A Large-scale Analysis of Content Modification by Open HTTP Proxies

Giorgos Tsirantonakis,* Panagiotis Ilia,* Sotiris Ioannidis,*
Elias Athanasopoulos,+ Michalis Polychronakis#

* FORTH, Greece
+ University of Cyprus, Cyprus
# Stony Brook University, USA

FORTH
Foundation for Research & Technology - Hellas

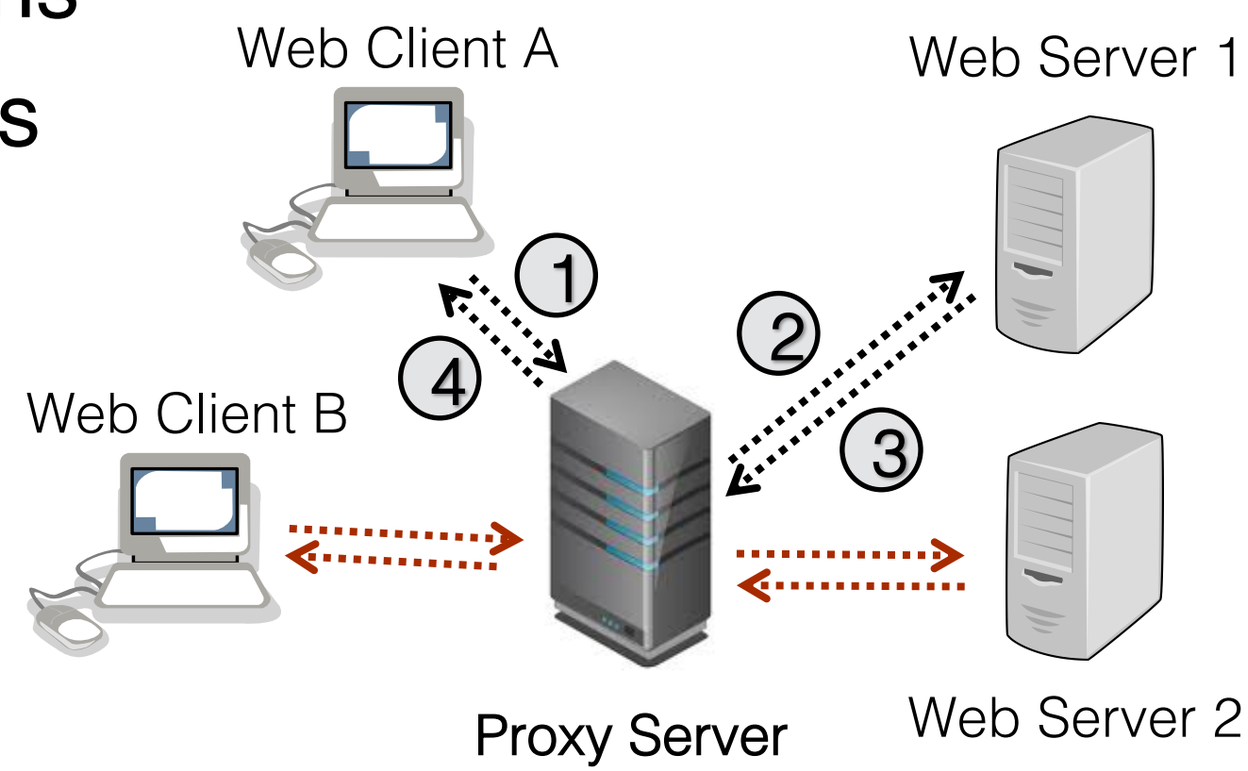
NDSS '18, **February 20, 2018, San Diego, CA, USA**

Outline

- Introduction / Motivation
- Objectives
- Methodology
- Analysis
  - Proxy characteristics
  - Malicious behavior
- Conclusions

Introduction

- HTTP / HTTPS proxies are popular
  - Numerous proxy list websites
  - Thousands of proxies
- Access content that is blocked
  - Geographical restrictions
  - Content filtering policies
  - Censorship
- “Preserve” anonymity
  - Hide IP address
  - Ad Blocking



Introduction

- Obviously, HTTP proxies can possibly
  - Tamper with transmitted content
  - Snoop for sensitive user data
- A **malicious** proxy can **monetize** traffic
  - Inject / replace ads
  - Collect sensitive information
  - Distribute or spread malware / spyware
  - Mount phishing attacks
  - Inject code for XSS, DDoS, crypto-currency mining etc.

