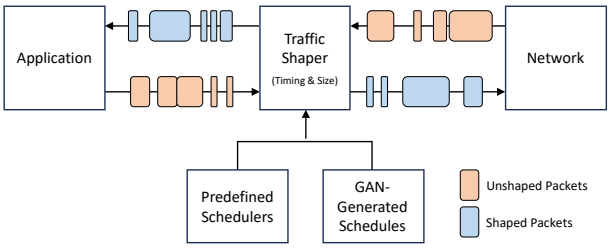

Figure 1: System architecture: the *Traffic Shaper* enforces a configurable schedule between the *Application* and the *Network* layer. Only one endpoint of the connection is shown.

As illustrated in Figure 1, traffic is first intercepted and buffered, then segmented, and finally released based on the shaping schedule that specifies both packet size and inter-arrival timing. This processing pipeline can be applied independently to either one side of the connection or both sides (i.e., the proxy and the client).

2.2 Shaping Schedules

We plan to generate shaping strategies for the framework using two approaches:

- **Predefined or User-defined Schedules:** A set of hand-crafted shaping strategies [1], enabling developers to stress-test shaping logic and quickly prototype defenses. These schedules mimic common traffic patterns, including constant-rate transmissions, server-driven activity with variable pacing, and multi-process background behavior, such as periodic pings.

- **GAN-Generated Schedules:** We also plan to leverage a generative adversarial network (GAN) trained on traffic traces (e.g., the Tranco Top 1000 sites [8]) to synthesize realistic, web-like traffic flows. The model will be trained on short session windows (e.g., the first 10 seconds of traffic), learning the distributions and temporal correlations of features, namely packet sizes, inter-arrival times, and burst patterns, to generate realistic and nuanced traffic schedules. While prior work, such as GANDaLF [7], uses GANs to enhance traffic analysis attacks, our approach leverages similar architectures to synthesize realistic traffic patterns from web access behavior for traffic scheduling policies. The model outputs timestamped sequences with specified packet sizes and inter-arrival timings, *(packet size, inter-arrival time)* tuples. The *Traffic Shaper* consumes these synthetic schedules in real-time, enabling the creation of simulated traffic patterns without requiring developers to manually design shaping strategies.

2.3 Deployment and Integration

We plan to implement the framework in Go to achieve high performance and enable cross-platform deployment, which makes it particularly suited for integration into systems like pluggable transports used by Tor bridges, lightweight VPNs (e.g., WireGuard [4]), or other censorship-resistant systems that lack built-in traffic obfuscation. The framework is designed to be lightweight and minimize the required code changes for adoption. It introduces only transient overhead, as shaping is confined to the early session window (e.g., the first N seconds, with N being a user-defined parameter). Although designed for early traffic shaping, N can be adjusted to extend shaping for longer periods or the entire session, depending on anticipated censor capabilities. Once the shaping phase is complete, normal traffic flow resumes, preserving the system's typical throughput and latency for the remainder of the connection.

3 Research Questions

We propose the following research questions concerning the effectiveness of early-session traffic shaping in mitigating traffic analysis and fingerprinting attacks:

RQ1: Does early-session traffic shaping mitigate existing fingerprints or introduce new identifiable patterns? We hypothesize that shaping traffic during the first few seconds of a connection will significantly reduce the effectiveness of traffic analysis tools in identifying and fingerprinting traffic. This early phase often provides sufficient information for classification, making it a prime target for fingerprinting. Early-session traffic has been extensively studied in academic literature and observed in real-world attacks [3]. To evaluate this, we will: (1) test the resistance against real-world censorship techniques [14], and its robustness against emerging deanonymization attacks, such as website fingerprinting and traffic correlation; and (2) assess whether the generated traffic is distinguishable from regular traffic, including analysis of any fingerprints potentially introduced by GAN-generated patterns.

RQ2: Will early-traffic shaping impact the systems performance? We anticipate that confining traffic shaping to the first N seconds of a connection (e.g., $N = 10$) will introduce only minimal overhead, resulting in negligible long-term impact on throughput and latency. This approach aims to preserve the performance characteristics of the underlying censorship evasion system for the remainder of the session. To evaluate this, we will measure key network performance metrics, including throughput and latency. We will compare these metrics against baseline unshaped traffic under varying network conditions and workloads to quantify any added overhead or degradation.

