The Discriminative Power of Cross-layer RTTs in Fingerprinting Proxy Traffic

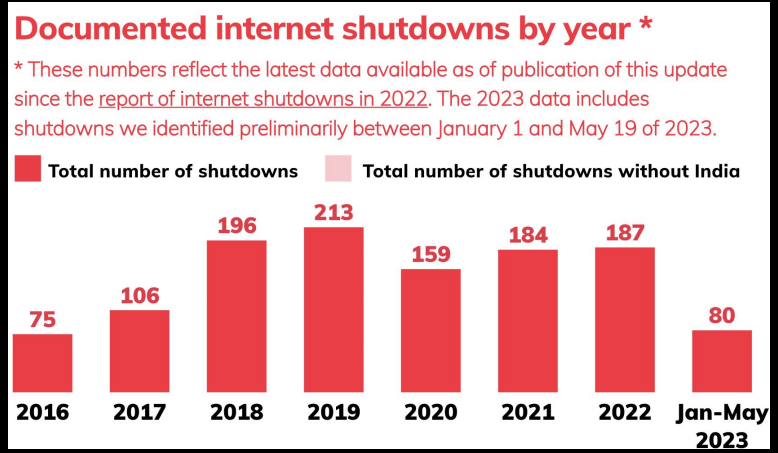
Diwen Xue, Robert Stanley, Piyush Kumar, Roya Ensafi
University of Michigan

Academics: Russia deployed new technology to throttle Twitter's traffic

Russia to spend over half a billion dollars to bolster internet censorship system

By **Gleb Stolyarov** and **Lucy Papachristou**

September 10, 2024 1:05 PM EDT · Updated 5 months ago



PETER GUEST BUSINESS 26.10.2023 01:42 PM

The UK's Controversial Online Safety Act Is Now Law

The UK government says its Online Safety Act will protect people, particularly children, on the internet. Critics say it's ineffective against dangerous misinformation and may be a threat to privacy.

EFF DONATE

The U.S. Government Wants To Control Online Speech to "Protect Kids"

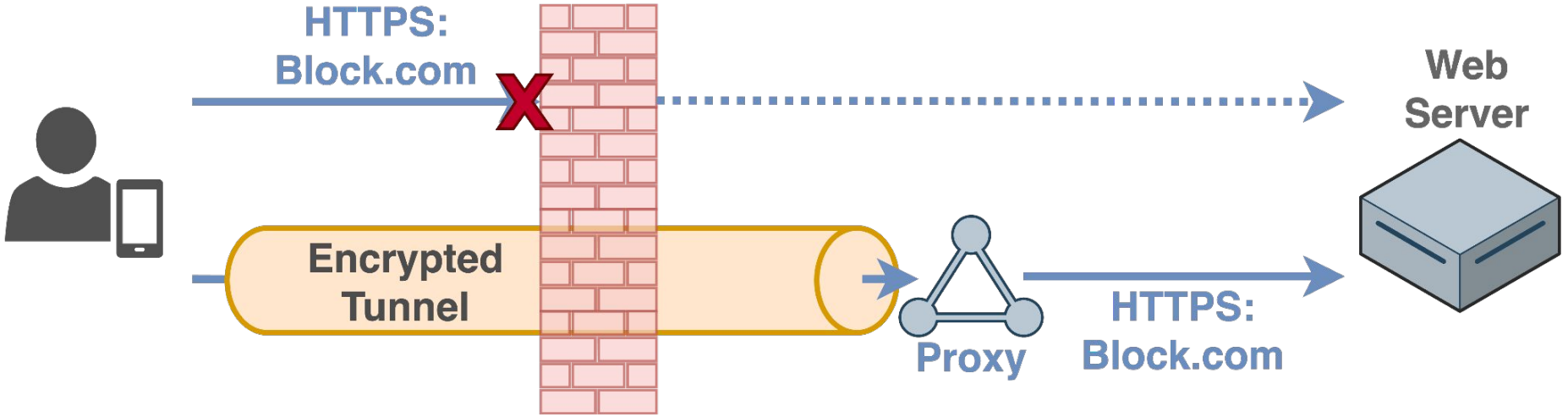
DEEPLINKS BLOG

Governor Abbott Announces Statewide Plan Banning Use Of TikTok

Austrian ISPs 'Had No Choice' But to Block Pirate Sites AND Cloudflare

Censored Planet

Circumvention Proxies



Censored Planet

Background: Obfuscated Proxies vs. Firewalls

A two-decade, adversarial arms race between tunneling tools and firewalls.