Motivation

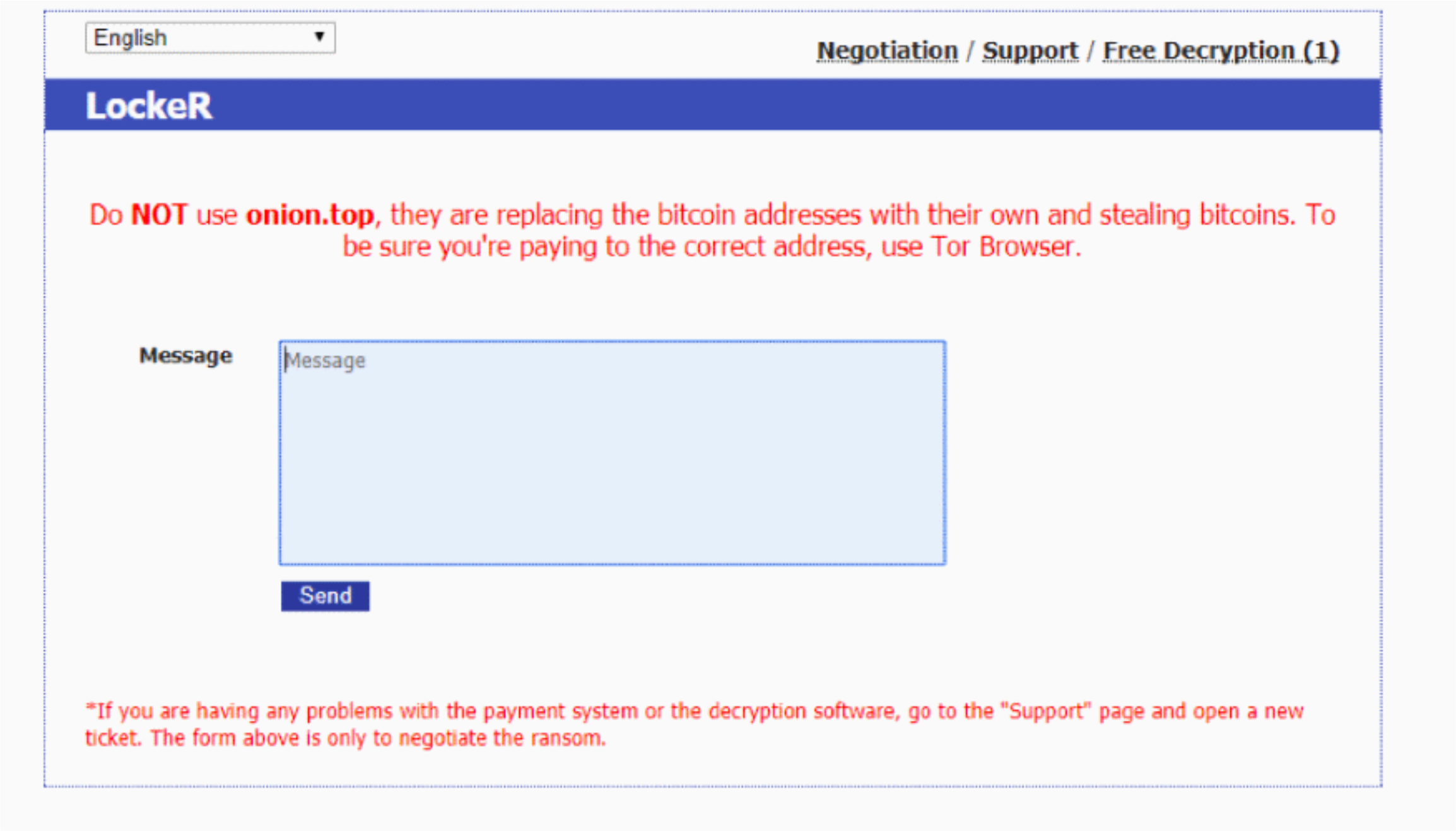
- Owning bad guys {& mafia} with javascript botnets

  (**Chema and Fernandez, DEFCON '12**)

  - Modify JS files to dynamically fetch malicious code
  - Collect cookies and user sensitive information
  - Take control of infected hosts (e.g., botnet)

- Onion.top proxy service
  - Tor-to-Web proxy (allows access to .onion domains)
  - Replace bitcoin address on ransomware payment sites
    - **LockeR**, **Sigma**, and **GlobeImposter**

Motivation


English
Negotiation / Support / Free Decryption (1)
LockeR
Do NOT use onion.top, they are replacing the bitcoin addresses with their own and stealing bitcoins. To be sure you're paying to the correct address, use Tor Browser.
Message
Message
Send
*If you are having any problems with the payment system or the decryption software, go to the "Support" page and open a new ticket. The form above is only to negotiate the ransom.

Objectives

- Detect cases of content modification
- Understand and assess proxies' behavior
- Measure the extent of content modification by rogue proxies

We designed and built a framework that

- Collects public HTTP proxies daily
- Tests proxies daily
  - 2 decoy websites (***honeysites***) & http://bbc.com
- Content modification detection (DOM Comparison)

Our service - http://proxyscan.ics.forth.gr

proxyscan.ics.forth.gr/

proxyscan.ics.forth.gr

Search

A Large-scale Analysis of Content Modification by Open HTTP Proxies

Open HTTP proxies offer a quick and convenient solution for routing web traffic towards a destination. They are an attractive option for bypassing IP-based filters and geo-location restrictions, circumventing content blocking and censorship, and in general, hiding the client's IP address when accessing a web server. Nevertheless, the consequences of routing traffic through an untrusted third party can be severe.

In order to determine the extent to which user traffic is manipulated while being relayed, we have designed a methodology for detecting proxies that, instead of passively relaying traffic, actively modify the relayed content. Beyond simple detection, our approach is capable of macroscopically attributing certain traffic modifications at the network level to well-defined malicious actions, such as ad injection, user fingerprinting, and redirection to malware landing pages.

This work will be presented in NDSS '18. More information about our methodology and findings can be found in the paper. In the following, we provide a list of proxies that were found to perform malicious content modifications/injections. As we continue our tests, this list will be updated on a daily basis to include all the newly found ones.

We strongly advise you to NOT use any proxy that is included in the following list!

Table with all malicious proxies | **Search a specific proxy**

The proxies in the following list were found to inject content in the fetched websites

IP	Port	Country
36.81.185.223	80	ID
222.34.238.133	8998	CN
182.121.17.158	8998	CN
123.235.54.176	8998	CN
103.58.117.228	3128	IN

Methodology - Collecting Proxies

Google search for “HTTP proxy list” – first **50** results

- Didn’t consider subscription-based list websites
- Left out identical / very similar websites

➢ **15 different popular proxy list websites**

For 2 months

- Automatically crawl **10** websites (daily)
- Manually exporting proxies from **5** sites (every 10 days)
  - Require registration, CAPTCHA etc.
- 1 subscription-based website (every 5 days, 1 month)

Methodology – Use of Honeysites

How can we test a Proxy?

