

The Anatomy of Anonymity

From Leaks and Loopholes in Internet Protocols
to Power Plays and the Politics of Encryption

Juan Tapiador

Director, COSEC Lab, UC3M

*"You can fool some of the people all of the time,
and all of the people some of the time, but you
cannot fool all of the people all of the time"*

–Attributed to Abraham Lincoln

Prelude

What does "anonymity" actually mean?

Anonymity is the state of being not identifiable (within a set of subjects)

Undetectability

Unobservability

Unlinkability

Pseudonymity

Deniability

Anonymity, Unlinkability, Undetectability, Unobservability, Pseudonymity, and Identity Management – A Consolidated Proposal for Terminology

(Version v0.31 Feb. 15, 2008)

Andreas Pfitzmann
TU Dresden
pfitza@inf.tu-dresden.de

Marit Hansen
ULD Kiel
marit.hansen@datenschutzzentrum.de

Archive of this Document

http://dud.inf.tu-dresden.de/Anon_Terminology.shtml (v0.5 and

Starting with v0.20, color is essential to understand the figures

Abstract

Based on the nomenclature of the early papers in the field, we both expressive and precise. More particularly, we define *anonymity*, *unobservability*, *pseudonymity* (*pseudonyms* and *digital pseudonymity*), *identity management*. In addition, we describe the relationship rationale why we define them as we do, and sketch the main properties defined.

Table of contents

1 Introduction	
2 Setting	
3 Anonymity	
4 Unlinkability	
5 Anonymity in terms of unlinkability	
6 Undetectability and unobservability	
7 Relationships between terms	
8 Known mechanisms for anonymity, undetectability, and unobservability	

sciendo

Proceedings on Privacy Enhancing Technologies ; 2019 (2):105–125

Christiane Kuhn*, Martin Beck, Stefan Schiffner, Eduard Jorswieck, and Thorsten Strufe

On Privacy Notions in Anonymous Communication

Abstract: Many anonymous communication networks (ACNs) with different privacy goals have been developed. Still, there are no accepted formal definitions of privacy goals, and ACNs often define their goals ad hoc. However, the formal definition of privacy goals benefits the understanding and comparison of different flavors of privacy and, as a result, the improvement of ACNs. In this paper, we work towards defining and comparing privacy goals by formalizing them as privacy notions and identifying their building blocks. For any pair of notions we prove whether one is strictly stronger, and, if so, which. Hence, we are able to present a complete hierarchy. Using this rigorous comparison between notions, we revise inconsistencies between the existing works and improve the understanding of privacy goals.

Keywords: Anonymity, Privacy notion, Anonymous Communication, Network Security

DOI 10.2478/popets-2019-0022

Received 2018-08-31; revised 2018-12-15; accepted 2018-12-16.

able. Additionally, many conceptual systems, like Mix-Nets [6], DC-Nets [4], Loopix [15] and Crowds [16] have been published.

The published ACNs address a variety of privacy goals. However, many definitions of privacy goals are ad hoc and created for a particular use case. We believe that a solid foundation for future analysis is still missing. This hinders the understanding and comparison of different privacy goals and, as a result, comparison and improvement of ACNs. In general, comparing privacy goals is difficult since their formalization is often incompatible and their naming confusing. This has contributed to a situation where existing informal comparisons disagree: e.g., Sender Unlinkability of Hevia and Micciancio's framework [12] and Sender Anonymity of AnOA [1] are both claimed to be equivalent to Sender Anonymity of Pfitzmann and Hansen's terminology [14], but significantly differ in the protection they actually provide. These naming issues further complicate understanding of privacy goals and hence analysis of ACNs.

To allow rigorous analysis, i.e. provable privacy, of ACNs, their goals need to be unambiguously defined.

Traffic Analysis

Goal is to hide the source, the destination, and/or the content of Internet flows from eavesdroppers

Traffic Analysis

Goal is to hide the **source**, the **destination**, and/or the content of Internet flows from eavesdroppers

What can be learned from leaks in network protocols and from access to systems?

Traffic Analysis

Goal is to hide the **source**, the **destination**, and/or the **content** of Internet flows from eavesdroppers

What can be learned from **leaks in network protocols** and from **access to systems**?

Increasingly unavailable except “at the end” because of **E2EE**

Irrelevant in many applications

Traffic Analysis



Goal is to hide the **source**, the **destination**, and/or the **content** of Internet flows from **eavesdroppers**

What can be learned from leaks in network protocols and from access to systems?

Increasingly unavailable except “at the end” because of E2EE

Irrelevant in many applications

Observability depends on vantage point

Traffic Analysis



Goal is to hide the **source**, the **destination**, and/or the **content** of Internet flows from **eavesdroppers**

What can be learned from **leaks in network protocols** and from **access to systems**?

Increasingly unavailable except “at the end” because of **E2EE**

Irrelevant in many applications

Observability depends on **vantage point**

User / Application

Application layer

Transport layer

Network layer

Link layer

Hardware layer

Traffic Analysis



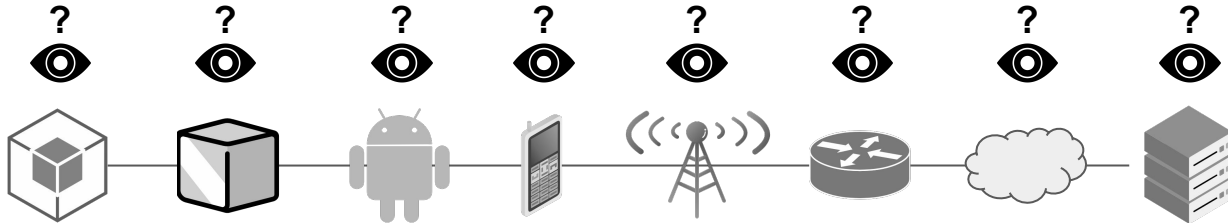
Goal is to hide the **source**, the **destination**, and/or the **content** of Internet flows from **eavesdroppers**

What can be learned from leaks in network protocols and from access to systems?

Increasingly unavailable except “at the end” because of **E2EE**

Irrelevant in many applications

Observability depends on vantage point



User / Application

Application layer

Transport layer

Network layer

Link layer

Hardware layer

This talk

Expectations management

Review of some key Internet networking technologies

- HTTP + DNS
- TLS/HTTPS + Do(H|T|Q), oDoH
- Middleboxes
- Domain fronting, ESNI, and ECH
- The skunk in the room: Web PKI
- ~~Proxies / VPNs~~

Focus on what (meta-)data network protocols leak to different eyeballs

- This is critical to understanding
 - The different in-path and out-of-path adversaries in the Internet
 - The strengths and weaknesses of different privacy technologies

Interlude

A simplified view of Internet communications

INTERVIEW

TELECOMMUNICATIONS

Vint Cerf on 3 Mistakes He Made in TCP/IP

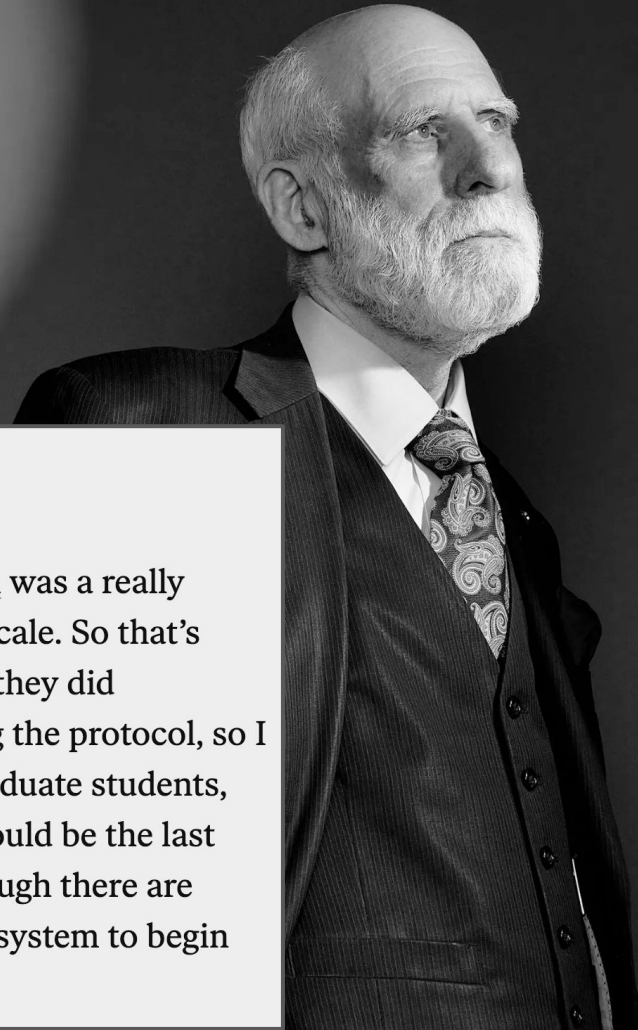
> The co-creator of the Internet's protocols admits his crystal ball had a few cracks