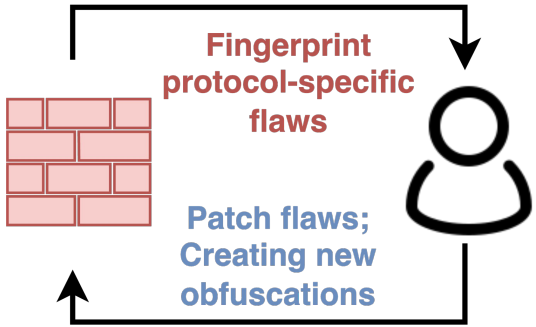
- **After blocking plain tunnels, go after “obfuscated” variants.**
- **An arms-race-driven evolution of obfuscation and detection methods.**

➔ TLS-based obfuscated proxies	➔ TLS fingerprinting attacks
➔ Fully-encrypted proxies	➔ Entropy-based traffic filter
➔ Probe-resistant proxies	➔ Active-probing fingerprinting
➔ Traffic shaping	➔ Traffic analysis
➔ …	➔ …

IRC Tip about Signature used to block Snowflake in Russia, 2022-May-16

Issue actions

Closed Issue created 2 years ago by **shelikhoo**



Make Snowflake's DTLS fingerprint more similar to popular WebRTC implementations

Issue actions

Open Issue created 4 years ago by **Cecylia Bocovich**

Prior Work:

- Target protocol-specific flaws
  - e.g., flawed TLS implementations
- Community Strategy: outpace firewalls that can't keep pace with every variant

IRC Tip about Signature used to block Snowflake in Russia, 2022-May-16

Issue actions

Closed

Make Snowflake's DTLS fingerprint more similar to popular WebRTC implementations

Issue actions

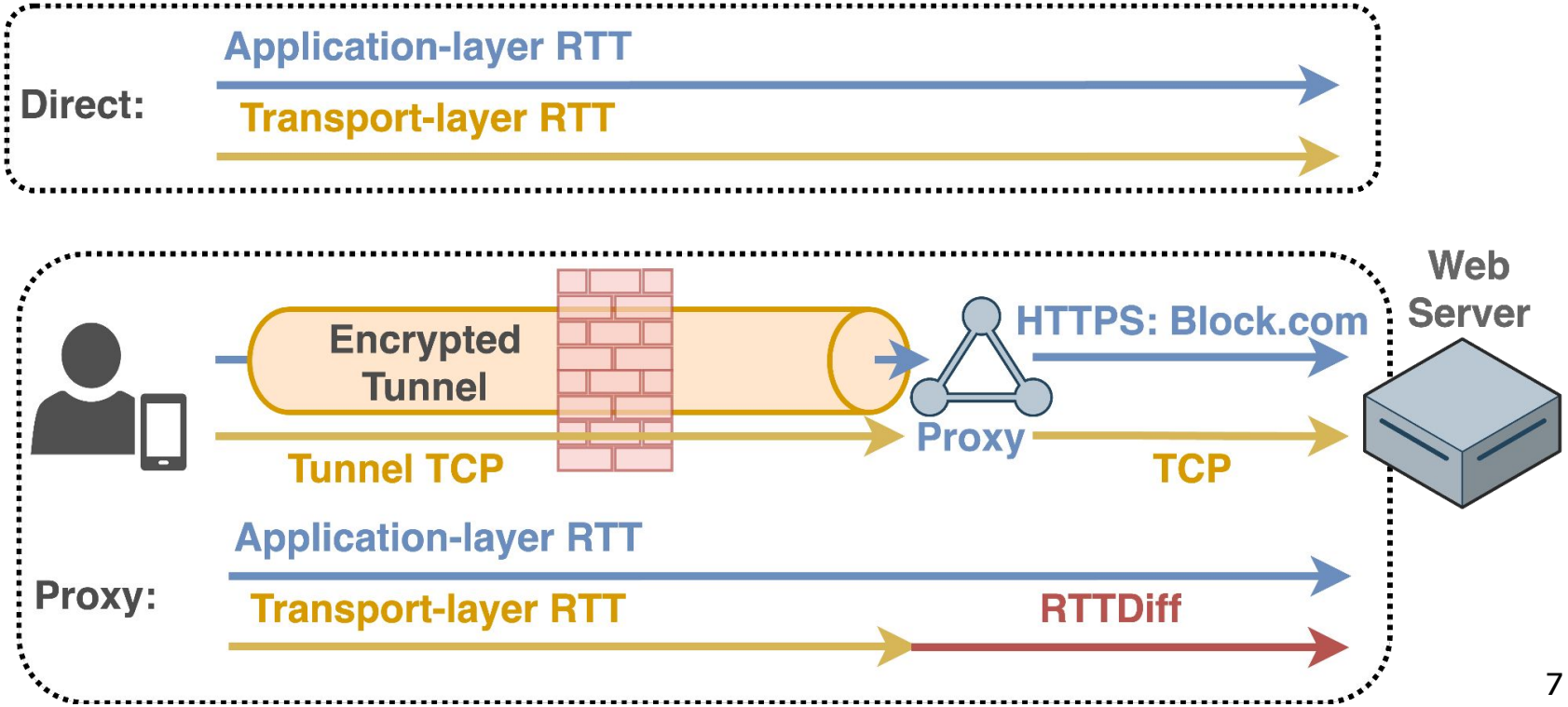
Open | Issue created 4 years ago by Cecylia Bocovich