- We could fetch a website twice
  - Once with a proxy, and once without

But, this does not work very well

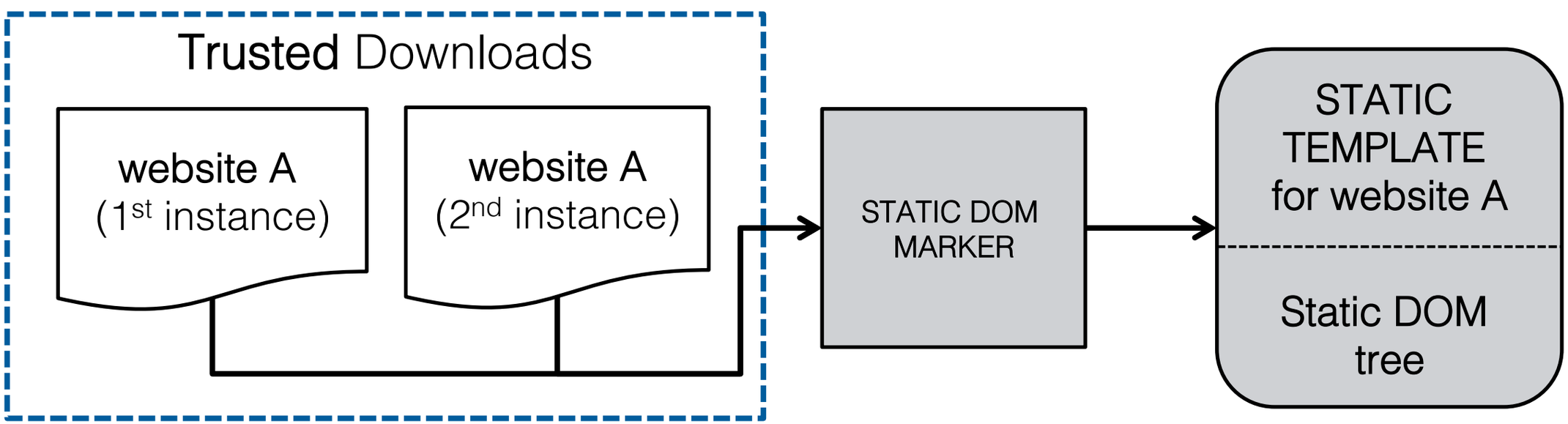
- Modern websites are highly-dynamic
  - e.g., content changes according to geolocation
- We cannot control the behavior of real websites

Thus, we use **decoy websites under our control**

Methodology – Use of Honeysites

Decoy websites under our control

- honeysite $\mathbf{h_1}$ – simple, completely **static**
- honeysite $\mathbf{h_2}$ – contains **dynamic** content
  - WordPress, contains JS elements
  - Fake ads - Google AdSense, Media.net & BuySellAds



Methodology – Testing the Proxies

- Fetch all 3 testing websites through a proxy
- Compare DOM tree with honeysite's **static template**
  - Identify content modification / injection of elements
- Do not compare **dynamic elements**
  - They are dynamic, they change anyway
  - But, we expect them to **change in a predictable way**
    - e.g., ad should be fetched from specific ad network

Methodology – Probing the Proxies

- Large number of proxies in our set
  - Collect proxies systematically
- Proxies are slow and not very reliable
  - ➢ Timeout interval **180** seconds
- Cannot test them all, multiple times per day, every day
  - Use TCP probes to identify responding (**alive**) proxies
  - A few probes almost every hour, **22 times per day**
  - When a proxy responds (one probe at least), we test it

Methodology – Clustering

- Two-level clustering
  - Identify position and type of injected elements
  - Group identical/similar cases together

- Keep track of the sequence of elements
  - Identify proxies that do not inject, but remove elements

- Manually inspected downloads from each cluster
- Use Firefox (with Firebug) to render downloads
  - Monitor outgoing requests to 3rd party domains

Analysis

- **144,349** proxies collected
- **65,871** unique proxies in our dataset
  - Same proxies exist in multiple proxy lists

- **49,444** alive proxies (responded to probes)
- **19,473** working proxies (fetched honeysites)

7,441 content modifying proxies (38.21%)

1,004 malicious proxies (5.15%)