BY TEKLA S. PERRY | 07 MAY 2023 | 2 MIN READ | 

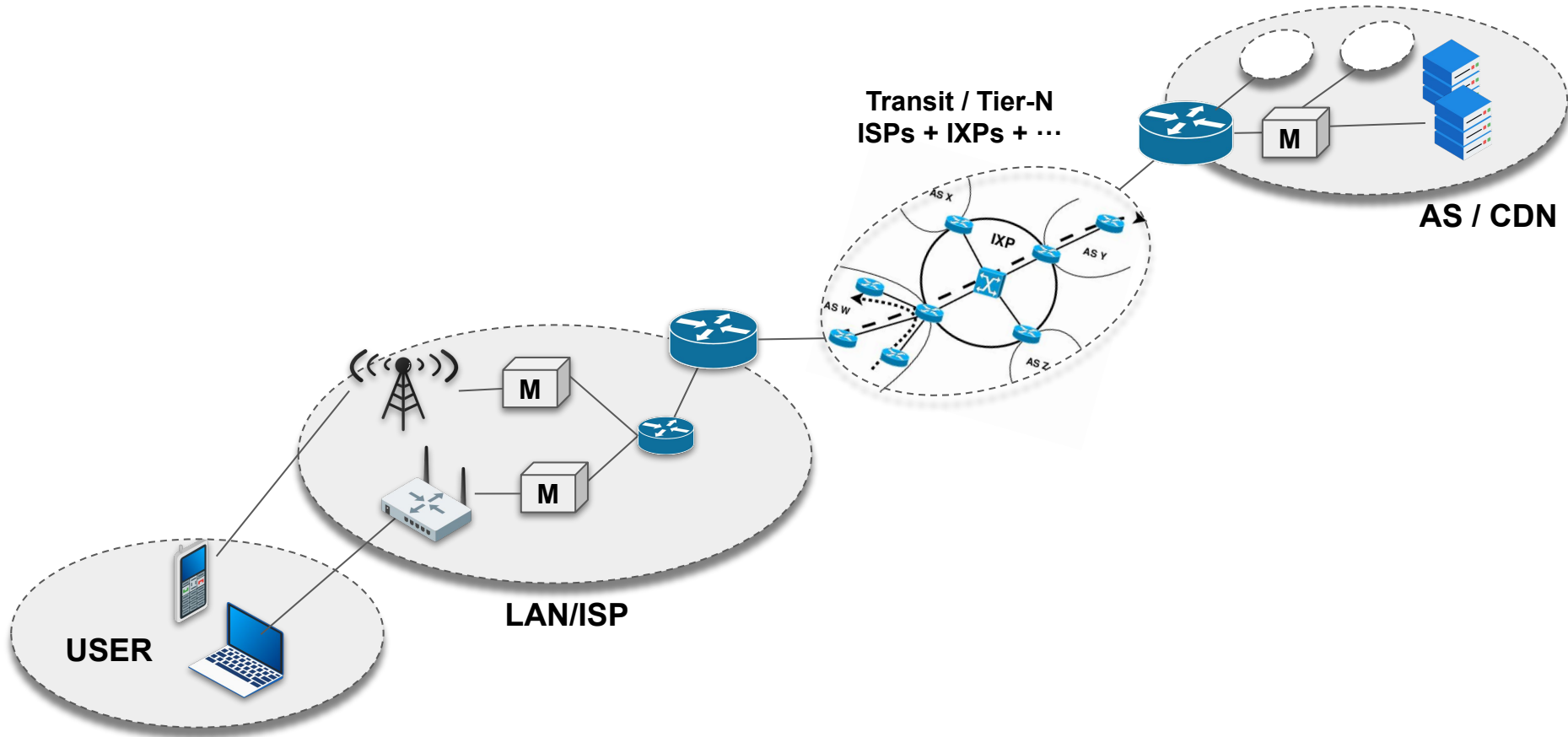
Tekla S. Perry is a senior editor at IEEE Spectrum.

2) "I didn't pay enough attention to security."

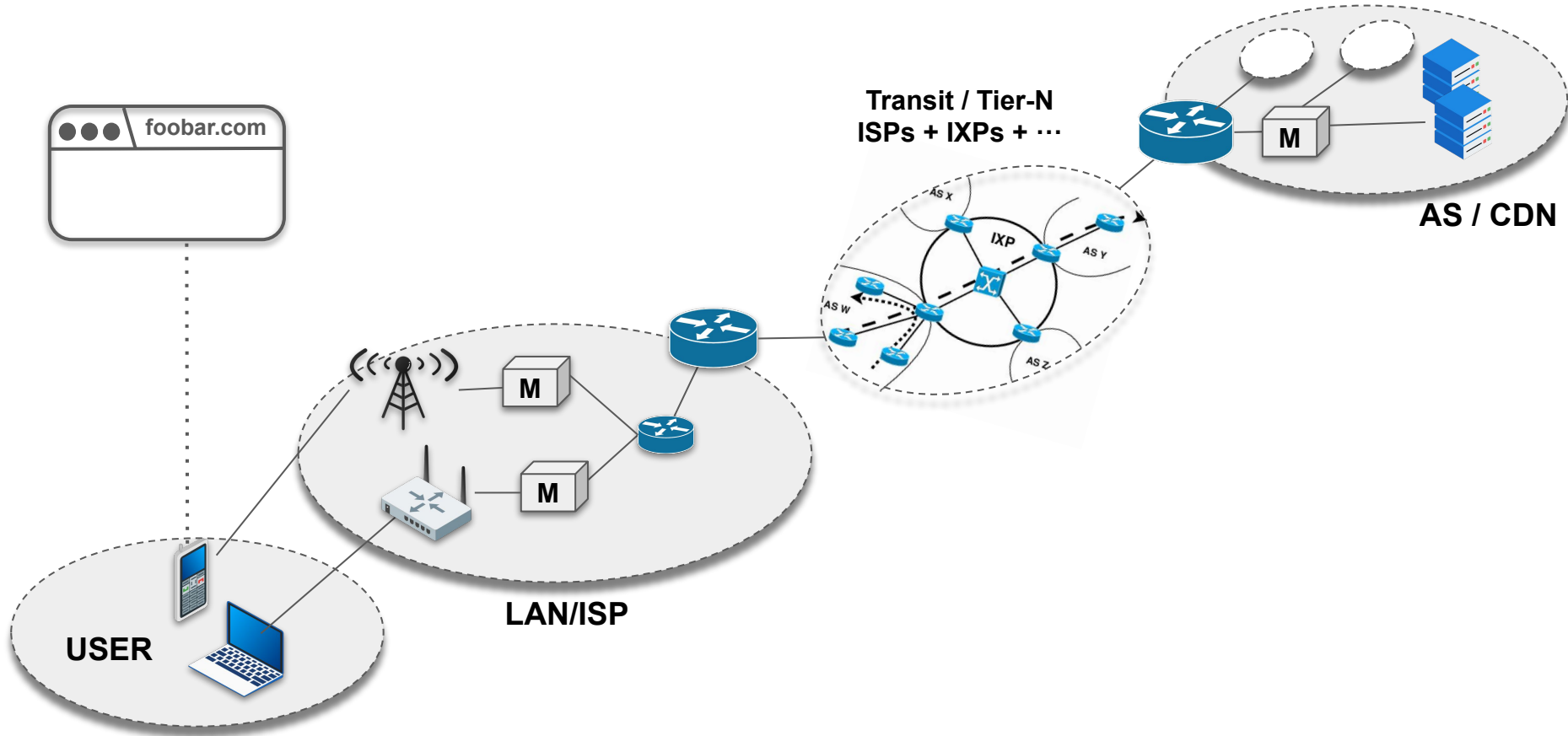
"Before public-key cryptography came around, key distribution was a really messy manual process," Cerf says. "It was awful, and it didn't scale. So that's why I didn't try to push that into the Internet. And by the time they did implement the RSA algorithm, I was well on my way to freezing the protocol, so I didn't push the crypto stuff. I still don't regret that, because graduate students, who were largely the people building and using the Internet, would be the last cohort of people I would rely on to maintain key discipline, though there are times when I wish we had put more end-to-end security in the system to begin with."