Must firewalls fingerprint each protocol separately?

- **Only need to detect *any* tunnel, not the specific protocol.**

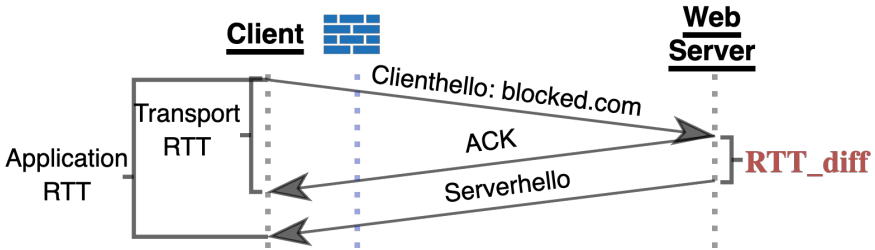
This Work: A timing-based, protocol-agnostic fingerprint for detecting traffic from obfuscated proxies.

Cross-layer RTT Diff as a Fingerprint for Tunnel



Censored Planet

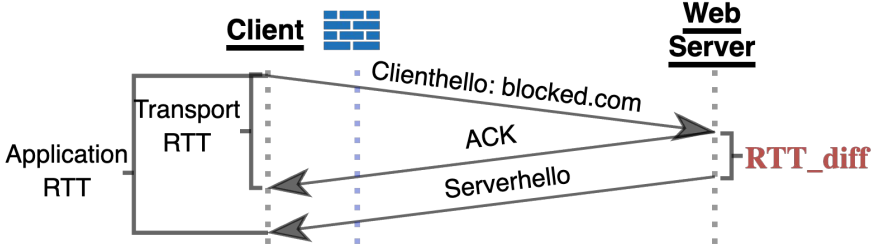
Cross-layer RTT Diff as a Fingerprint for Tunnel