Analysis

7,441 Content modifying proxies

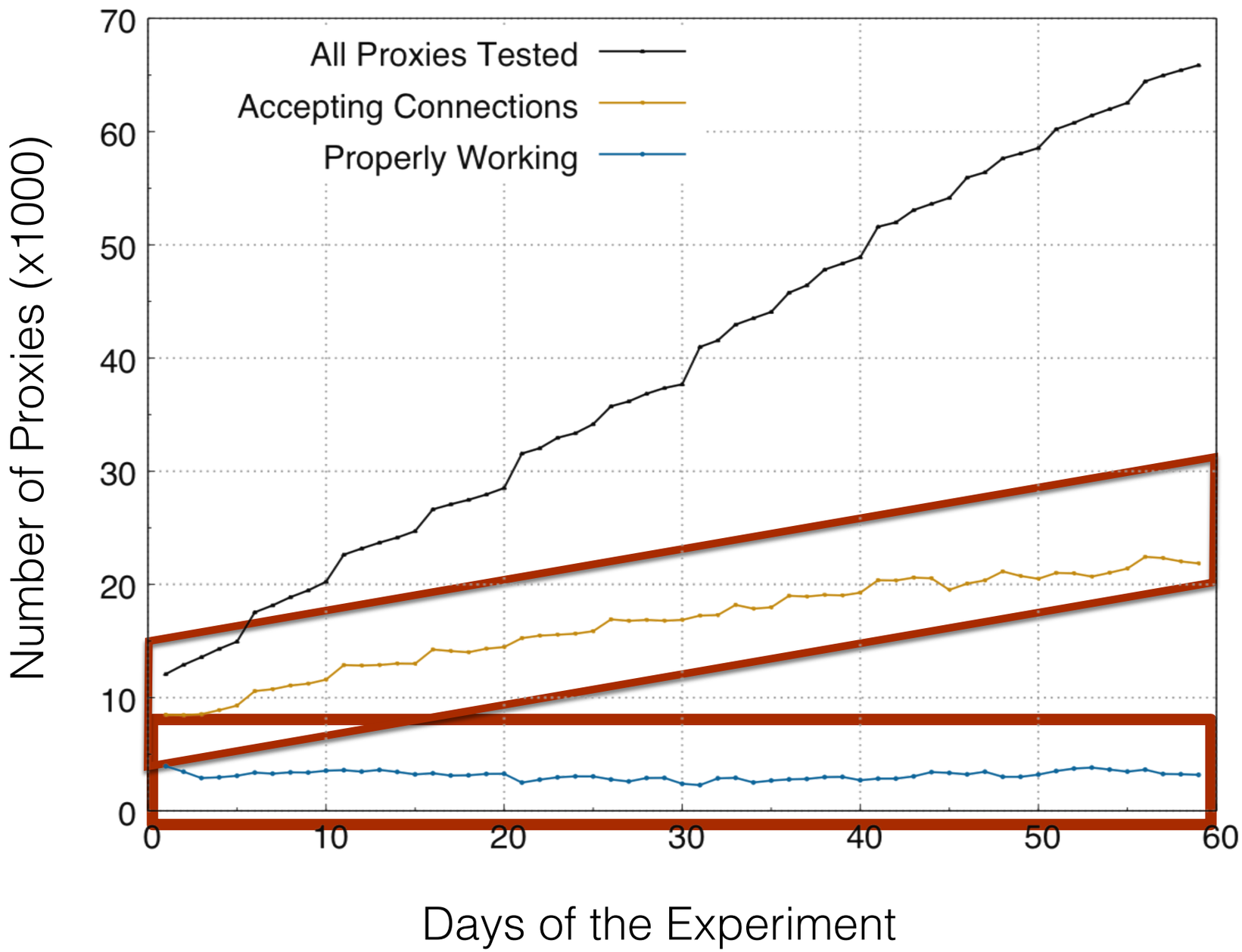
- Not necessarily malicious
- Most of them are **"privacy preserving"** proxies
  - Block content from 3rd parties (e.g., ads)
- Some proxies are **suspicious,** but **not malicious**
  - e.g., inject empty HTML elements

1,004 Malicious proxies

- Inject additional new content
- Replace existing content
- Block existing and inject new content