A simplified view to Internet communications: **HTTP + DNS**

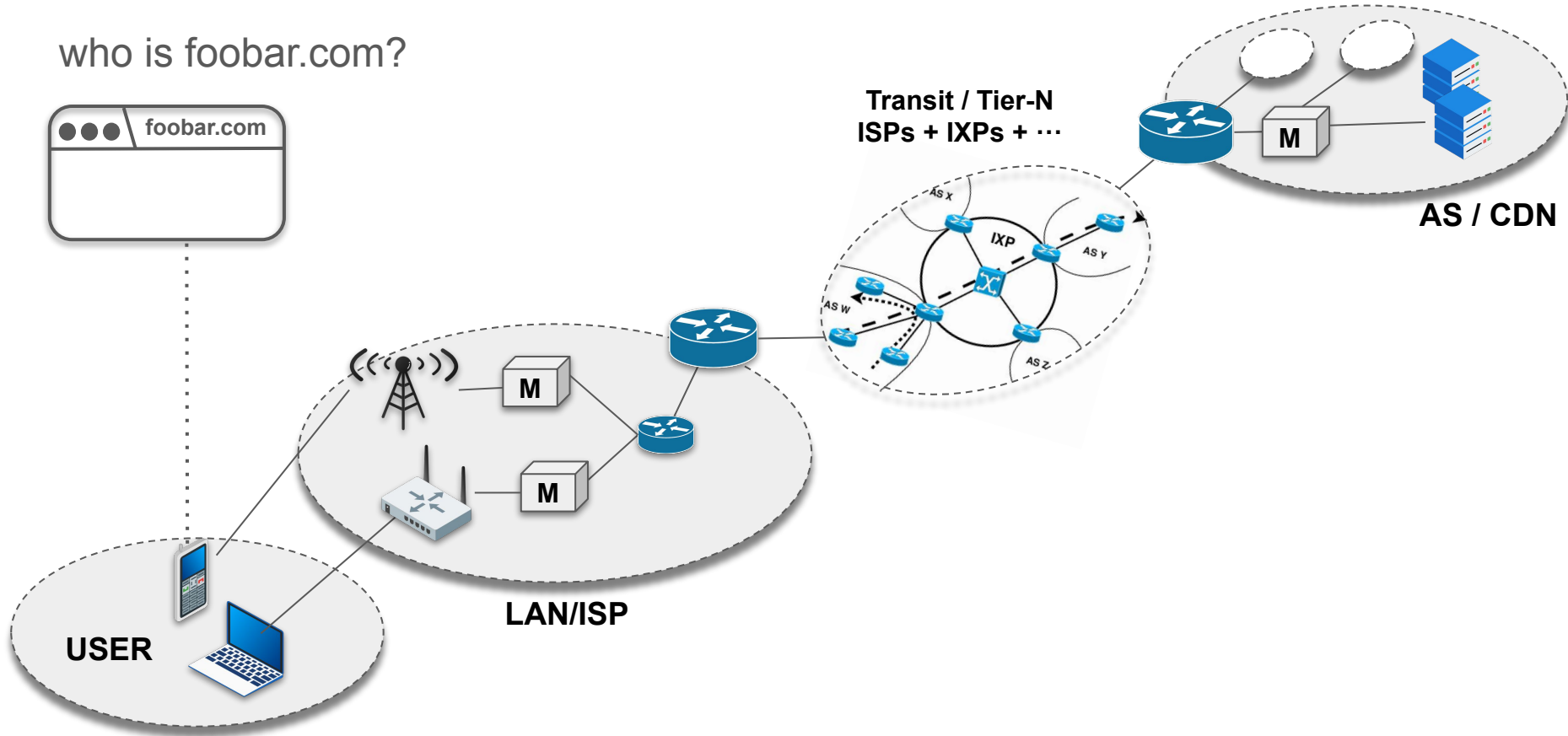


A simplified view to Internet communications: **HTTP + DNS**



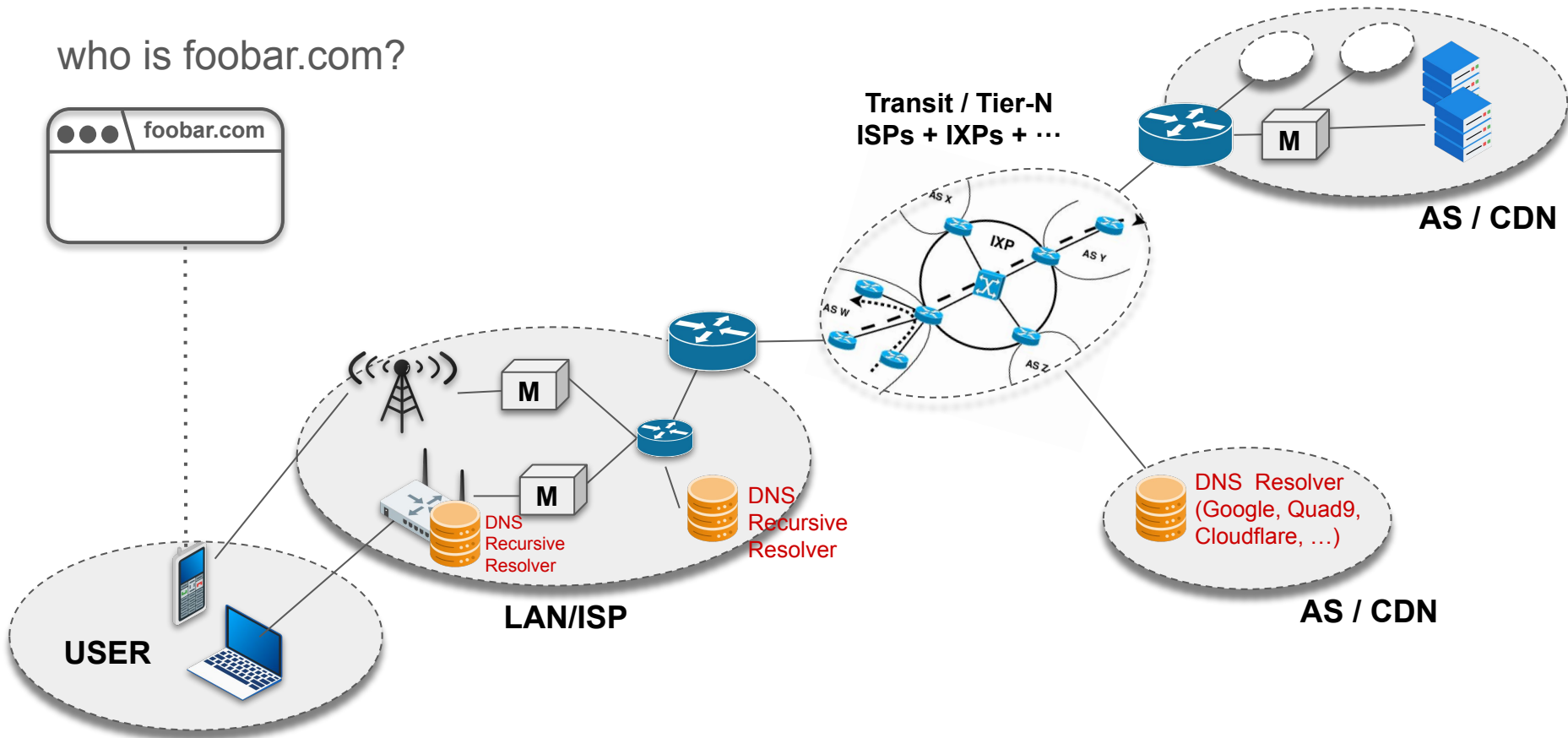
A simplified view to Internet communications: **HTTP + DNS**

who is foobar.com?



