Anticensorship in the Network Infrastructure

Eric Wustrow
Scott Wolchok
Ian Goldberg*
J. Alex Halderman

University of Michigan
*University of Waterloo
Background | Internet Censorship

- **No censorship**
- **Some censorship**
- **Country under surveillance from Reports Without Borders**
- **Most heavily censored nations**
Network-based Censorship

Government censors
Block websites containing “offensive” content
Commonly employ blacklist approach

Observed techniques
IP blocking, DNS blackholes, forged RST packets

Popular countermeasures
Mostly proxy based — Tor, Freenet, Ultrasurf, …
Problem: Cat-and-mouse game

Need to communicate proxy addresses to users but not to censors, or else they’ll be blocked too!
Our Approach | **Telex**

**Operates in the network infrastructure**
- Components placed at ISP between the censor's network and non-blocked portions of the Internet.
- We call this **end-to-middle proxying**

**Focuses on avoiding detection by the censor**
- Complements anonymity systems such as Tor

**Employs a form of deep-packet inspection**
- Turns common censor technology on its head

**Has no secrets to communicate to users in advance**
- Relies instead on public-key steganography

**Provides state-level response to state censorship**
- We envision government incentives for ISPs
Censor ... controls client’s network, but not external network
... blocks according to a blacklist
... allows HTTPS connections to non-blocked sites
Telex | Overview

User’s Computer | ISP Infrastructure | Websites

- Browser
- Censor
- Router

NotBlocked.com
unobjectionable site
(permitted by censor)

Blocked.com
prohibited site
(blocked by censor)

Request for permitted site | Request for prohibited site
Telex | Overview

User’s Computer | ISP Infrastructure | Websites

Browser | Telex Client | Router | NotBlocked.com (unobjectionable site, permitted by censor)

Censor | Request for permitted site | Request for prohibited site

Blocked.com (prohibited site, blocked by censor)
Telex | Overview

User’s Computer | ISP Infrastructure | Websites

- Browser
- Telex Client

Request for permitted site

Request for prohibited site

NotBlocked.com unobjectionable site (permitted by censor)

Blocked.com prohibited site (blocked by censor)
Telex | Overview

User’s Computer | ISP Infrastructure | Websites

- Browser
- Telex Client
- Censor
- Invisible Tag
- Router
- Telex Station
- NotBlocked.com unobjectionable site (permitted by censor)
- Blocked.com prohibited site (blocked by censor)

Request for **permitted** site

Request for **prohibited** site
1. **Client** starts TLS connection to **NotBlocked.com**

   ![Diagram showing a TLS ClientHello message with nonce T](image)

2. **Station** recognizes using private key, but **Censor** can’t tell from normal random nonce
3. Client negotiates TLS session key with NotBlocked and leaks it to Station

- Tag communicates shared secret S to Station
- Client uses S in place of random coins for key generation
- Station simulates Client, derives same TLS key
4. Station verifies Finished message from NotBlocked, switches from observer to MITM

[ GET http://blocked.com/ HTTP/1.0 ]

5. Client sends encrypted request for blocked content

6. Station intercepts, decrypts, and proxies request
Details | **Connection Tagging**

Application of **public-key steganography**

Client (anyone) generates tags
Station (and only the station) detects tags

Our requirements:
- Short (28 bytes)
- Indistinguishable from random (for the censor)
- Conveys a shared secret
- Fast to recognize (for the station)
- Low false positives

**Solution:** Diffie-Hellman over elliptic curves … *with a twist!*
Telex | Prototype Implementation

CAUTION Experimental proof-of-concept software. *Not safe for use under real-world censorship!*
Prototype | Flow Diversion

From client To NotBlocked.com

 TLS Traffic to Telex Injected packets Flow dropper commands

Capable of dropping flows on command (e.g. “stop automatically forwarding for client ↔ NotBlocked.com”)

Sends copy of incoming TLS packets to other Telex components

Telex components may inject spoofed packets as either endpoint

We use software router (Linux/iptables/ipset)
Prototype | Tag Recognition

Reconstructs TCP flows, extracts TLS nonces, etc.

Based on Bro for flow reconstruction, fast elliptic curve code
  Checks 11,000 tags/second-core on 3GHz Intel Core 2 Duo

When tag found, commands router to drop flow,
  then explicitly forwards packets until end of TLS handshake

300 SLOC Bro script; 450 SLOC C++
Prototype | Proxy Service

Shunts data between client’s TLS connection and configurable services

Kernel module creates spoofed TCP socket, glue library strips SSL

1250 SLOC C

---

HTTP Proxy (Squid)

Tor Bridge

SSH Server

Blocked.com

Proxied request

alternative to port knocking
Prototype | **Telex Client**

Forwards arbitrary TCP port via tagged TLS connections
Based on libevent and (modified) OpenSSL
Currently Windows and Linux
1200 SLOC C++
Prototype | Test Deployment

Single Telex Station on lab-scale “ISP” at Michigan

Hosted sites

- **NotBlocked.telex.cc**
  - Unobjectionable content *

- **Blocked.telex.cc**
  - Simulated censored site
  - only reachable via Telex

Early experiences

Three authors used Telex for daily browsing since May

Streamed HD YouTube via PlanetLab node in Beijing

Also, I got haxed ... whoops!
Eric says: Kittens are cute!
**Goal:** Resist realistic passive or active attacks that would deny service on a wide scale

**Future:** Respond to growing censor sophistication

*Censors might try to …*

- Perform deep traffic analysis
- Tunnel traffic around Telex (buy VPN …)
- Mandate own HTTPS proxies or CAs
- Block every potential NotBlocked.com
- Employ various routing tricks
- DoS the Telex Stations
Where to deploy? (And how to model?)
How to convince ISPs to deploy?
Scaling Telex DPI to core network?
Preventing private key compromise?
End-to-middle proxying—
New approach to resisting Internet censorship
Focus on hiding use of the service
Based on public-key steganography, repurposes DPI and MITM for anti-censorship
Proof-of-concept operating today, but wide-scale deployment needs ISP or (perhaps) government cooperation
Telex | Anticensorship in the Network Infrastructure

https://telex.cc

Eric Wustrow          Scott Wolchok
Ian Goldberg*       J. Alex Halderman

University of Michigan
*University of Waterloo