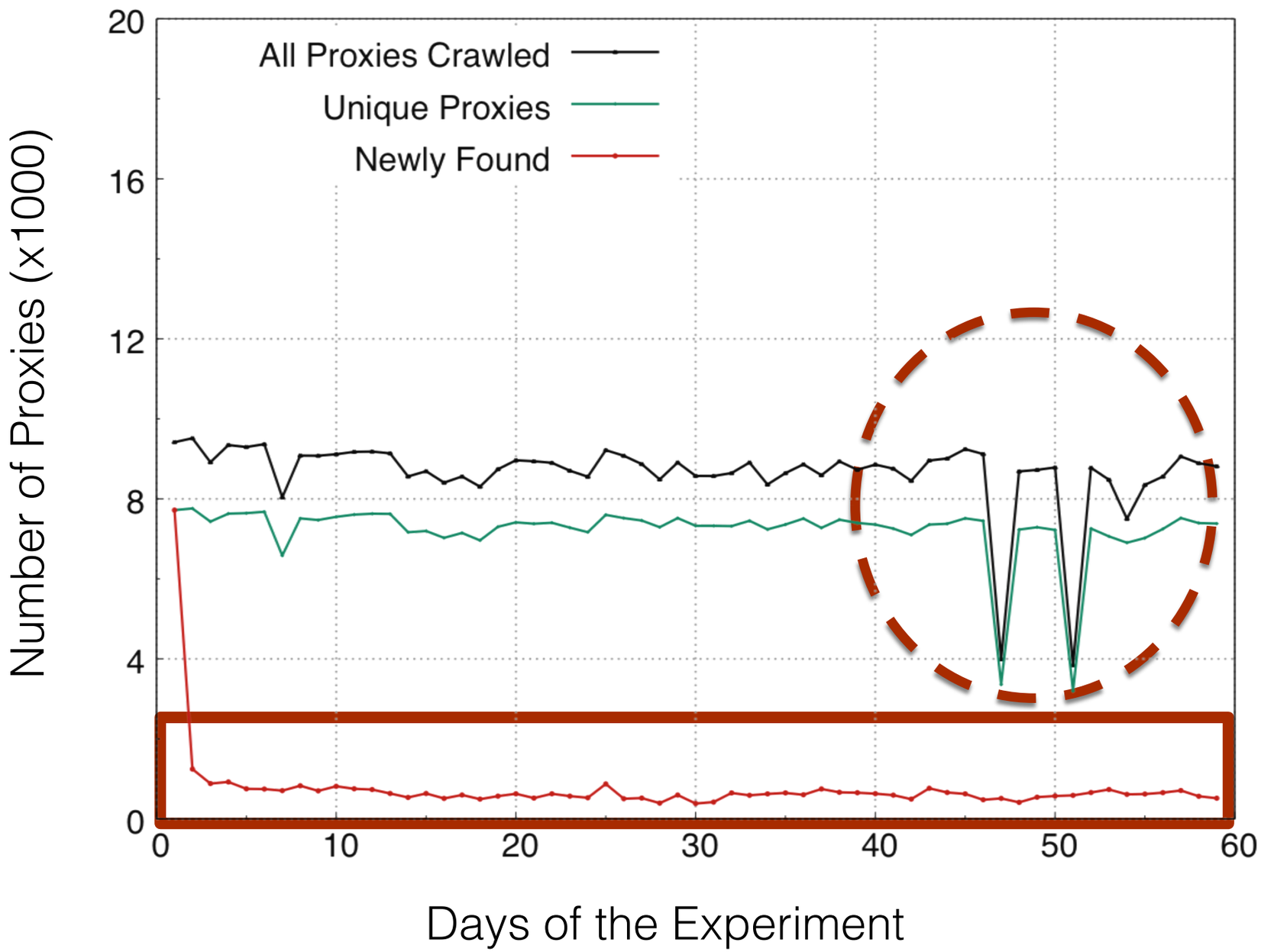
Analysis – Proxy Characteristics

Proxies in our dataset (per day)



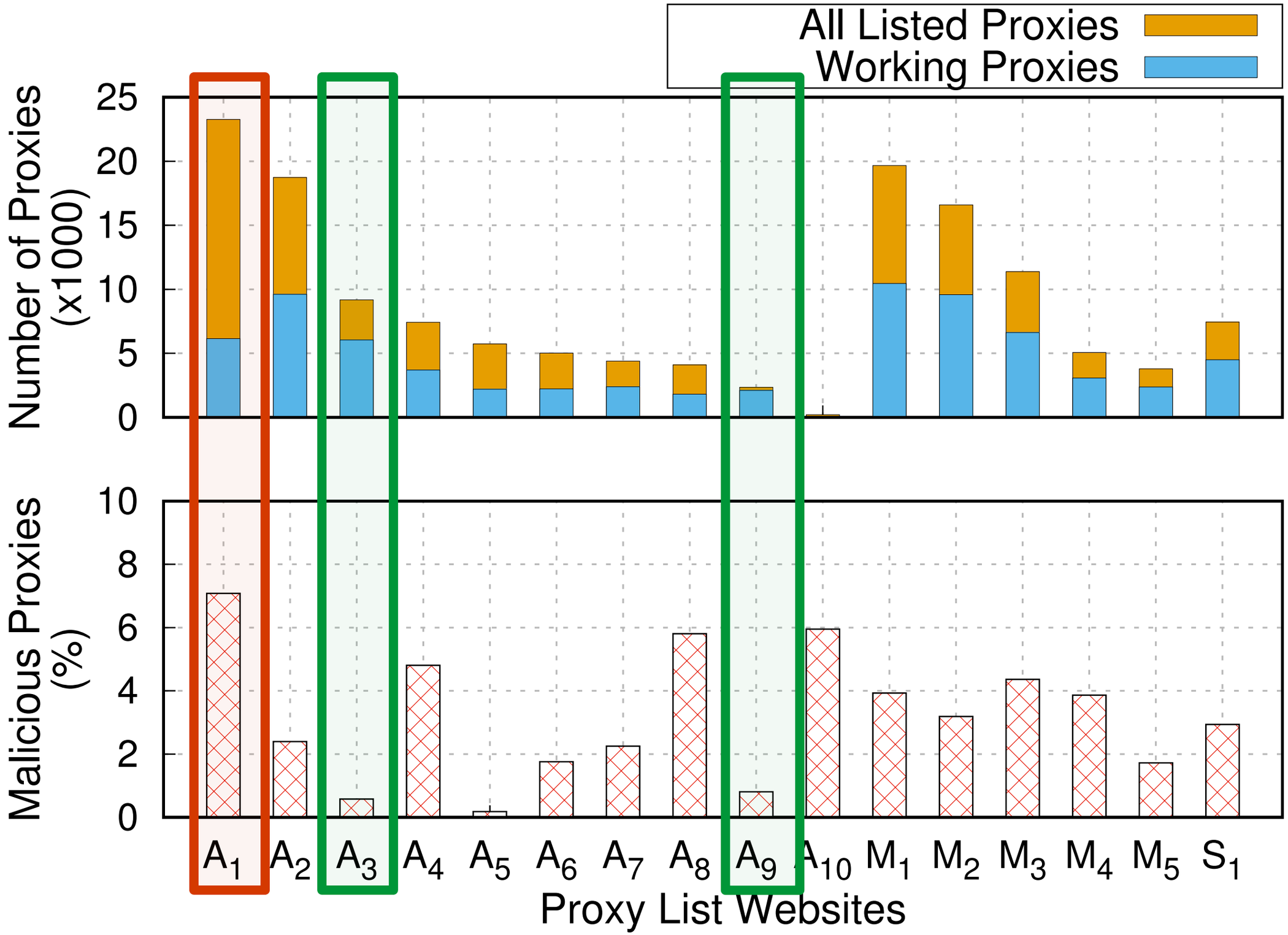
Analysis – Proxy Characteristics

Proxies crawled every day (10 list websites)