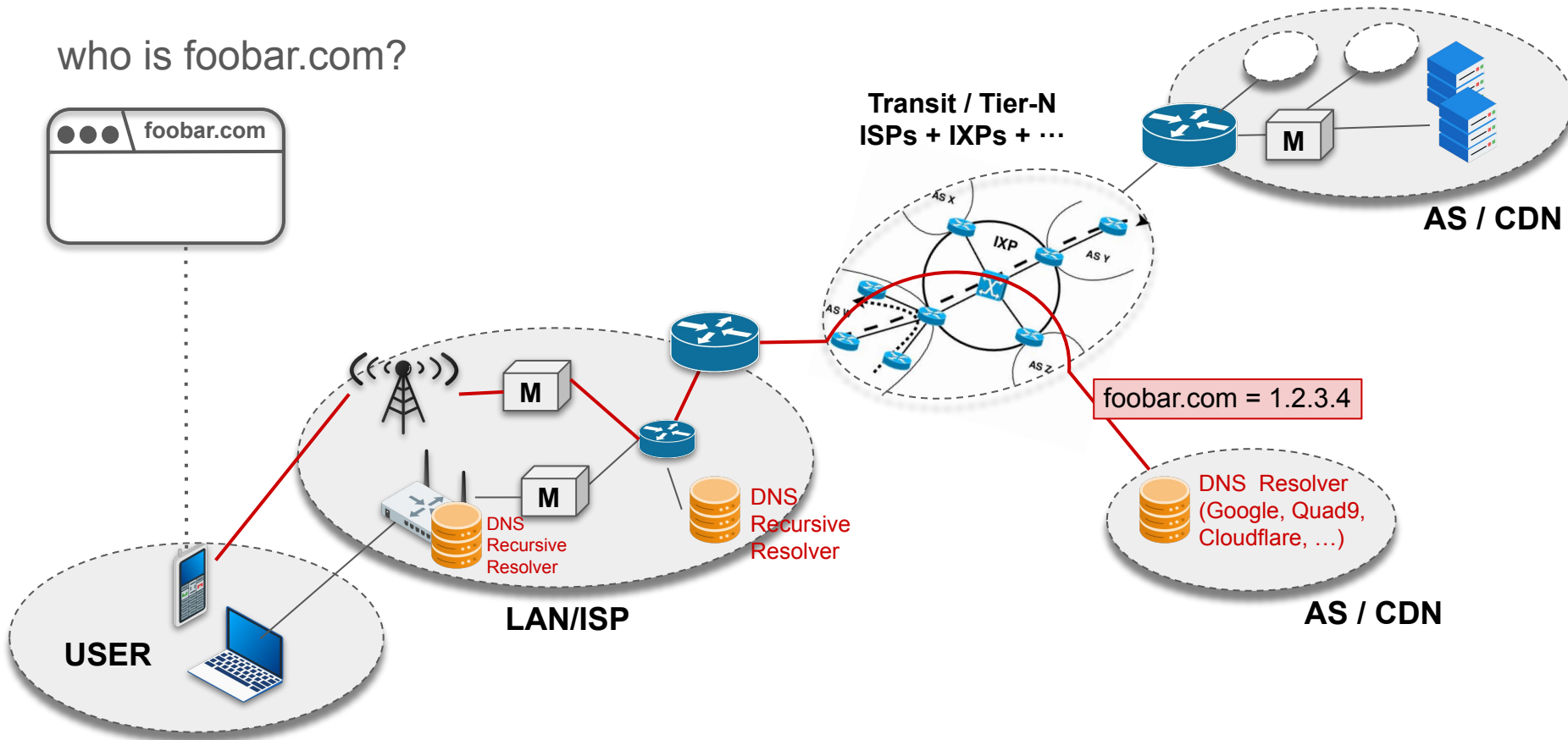
A simplified view to Internet communications: **HTTP + DNS**

who is foobar.com?



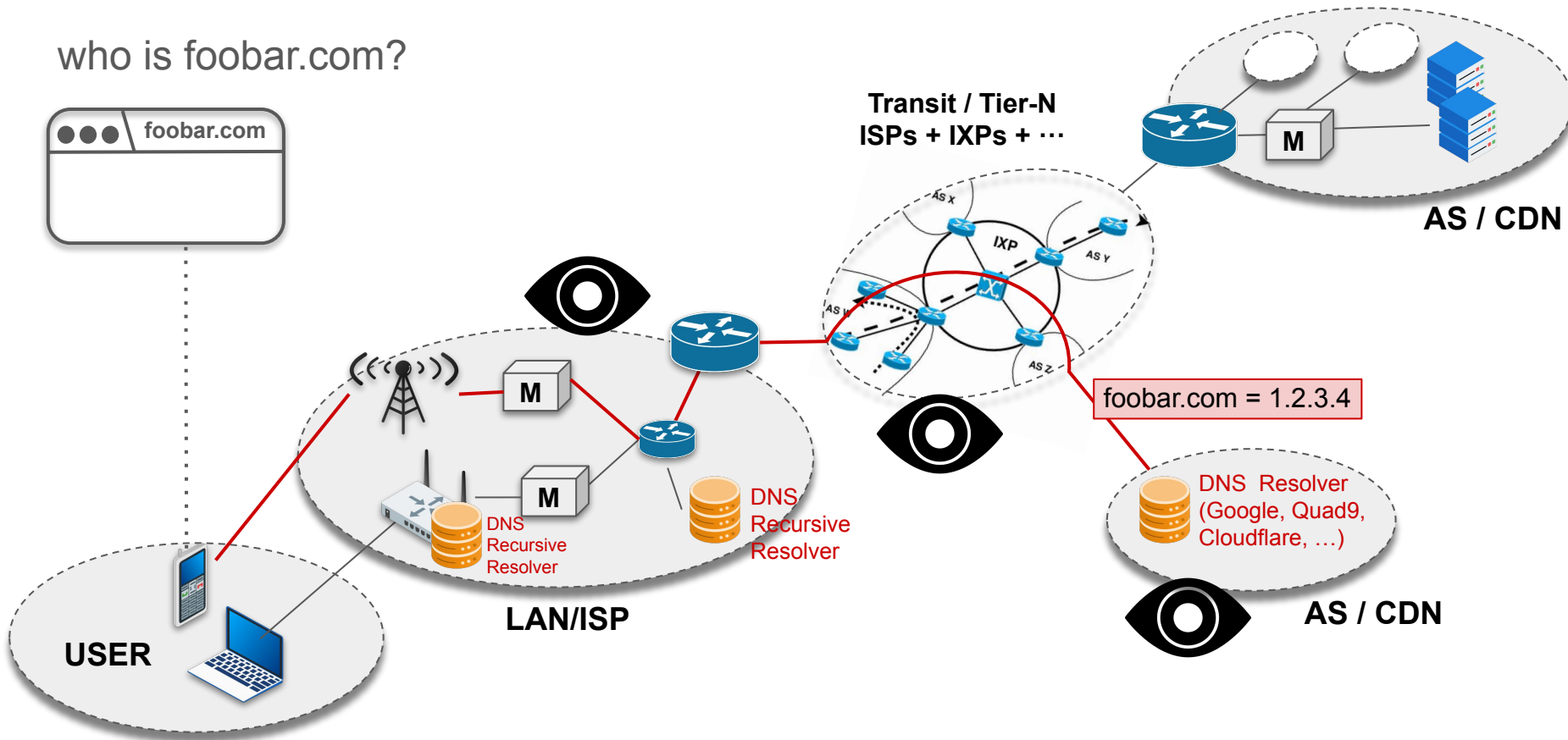
A simplified view to Internet communications: **HTTP + DNS**

who is foobar.com?

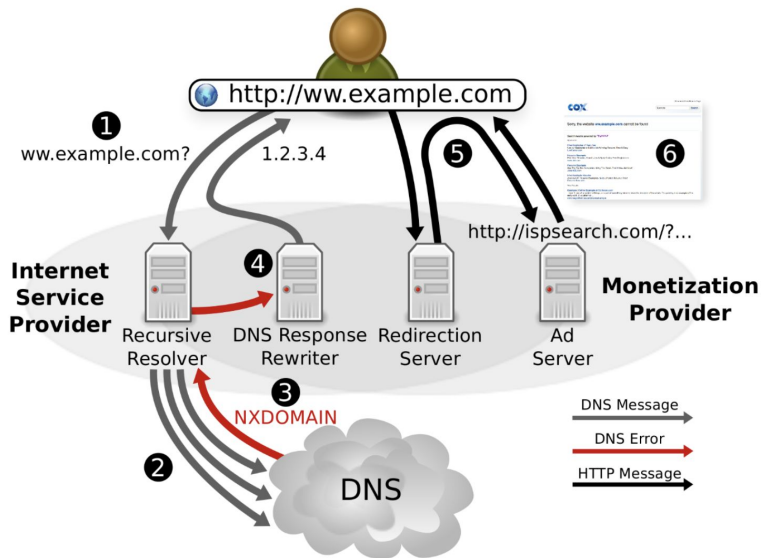


A simplified view to Internet communications: **HTTP + DNS**

who is foobar.com?