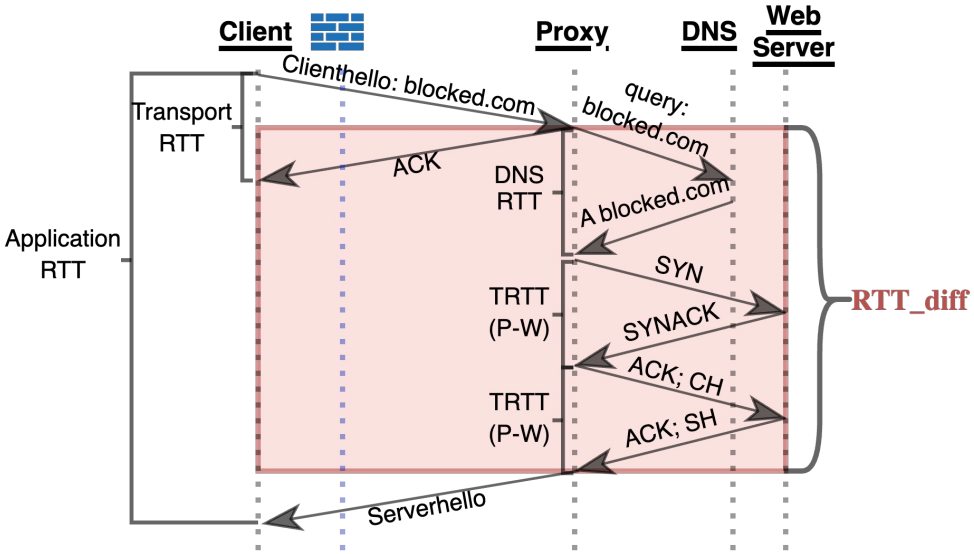
RTTdiff ~= processing delay

Censored Planet

Cross-layer RTT Diff as a Fingerprint for Tunnel



RTTdiff ~= processing delay



RTTdiff ~= processing delay + propagation delay

Censored Planet

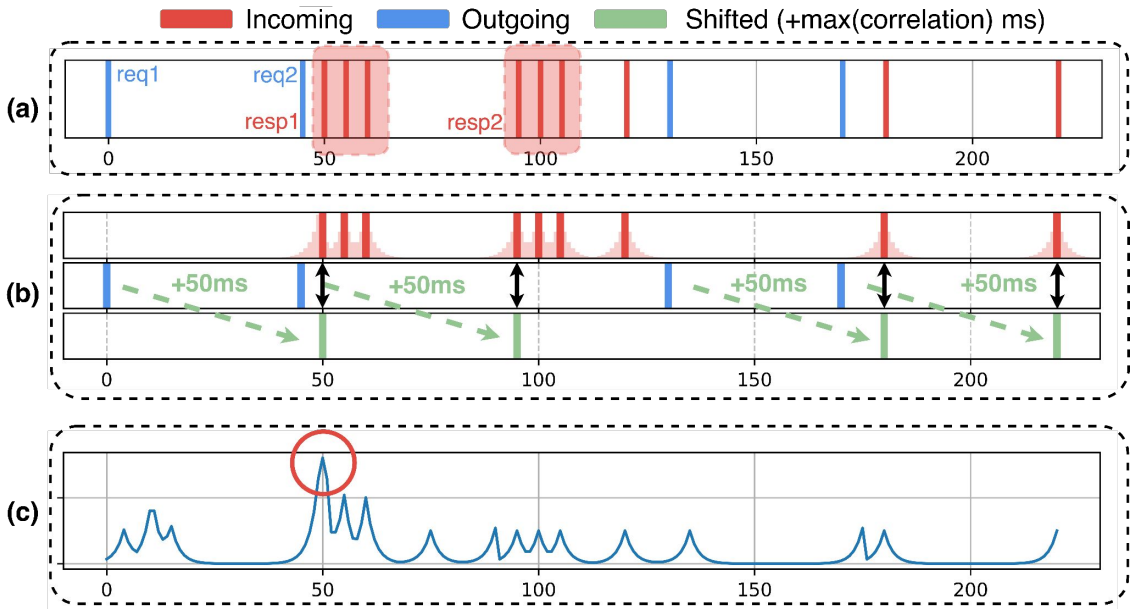
Efficacy & Assumptions

Fingerprint does *not* depend on…

- ↪ Client's location
- ↪ Firewall's relative position
- ↪ Tunneling protocol

What matters:

- ↪ Visibility of RTTDiff in the presence of encryption



Idea: cross-correlating request & response patterns to estimate application RTT

Censored Planet

Efficacy & Assumptions

Fingerprint does *not* depend on…

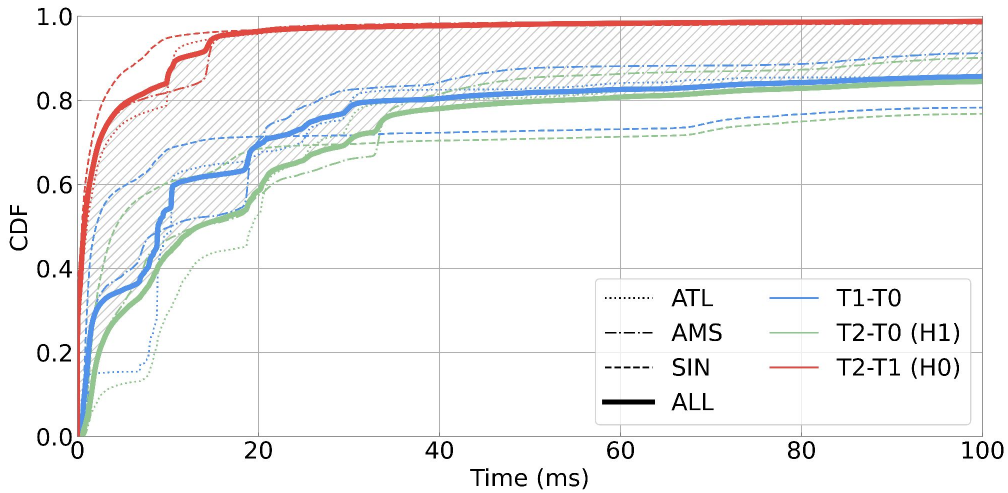
- ↪ Client's location
- ↪ Firewall's relative position
- ↪ Tunneling protocol