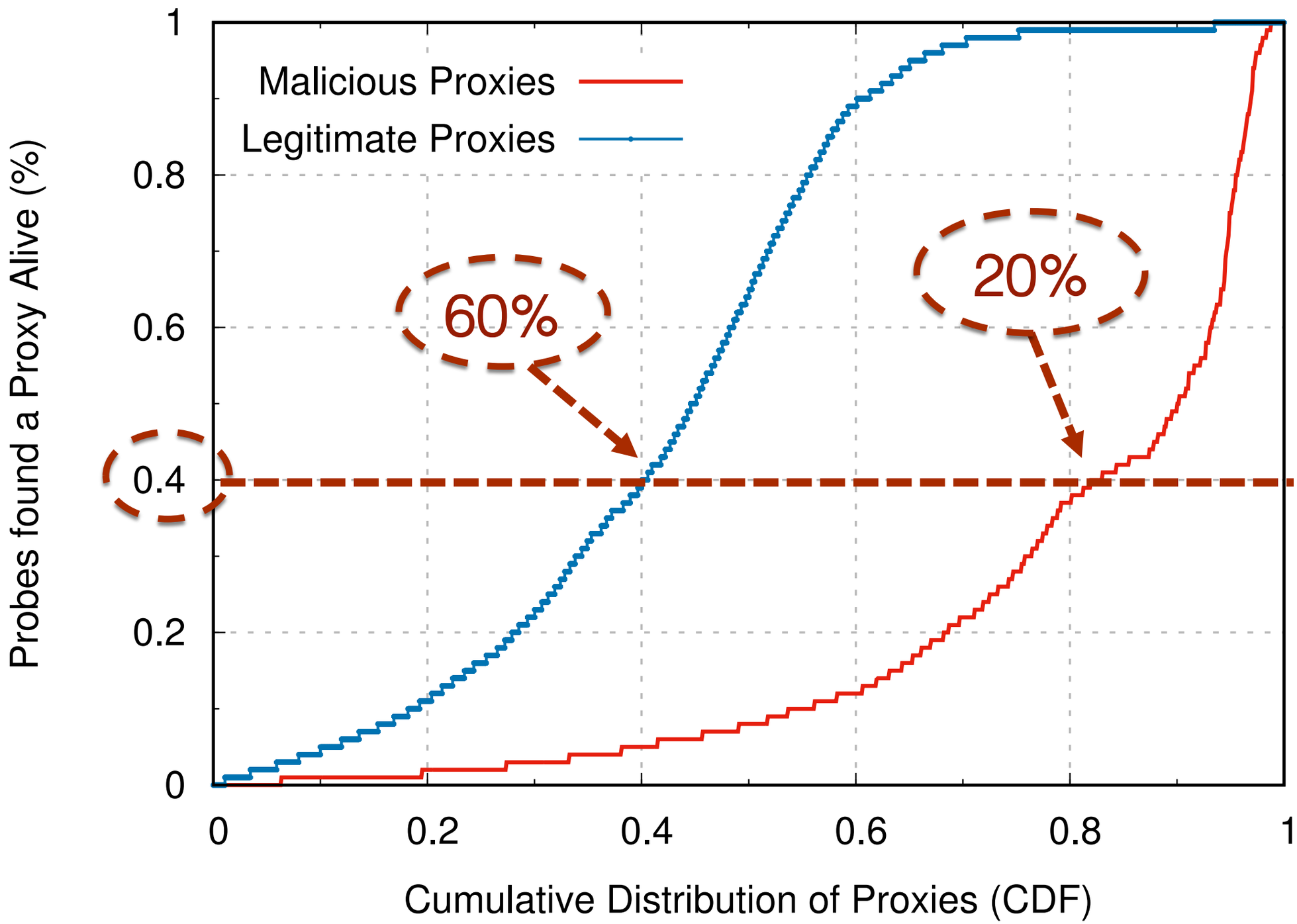


4-6% new proxies

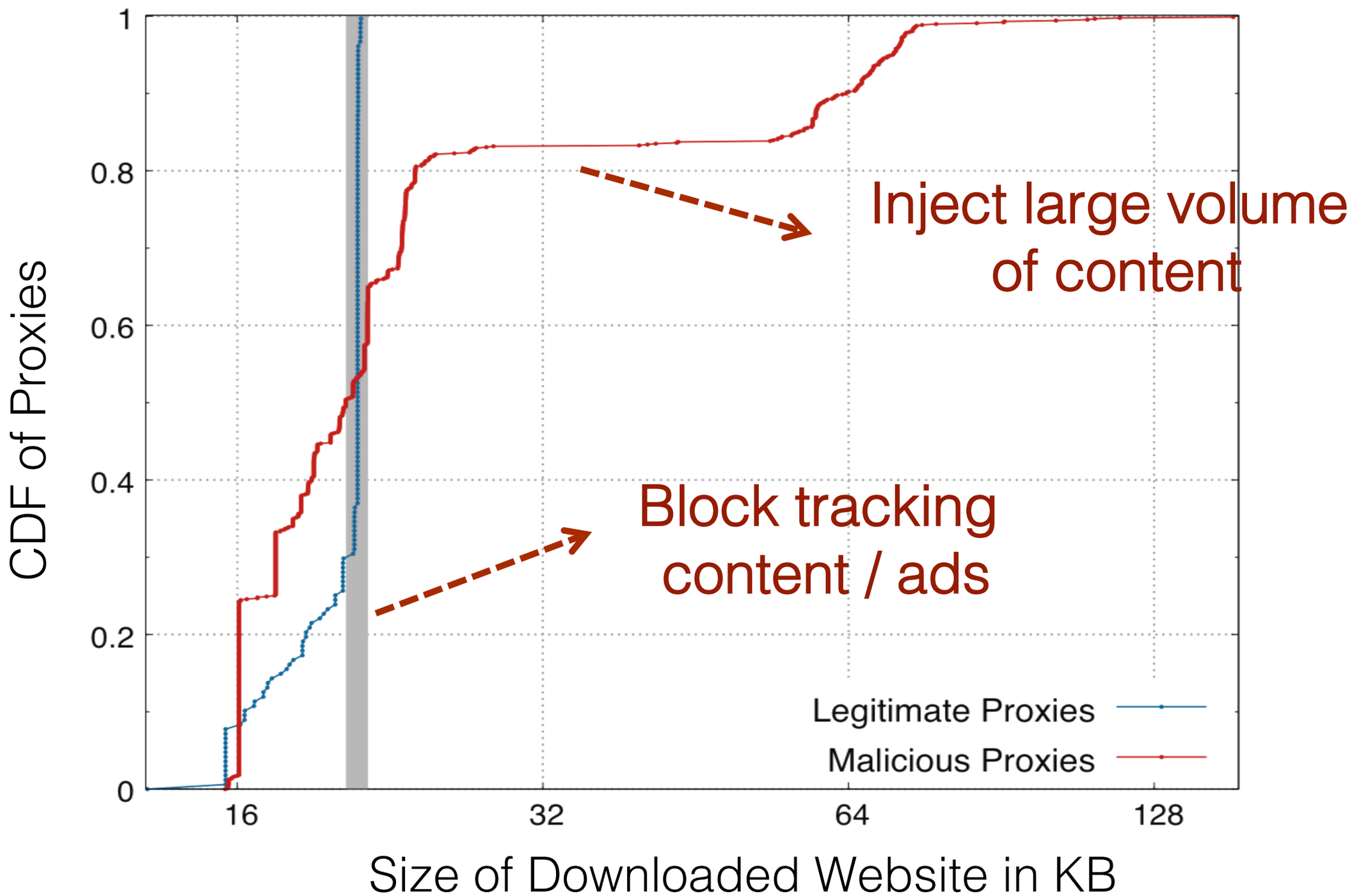
Analysis – Proxy List Websites



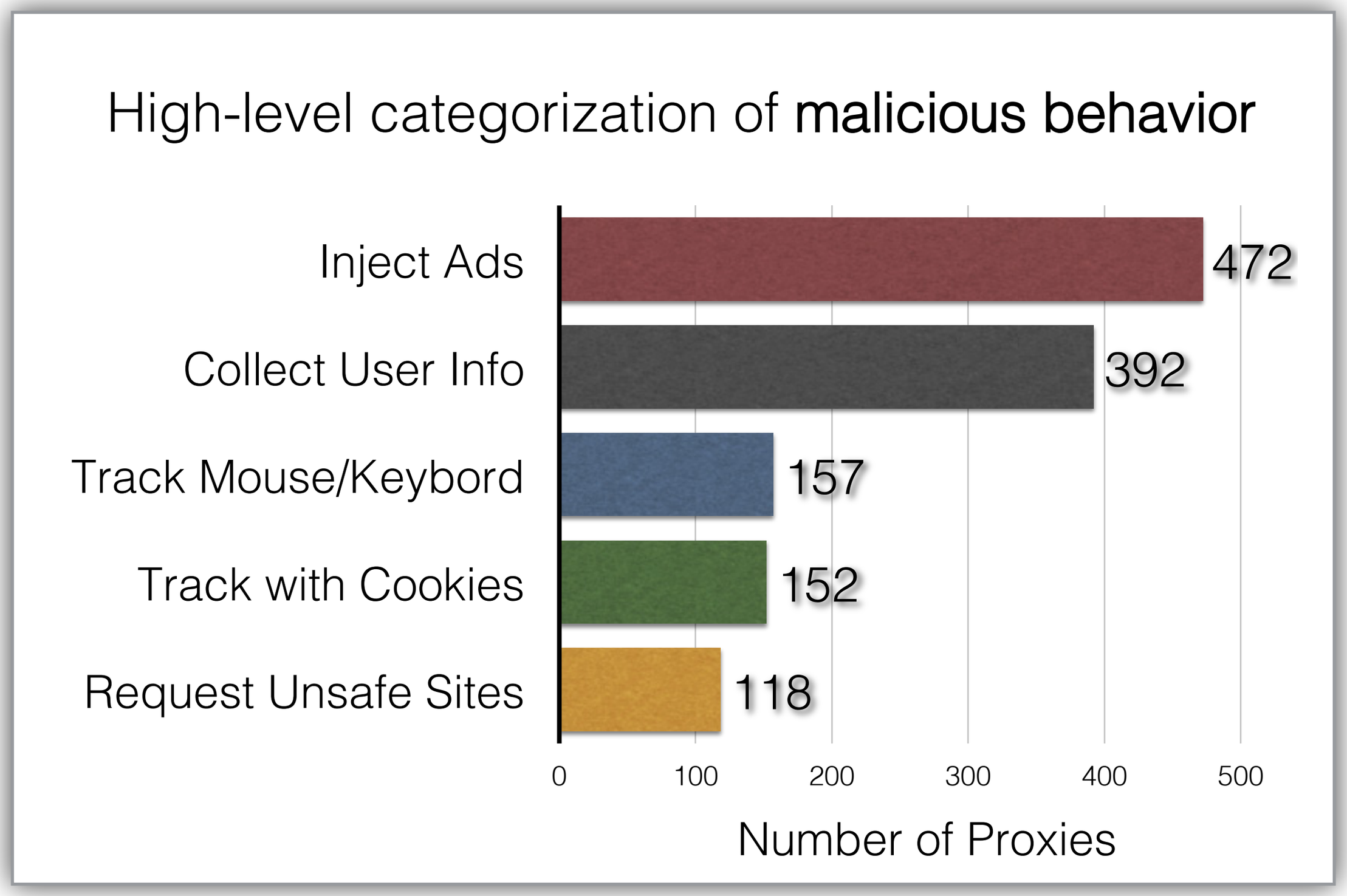
Analysis – Lifetime & Reliability



Analysis – Size of Fetched Content

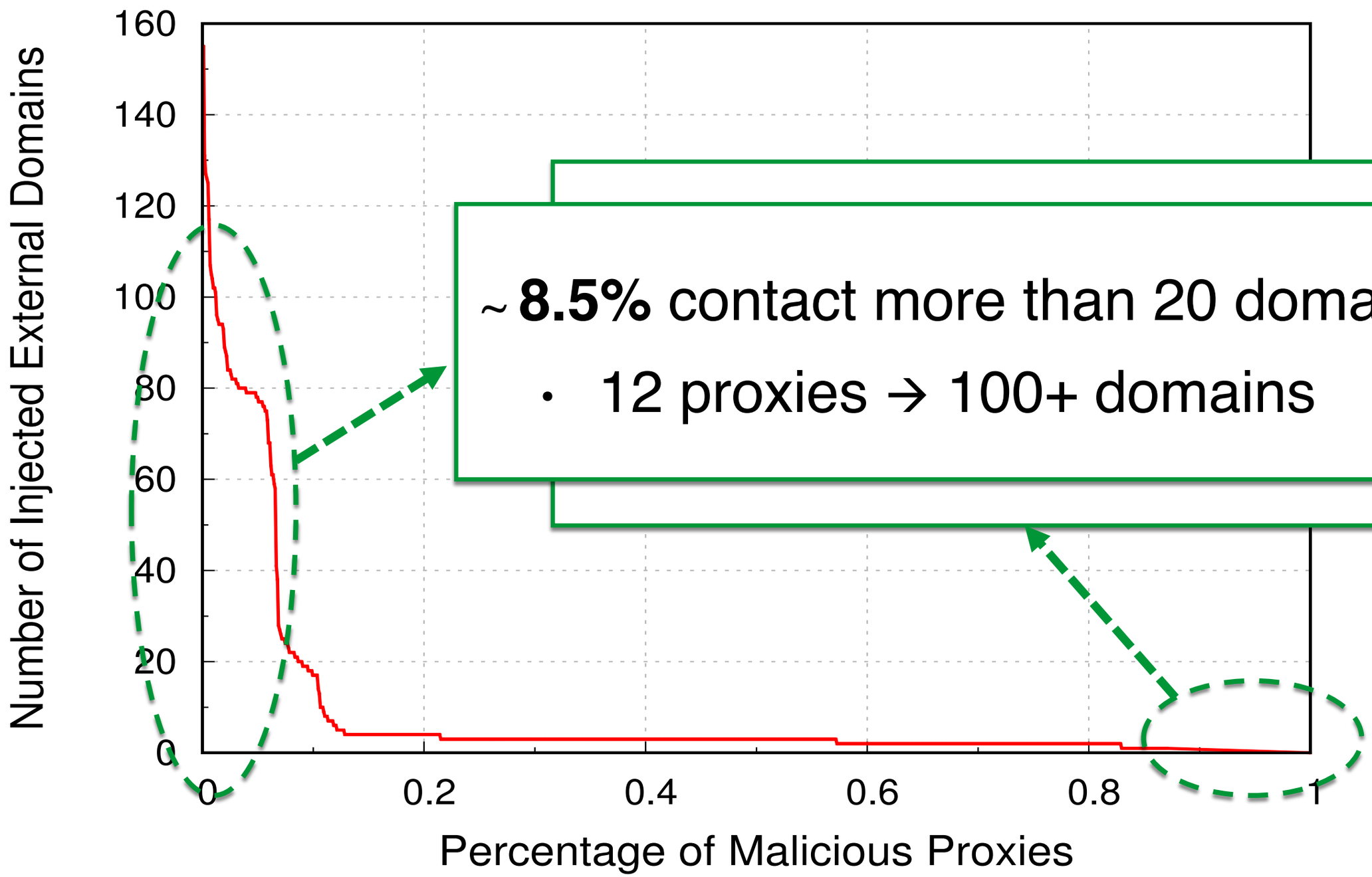


Analysis – Malicious Proxies



Analysis – Malicious Proxies

Outgoing requests to 3rd parties



~ **8.5%** contact more than 20 domains

- 12 proxies → 100+ domains

Analysis – 3rd Party Domains

tongji.baidu.com	556	www.onclickcool.com	104
cfs.uzone.id	140	agm.abounddinged.com	104
a01.uadexchange.com	124	yellow-elite.men	103
up.filmkaynagi.com	113	demisedcolonnaded.com	102