DNS is power: **ISP DNS redirection and injection**



USENIX Workshop on Free and Open Communications on the Internet (FOCI'11)

Redirecting DNS for Ads and Profit

Nicholas Weaver
ICSI

nweaver@icir.org

Christian Kreibich
ICSI

christian@icir.org

Vern Paxson
ICSI & UC Berkeley

vern@cs.berkeley.edu

Abstract

Internet Service Providers (ISPs) increasingly try to grow their profit margins by employing “error traffic monetization,” the practice of redirecting customers whose DNS lookups fail to advertisement-oriented Web servers. A small industry of companies provides the associated machinery for ISPs to engage in this monetization, with the companies often participating in operating the service as well. We conduct a technical analysis of DNS error traffic monetization evident in 66,000 *Natalyzr* sessions, including fingerprinting derived from patterns seen in the resulting ad landing pages. We identify major players in this industry, their ISP affiliations over time, and available user opt-out mechanisms. One monetization vendor, Paxfire, transgresses the error-based model and also reroutes all user search queries to Bing, Yahoo, and (sometimes) Google via proxy servers controlled or provided by Paxfire.

In the *ICSI Natalyzr* [8], our widely used network debugging and diagnostic tool,² we have employed tests for various forms of NXDOMAIN wildcarding since we started offering the service in mid-2009. In this paper we illuminate the DNS error monetization market by combining *Natalyzr*'s measurements with an analysis of the redirection pages collected between January 2010 and May 2011, the location and content of the ad servers, and the marketing material provided by the companies involved. We identify ISPs employing DNS error monetization, their choice of monetization provider (including shifts of provider and apparent in-house realization), potential redirection policy customizations, as well as availability of opt-out mechanisms.

We also observe a more aggressive form of DNS-driven traffic manipulation, *search-engine proxying*.

DNS is power: information controls and surveillance

USENIX Security '17

Global Measurement of DNS Manipulation

Paul Pearce[°] Ben Jones[†] Frank Li[°] Roya Ensafi[†]
Nick Feamster[‡] Nick Weaver[‡] Vern Paxson[°]

[°]University of California, Berkeley [†]Princeton University

[‡]International Computer Science Institute

{pearce, frankli, vern}@cs.berkeley.edu {bj6, rensafi, feamster}@cs.princeton.edu
nweaver@icsi.berkeley.edu

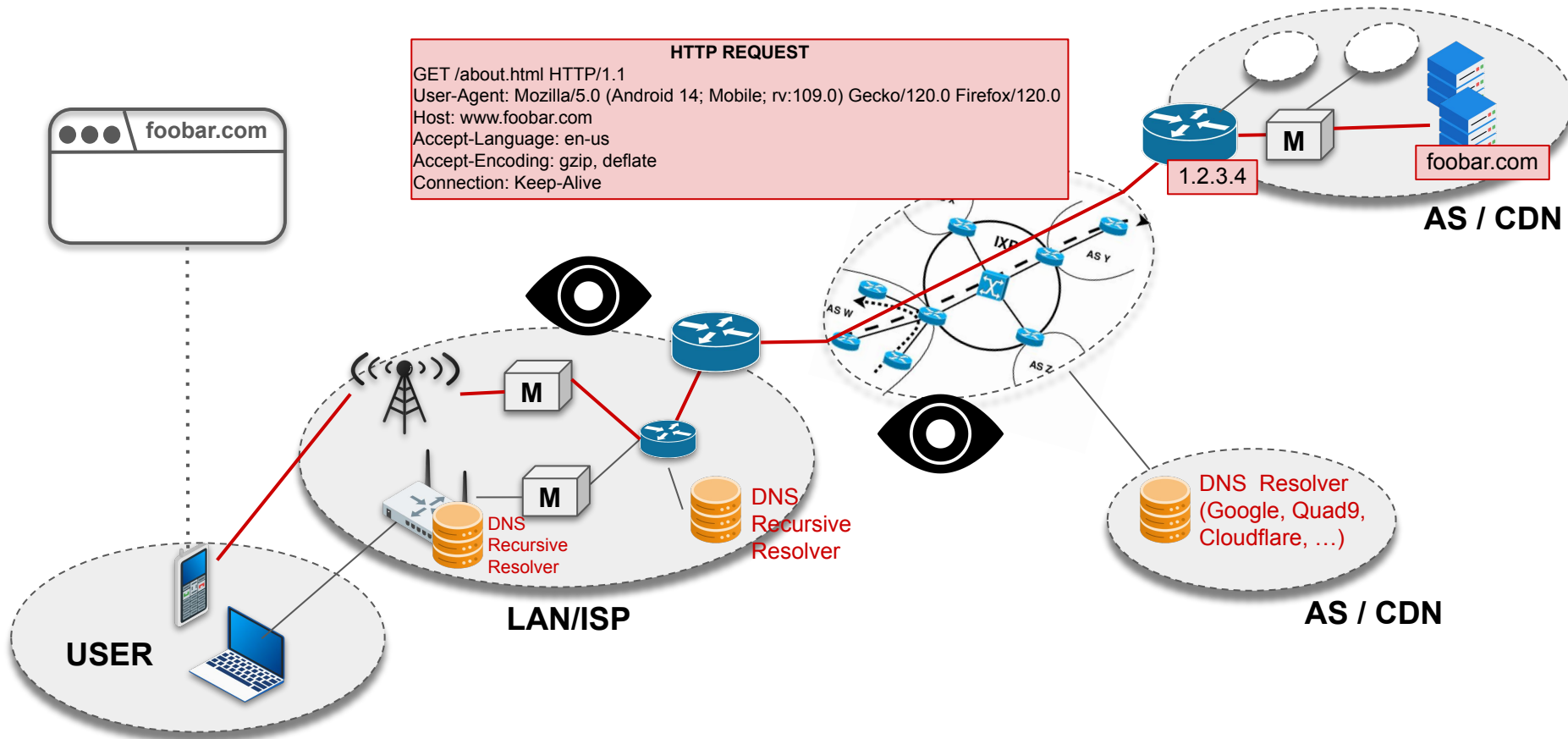
Abstract

Despite the pervasive nature of Internet censorship and the continuous evolution of how and where censorship is applied, measurements of censorship remain comparatively sparse. Understanding the scope, scale, and evolution of Internet censorship requires global measurements, performed at regular intervals. Unfortunately, the state of the art relies on techniques that, by and large, require users to directly participate in gathering these measurements, drastically limiting their coverage and inhibiting regular data collection. To facilitate large-scale measurements that can fill this gap in understanding, we develop Iris, a scalable, accurate, and ethical method to measure global manipulation of DNS resolutions. Iris reveals widespread DNS manipulation of many domain names; our findings both confirm anecdotal or limited results from previous work and reveal new patterns in DNS manipulation.