What matters:

- ↪ Visibility of RTTDiff in the presence of encryption
- ↪ Decide if observed RTTDiff indicates tunnel routing

Framework: Sequential Hypothesis Testing

- H0: Direct; H1: Proxied
- Identify the presence/absence of a prior, based on separation of distribution under different priors



Fingerprint Sensitivity

Setup:

- ➔ State-of-the-art popular obfuscated proxies
- ➔ Geographically distributed clients & proxies
- ➔ Top domains as web servers

Except obfs4, results across all protocols are practically identical

Proxy/Client	DTW	HKG	TYO	CEK	ARN
Remote DNS Resolution, CrUX Global 5K					
ATL	.207 / .819	.233 / .828	.219 / .784	.201 / .802	.215 / .791
SIN	.177 / .738	.172 / .727	.180 / .743	.199 / .732	.201 / .738
AMS	.201 / .775	.181 / .747	.201 / .759	.197 / .711	.172 / .766
Local DNS Resolution, CrUX Global 5K					
ATL	.372 / .905	.340 / .876	.377 / .880	.443 / .927	.448 / .907
SIN	.455 / .898	.313 / .842	.315 / .851	.438 / .892	.424 / .880
AMS	.435 / .905	.337 / .839	.328 / .856	.293 / .854	.339 / .877
Remote DNS Resolution, CrUX Regional 5K					
ATL	- / -	.186 / .712	.193 / .765	.410 / .879	.364 / .851
SIN	- / -	.147 / .719	.133 / .748	.330 / .851	.313 / .842
AMS	- / -	.176 / .658	.190 / .722	.352 / .827	.221 / .808

- ➔ Per-flow moderately effective, exposure amplified when aggregated by website visits
- ➔ Factors such as DNS handling and CDN connectivity would affect fingerprint's efficacy

Fingerprint Specificity (Collateral damage)

Setup:

- ➔ Collaborate with a regional ISP
- ➔ Apply fingerprint to mirrored real-user traffic (~50 Gbps)
- ➔ Conservatively consider all detections as false positives

Estimated FPR ~= 0.6%

- ➔ Comparable to reports of real-world censoring deployments*
- ➔ Potential categorical false positives (e.g., email)

Category	Identifier	Percentage of All Positives (%)
Rmt Port	443	57.88
	993	33.29
	80	4.47
	5222	0.43
	9001	0.30
SNI	apple.imap.mail.*.com	14.89
	imap.*.com	5.32
	android.imap.mail.*.com	2.91
	imap.mail.*.com	2.90
	*.*health.com	2.13
	(empty) / Not applicable	17.47

* How the Great Firewall of China Detects and Blocks Fully Encrypted Traffic. USENIX'23

Potential Mitigation

➔ Would a firewall deploy this?

- Demonstrated practicality; Broad applicability; Complementary to other detection methods
- Relies on subtle timing; Potential for non-trivial collateral damage.
- *Don't rely on network unreliability as the only defense.*

➔ Countermeasures discussed in the paper:

- Proxy configurations
- Multiplexing
- Traffic splitting
- TCP Delayed ACK
- Traffic scheduler

→ likely creating new timing patterns that are fingerprintable

Potential Mitigation

➔ obfs4 / Scramblesuit

- Seeded randomness at install; one random “shape” (timing, size) per obfs4 server
- Finding: *lower performance* **yet** *increase exposure*
  - When application is quiet, obfs4 is quiet
  - Only inflates packet delay/size, can’t obfuscate inherently large RTT/size patterns

Analogous observation:
Xue, Diwen, et al. “Fingerprinting Obfuscated Proxy Traffic with Encapsulated TLS Handshakes", USENIX’24

➔ Future directions

- Flexible obfuscation to support arbitrary timing patterns
- Define a “normal” timing shape
- Balance performance overhead
- Avoid convergent obfuscation that becomes new fingerprint

The Discriminative Power of Cross-layer RTTs in Fingerprinting Proxy Traffic

Diwen Xue, Robert Stanley, Piyush Kumar, Roya Ensafi
University of Michigan