Acknowledgments

The authors would like to thank the anonymous reviewers for their constructive feedback. This work was supported by the FCT Ph.D. scholarship grant Ref. (PRT/BD/154787/2023), awarded by the CMU Portugal Affiliated Ph.D. program.

References

[1] 2023. Looking for: A good paper on how TLS-in-TLS detection works? https://github.com/net4people/bbs/issues/281. [Accessed 15-01-2025].

[2] Cecylia Bocovich, Arlo Breault, David Fifield, Serene, and Xiaokang Wang. 2024. Snowflake, a censorship circumvention system using temporary WebRTC proxies. In *33rd USENIX Security Symposium (USENIX Security 24)*. USENIX Association, Philadelphia, PA, 2635–2652. https://www.usenix.org/conference/usenixsecurity24/presentation/bocovich

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

[4] Jason A. Donenfeld. 2017. WireGuard: Next Generation Kernel Network Tunnel. In *Proceedings of the 2017 Network and Distributed System Security Symposium (NDSS)*. Internet Society, San Diego, CA, USA. https://www.ndss-symposium.org/ndss2017/ndss-2017-programme/wireguard-next-generation-kernel-network-tunnel/

[5] Sergey Frolov and Eric Wustrow. 2019. The use of TLS in Censorship Circumvention. In *Network and Distributed System Security*. The Internet Society.

[6] Alexander Master and Christina Garman. 2023. A worldwide view of nation-state internet censorship. *Free and Open Communications on the Internet* (2023).

[7] Se Eun Oh, Nate Mathews, Mohammad Saidur Rahman, Matthew Wright, and Nicholas Hopper. 2021. GANDaLF: GAN for data-limited fingerprinting. *Proceedings on privacy enhancing technologies* 2021, 2 (2021).

[8] Vincent Pochat, Tom Van Goethem, Samaneh Tajalizadehkhoob, Maciej Korczyński, and Wouter Joosen. 2019. Tranco: A Research-Oriented Top Sites Ranking Hardened Against Manipulation. In *Proceedings of the Network and Distributed System Security Symposium (NDSS)*.

[9] Vera Rimmer, Davy Preuveneers, Marc Juarez, Tom Van Goethem, and Wouter Joosen. 2018. Automated Website Fingerprinting through Deep Learning. In *Proceedings 2018 Network and Distributed System Security Symposium (NDSS 2018)*. Internet Society. https://doi.org/10.14722/ndss.2018.23105

[10] Vera Rimmer, Theodor Schnitzler, Tom Van Goethem, Abel Rodríguez Romero, Wouter Joosen, and Katharina Kohls. 2022. Trace Oddity: Methodologies for Data-Driven Traffic Analysis on Tor. *Proceedings on Privacy Enhancing Technologies* 3 (2022), 314–335.

[11] 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 SIGSAC conference on computer and communications security*. 1928–1943.

[12] Liang Wang, Kevin P. Dyer, Aditya Akella, Thomas Ristenpart, and Thomas Shrimpton. 2015. Seeing through Network-Protocol Obfuscation. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security* (Denver, Colorado, USA) *(CCS '15)*. Association for Computing Machinery, New York, NY, USA, 57–69. https://doi.org/10.1145/2810103.2813715

[13] Michael Wrana, Diogo Barradas, and N Asokan. 2025. SoK: The Spectre of Surveillance and Censorship in Future Internet Architectures. *Proceedings on Privacy Enhancing Technologies* (2025).

[14] Mingshi Wu, Jackson Sippe, Danesh Sivakumar, Jack Burg, Peter Anderson, Xiaokang Wang, Kevin Bock, Amir Houmansadr, Dave Levin, and Eric Wustrow. 2023. How the Great Firewall of China Detects and Blocks Fully Encrypted Traffic. In *USENIX Security Symposium*. USENIX. https://www.usenix.org/system/files/sec23fall-prepub-234-wu-mingshi.pdf