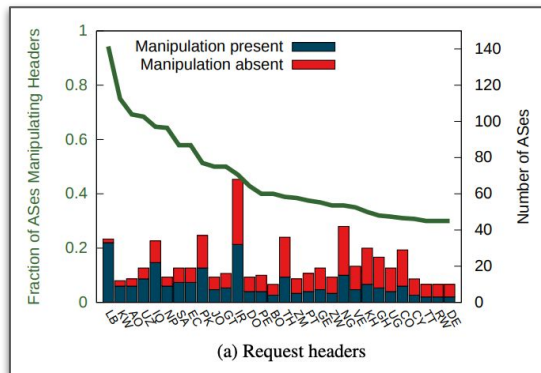
Country	Median Manipulated	Number Resolvers	Max Manipulated
Iran	6.02%	122	22.41%
China	5.22%	62	8.40%
Indonesia	0.63%	80	9.95%
Greece	0.28%	62	0.83%
Mongolia	0.17%	6	0.36%
Iraq	0.09%	7	5.79%
Bermuda	0.04%	2	0.09%
Kazakhstan	0.04%	14	3.90%
Belarus	0.04%	18	0.30%

Rank	Domain	Category	Countries
1	www.pokerstars.com	Gambling	19
2	betway.com	Gambling	19
3	pornhub.com	Pornography	19
4	youporn.com	Pornography	19
5	xvideos.com	Pornography	19
6	thepiratebay.org	P2P File Sharing	18
7	thepiratebay.se	P2P File Sharing	18
8	xhamster.com	Pornography	18
9	www.partypoker.com	Gambling	17
10	beeg.com	Pornography	17
80	torproject.org	Anonymity & Censorship	12
181	twitter.com	Twitter	9
250	www.youtube.com	Google Video	8
495	www.citizenlab.org	Freedom of Expression	4
606	www.google.com	Google	3

A simplified view to Internet communications: **HTTP + DNS**



HTTP header manipulation



Exploring HTTP Header Manipulation In-The-Wild

Gareth Tyson
Queen Mary University of
London
gareth.tyson@qmul.ac.uk

Ignacio Castro
Queen Mary University of
London
i.castro@qmul.ac.uk

Shan Huang
Queen Mary University of
London
shan.huang@qmul.ac.uk

Vasile C. Perta
Sapienza University of Rome
perta@di.uniroma1.it

Felix Cuadrado
Queen Mary University of
London
felix.cuadrado@qmul.ac.uk

Arjuna Sathiseelan
University of Cambridge
arjuna@cl.cam.ac.uk

Steve Uhlig
Queen Mary University of
London
steve.uhlig@qmul.ac.uk

ABSTRACT

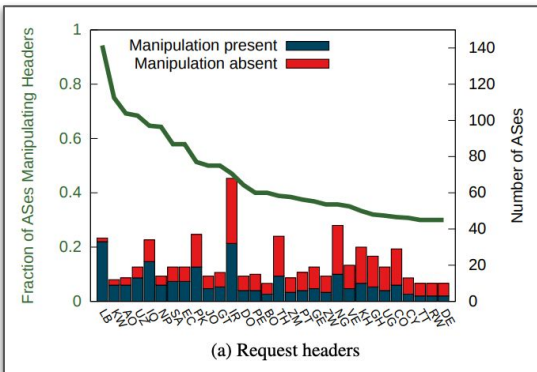
Headers are a critical part of HTTP, and it has been shown that they are increasingly subject to middlebox manipulation. Although this is well known, little is understood about the general regional and network trends that underpin these manipulations. In this paper, we collect data on thousands of networks to understand how they intercept HTTP headers in-the-wild. Our analysis reveals that 25% of measured ASes modify HTTP headers. Beyond this, we witness distinct trends among different regions and AS types; e.g., we observe high numbers of cache headers in poorly connected regions. Finally, we perform an in-depth analysis of the types of manipulations and how they differ across regions.

measurement platform using the Hola peer-to-peer proxy network [2] (§3). Using this platform, we craft and forward HTTP requests via third party networks to a web server we control. By monitoring both the request and response endpoints, we can discover manipulations performed by these networks. Exploiting Hola, we launch over 400k HTTP queries from 143k vantage points in 3818 Autonomous Systems (ASes) — one of the largest studies of its kind. Unlike techniques using controlled infrastructures (e.g., Planetlab), this provides unique visibility on a range of network types in countries rarely studied, e.g., over 400 ASes in Africa (§4).

In this paper we explore the propensity of different network types and regions to manipulate HTTP headers, in terms of both frequency (§5), and content (§6). We find that header manipulation

Header Type	#Headers in Category		Total #Headers injected/modified	
	Request	Response	Request	Response
Cache	4	9	8419	3799
Operational	12	9	5090	63
Feature	8	3	639	1884
Information	1	5	20	20
Unknown	4	3	10	41

HTTP header manipulation



Header Type	#Headers in Category		Total #Headers injected/modified	
	Request	Response	Request	Response
Cache	4	9	8419	
Operational	12	9	5090	
Feature	8	3	639	
Information	1	5	20	
Unknown	4	3	10	

6.4 Information Headers

Information headers contain metadata that describes the client or server. Information headers are rarely seen in the data, with a slightly higher propensity to see them in developed regions: NA and EU. An interesting example is the User-Agent header, which informs the server of the type of browser requesting the page. We find 15 ASes manipulating this, and downgrading the browser version, e.g., from Firefox 5.0 to 4.0. We even see 378 IP addresses where the HTTP version is downgraded to 1.0 (from 1.1). In 82% of the samples, these requests had passed through a Squid proxy. Worryingly, we often see old middlebox software: 34% of Squid samples are running version 2.7 or older (last updated 2010). We even find 22 ASes using Squid software that has not been updated for at least a decade (v2.5). These are overwhelmingly in countries that rank lowly in the Web Index; apart from two ASes in Australia and Belgium, the highest ranked country is 32nd (Czech Republic).

Finally, we observed 28 responses in which a Set-Cookie header was injected. A Croatian AS was responsible for 8 of these, likely part of monitoring or customer tracking [5, 4]. There were a further 20 samples that had cookies returned due to interceptions by various other types of middleboxes (e.g., Netscaler, Cisco Access Control). This actually highlights a particularly worrying feature of Hola, as it allows users to obtain the cookie identifiers of others.

6.5 Unknown Headers

It is worth briefly noting that we could not conclusively classify a number of headers: X-Client-TOS (4 ASes), SFID, X-TMV-Type (2 ASes), X-DG-TaggedAs, X-IMForwards (1 AS) and the enigmatic - - - - - (1 AS). The fact that no public documentation exists perhaps indicates that notable subsets of HTTP can no longer be considered “standard”. The region with the greatest proportion of these is AF, although they also occur in NA and AS.

WWW '17

Header Manipulation In-The-Wild

Shan Huang
Queen Mary University of London
shuang@qmul.ac.uk

Felix Cuadrado
Queen Mary University of London
felix.cuadrado@qmul.ac.uk

Vasile C. Perta
University of Rome
vperta@di.uniroma1.it

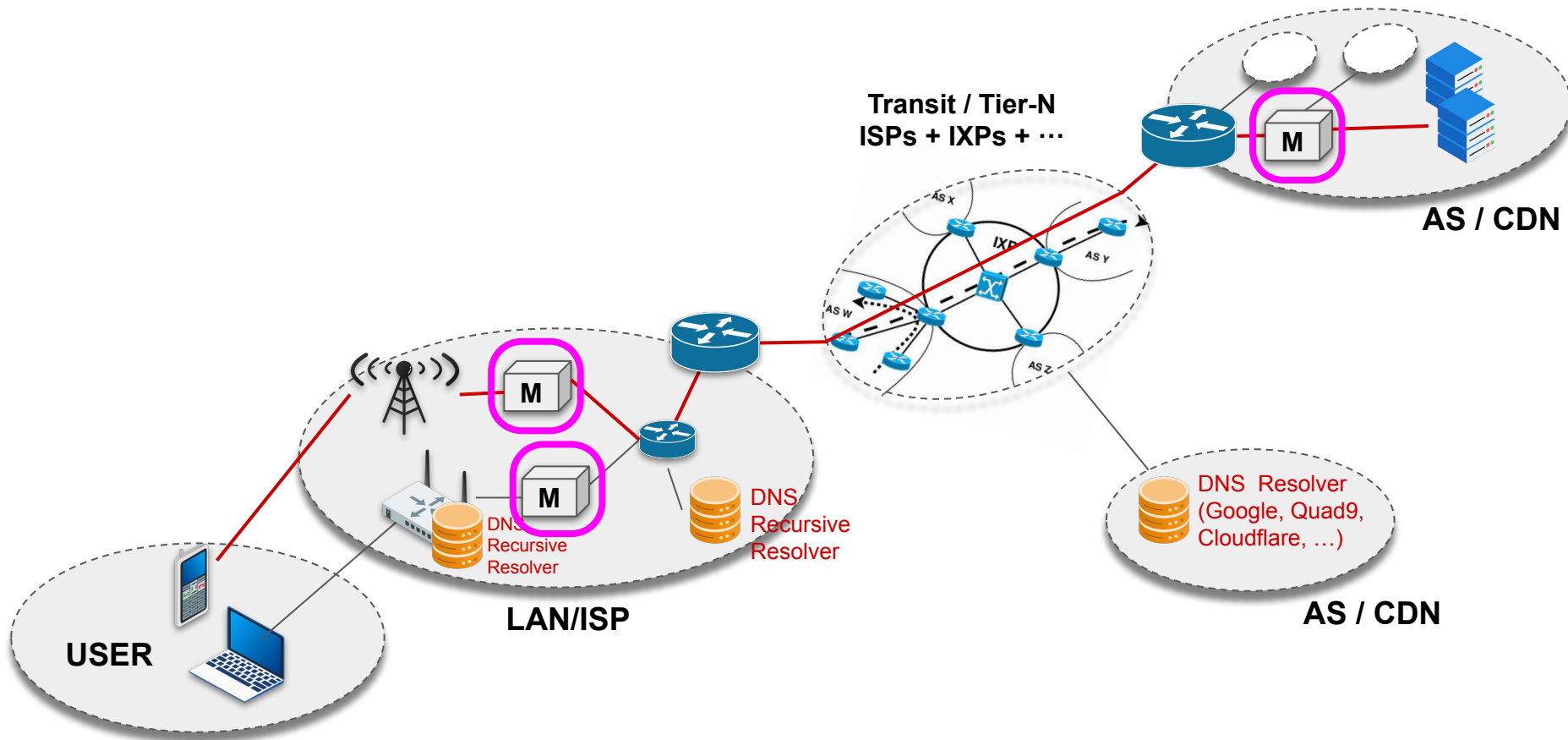
Arjuna Sathiseelan
University of Cambridge
arjuna@cl.cam.ac.uk