a.akamaihd.net	109	intext.nav-links.com	102
urlvalidation.com	107	www.tr553.com	101
i.qkntjs.info	106	ruu.outputsteddy.com	101
adnotbad.com	106	s.lm15d.com	74
ratexchange.net	105	rtax.criteo.com	72
i.tonginjs.info	105	www.donation-tools.org	69

Analysis – Interesting Findings

Some proxies change behavior according to relayed content

- 37 proxies injected content in $\boldsymbol{h_2}$ but not $\boldsymbol{h_1}$
- 10 proxies injected ads only in AdSense's iframes

- 2 proxies replaced ***publisher's ID*** with theirs
  (ads from Media.net)

- 41 malicious proxies **did not always** perform injections
  - Injected scripts/ads sporadically, only in some tests
  - In other tests, **exhibited benign behavior!**

Limitations / Future Work

- Rogue proxy operators **may anticipate our testing attempts**
- Honeysites **can be easily identified**
  - Larger and more diverse set of honeysites
  - Expose only few honeysites to each proxy
  - Specialized honeysites e.g., banking, health
- Include more **proxy list websites**

Conclusions

- Rogue proxies can modify / inject content
- Designed a framework
  - Collect proxies from 15 popular proxy list websites
  - Test them regularly with the use of decoy websites
- Only 19,473 proxies found to work properly
- Detected 1,004 malicious proxies
- Analyzed their behavior, with regards to relayed content

http://proxyscan.ics.forth.gr