Steve Uhlig
Queen Mary University of London
s.uhlig@qmul.ac.uk

measurement platform using the Hola peer-to-peer proxy network [2] (§3). Using this platform, we craft and forward HTTP requests via third party networks to a web server we control. By monitoring both the request and response endpoints, we can discover manipulations performed by these networks. Exploiting Hola, we launch over 400k HTTP queries from 143k vantage points in 3818 Autonomous Systems (ASes) — one of the largest studies of its kind. Unlike techniques using controlled infrastructures (e.g., Planetlab), this provides unique visibility on a range of network types in countries rarely studied, e.g., over 400 ASes in Africa (§4). In this paper we explore the propensity of different network types and regions to manipulate HTTP headers, in terms of both frequency (§5), and content (§6). We find that header manipulation

Middleboxes

Everywhere



Netalyzr: Illuminating The Edge Network

Christian Kreibich
ICSI
1947 Center Street
Berkeley, CA, 94704, USA
christian@icir.org

Boris Nechaev
HIIT & Aalto University
PO Box 19800
00076 Aalto, Finland
boris.nechaev@hiit.fi

Nicholas Weaver
ICSI
1947 Center Street
Berkeley, CA, 94704, USA
nweaver@icsi.berkeley.edu

Vern Paxson
ICSI & UC Berkeley
1947 Center Street
Berkeley, CA, 94704, USA
vern@cs.berkeley.edu

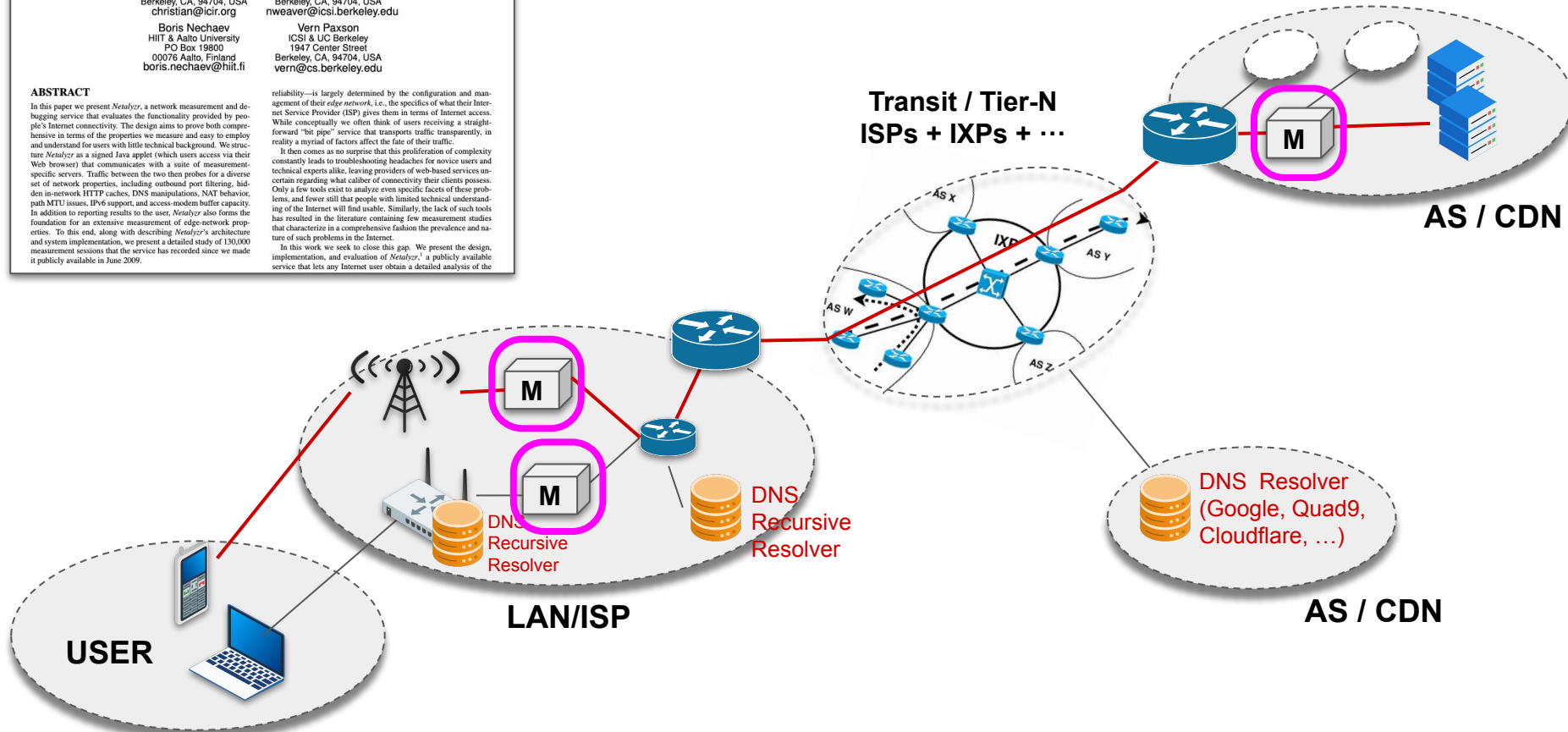
ABSTRACT

In this paper we present *Netalyzr*, a network measurement and debugging service that evaluates the functionality provided by people's Internet connectivity. The design aims to prove both comprehensive in terms of the properties we measure and easy to employ and understand for users with little technical background. We structure *Netalyzr* as a signed Java applet (which users access via their Web browser) that communicates with a suite of measurement-specific servers. Traffic between the two then probes for a diverse set of network properties, including outbound port filtering, hidden in-network HTTP caches, DNS manipulations, NAT behavior, path MTU issues, IPv6 support, and access-modem buffer capacity. In addition to reporting results to the user, *Netalyzr* also forms the foundation for an extensive measurement of edge-network properties. To this end, along with describing *Netalyzr*'s architecture and system implementation, we present a detailed study of 130,000 measurement sessions that the service has recorded since we made it publicly available in June 2009.

reliability—is largely determined by the configuration and management of their *edge network*, i.e., the specifics of what their Internet Service Provider (ISP) gives them in terms of Internet access. While conceptually we often think of users receiving a straightforward “bit pipe” service that transports traffic transparently, in reality a myriad of factors affect the fate of their traffic.

It then comes as no surprise that this proliferation of complexity constantly leads to troubleshooting headaches for novice users and technical experts alike, leaving providers of web-based services uncertain regarding what caliber of connectivity their clients possess. Only a few tools exist to analyze even specific facets of these problems, and fewer still that people with limited technical understanding of the Internet will find usable. Similarly, the lack of such tools has resulted in the literature containing few measurement studies that characterize in a comprehensive fashion the prevalence and nature of such problems in the Internet.

In this work we seek to close this gap. We present the design, implementation, and evaluation of *Netalyzr*, a publicly available service that lets any Internet user obtain a detailed analysis of the



Netalyzer: Illuminating The Edge Network

Christian Kreibich
ICSI
1947 Center Street
Berkeley, CA, 94704, USA
christian@icir.org

Boris Nechaev
HIIT & Aalto University
PO Box 19800
00076 Aalto, Finland
boris.nechaev@hiit.fi

Nicholas Weaver
ICSI
1947 Center Street
Berkeley, CA, 94704, USA
nweaver@icsi.berkeley.edu

Vern Paxson
ICSI & UC Berkeley
1947 Center Street
Berkeley, CA, 94704, USA
vern@cs.berkeley.edu

ABSTRACT

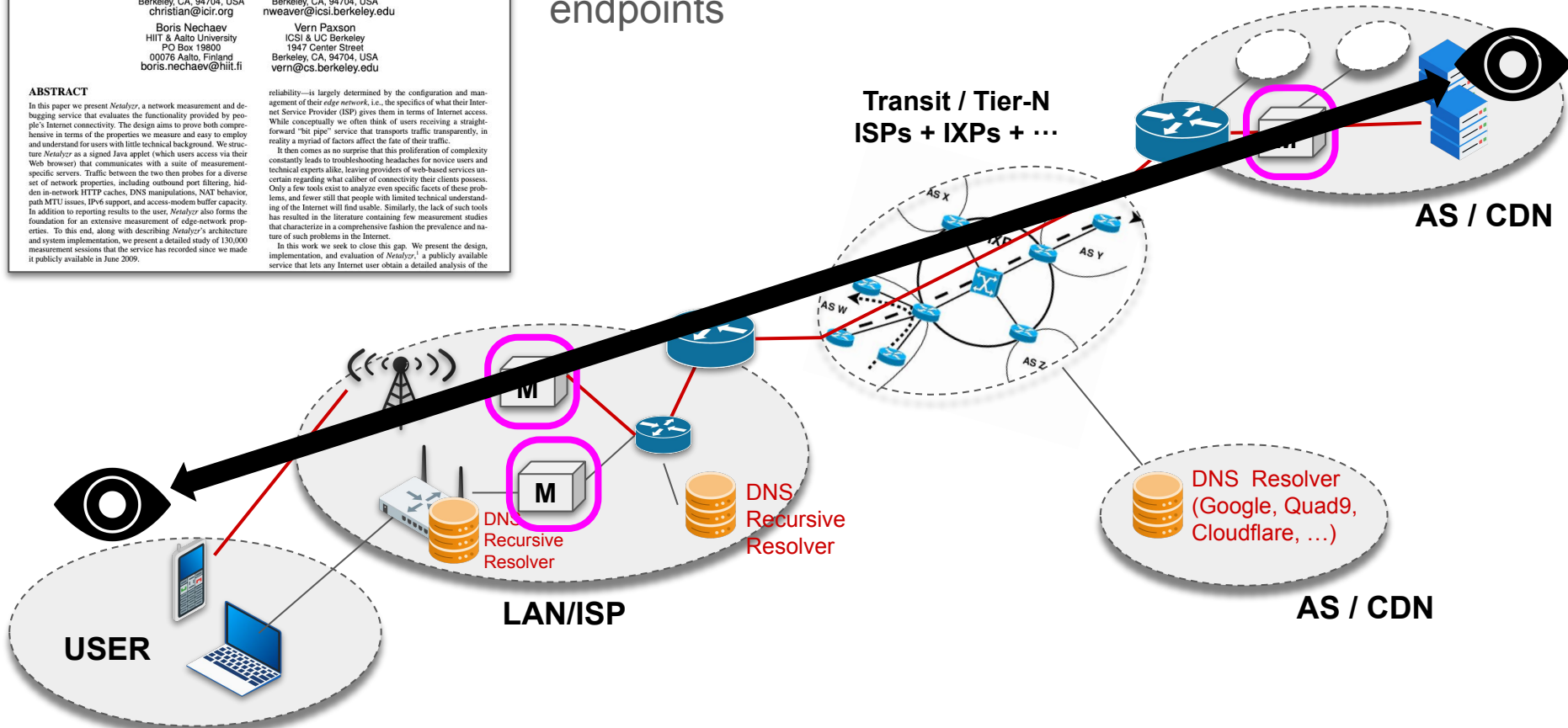
In this paper we present *Netalyzer*, a network measurement and debugging service that evaluates the functionality provided by people's Internet connectivity. The design aims to prove both comprehensive in terms of the properties we measure and easy to employ and understand for users with little technical background. We structure *Netalyzer* as a signed Java applet (which users access via their Web browser) that communicates with a suite of measurement-specific servers. Traffic between the two then probes for a diverse set of network properties, including outbound port filtering, hidden in-network HTTP caches, DNS manipulations, NAT behavior, path MTU issues, IPv6 support, and access-modem buffer capacity. In addition to reporting results to the user, *Netalyzer* also forms the foundation for an extensive measurement of edge-network properties. To this end, along with describing *Netalyzer*'s architecture and system implementation, we present a detailed study of 130,000 measurement sessions that the service has recorded since we made it publicly available in June 2009.

reliability—is largely determined by the configuration and management of their *edge network*, i.e., the specifics of what their Internet Service Provider (ISP) gives them in terms of Internet access. While conceptually we often think of users receiving a straightforward “bit pipe” service that transports traffic transparently, in reality a myriad of factors affect the fate of their traffic.

It then comes as no surprise that this proliferation of complexity constantly leads to troubleshooting headaches for novice users and technical experts alike, leaving providers of web-based services uncertain regarding what caliber of connectivity their clients possess. Only a few tools exist to analyze even specific facets of these problems, and fewer still that people with limited technical understanding of the Internet will find usable. Similarly, the lack of such tools has resulted in the literature containing few measurement studies that characterize in a comprehensive fashion the prevalence and nature of such problems in the Internet.

In this work we seek to close this gap. We present the design, implementation, and evaluation of *Netalyzer*, a publicly available service that lets any Internet user obtain a detailed analysis of the

Finding middleboxes is relatively “easy” if you control both endpoints



Netalyzer: Illuminating The Edge Network

Christian Kreibich
ICSI
1947 Center Street
Berkeley, CA, 94704, USA
christian@icsi.org

Boris Nechaev
HIIT & Aalto University
PO Box 19800
00076 Aalto, Finland
boris.nechaev@hiit.fi

Nicholas Weaver
ICSI
1947 Center Street
Berkeley, CA, 94704, USA
nweaver@icsi.berkeley.edu

Vern Paxson
ICSI & UC Berkeley
1947 Center Street
Berkeley, CA, 94704, USA
vern@cs.berkeley.edu

ABSTRACT

In this paper we present *Netalyzer*, a network measurement and debugging service that evaluates the functionality provided by people's Internet connectivity. The design aims to prove both comprehensive in terms of the properties we measure and easy to employ and understand for users with little technical background. We structure *Netalyzer* as a signed Java applet (which users access via their Web browser) that communicates with a suite of measurement-specific servers. Traffic between the two then probes for a diverse set of network properties, including outbound port filtering, hidden in-network HTTP caches, DNS manipulations, NAT behavior, path MTU issues, IPv6 support, and access-remote buffer capacity. In addition to reporting results to the user, *Netalyzer* also forms the foundation for an extensive measurement of edge-network properties. To this end, along with describing *Netalyzer*'s architecture and system implementation, we present a detailed study of 130,000 measurement sessions that the service has recorded since we made it publicly available in June 2009.

reliability—is largely determined by the configuration and management of their *edge network*, i.e., the specifics of what their Internet Service Provider (ISP) gives them in terms of Internet access. While conceptually we often think of users receiving a straightforward “bit pipe” service that transports traffic transparently, in reality a myriad of factors affect the fate of their traffic.

It then comes as no surprise that this proliferation of complexity constantly leads to troubleshooting headaches for novice users and technical experts alike, leaving providers of web-based services uncertain regarding what caliber of connectivity their clients possess. Only a few tools exist to analyze even specific facets of these problems, and fewer still that people with limited technical understanding of the Internet will find usable. Similarly, the lack of such tools has resulted in the literature containing few measurement studies that characterize in a comprehensive fashion the prevalence and nature of such problems in the Internet.

In this work we seek to close this gap. We present the design, implementation, and evaluation of *Netalyzer*, a publicly available service that lets any Internet user obtain a detailed analysis of the

CGNs (Carrier-Grade NATs)

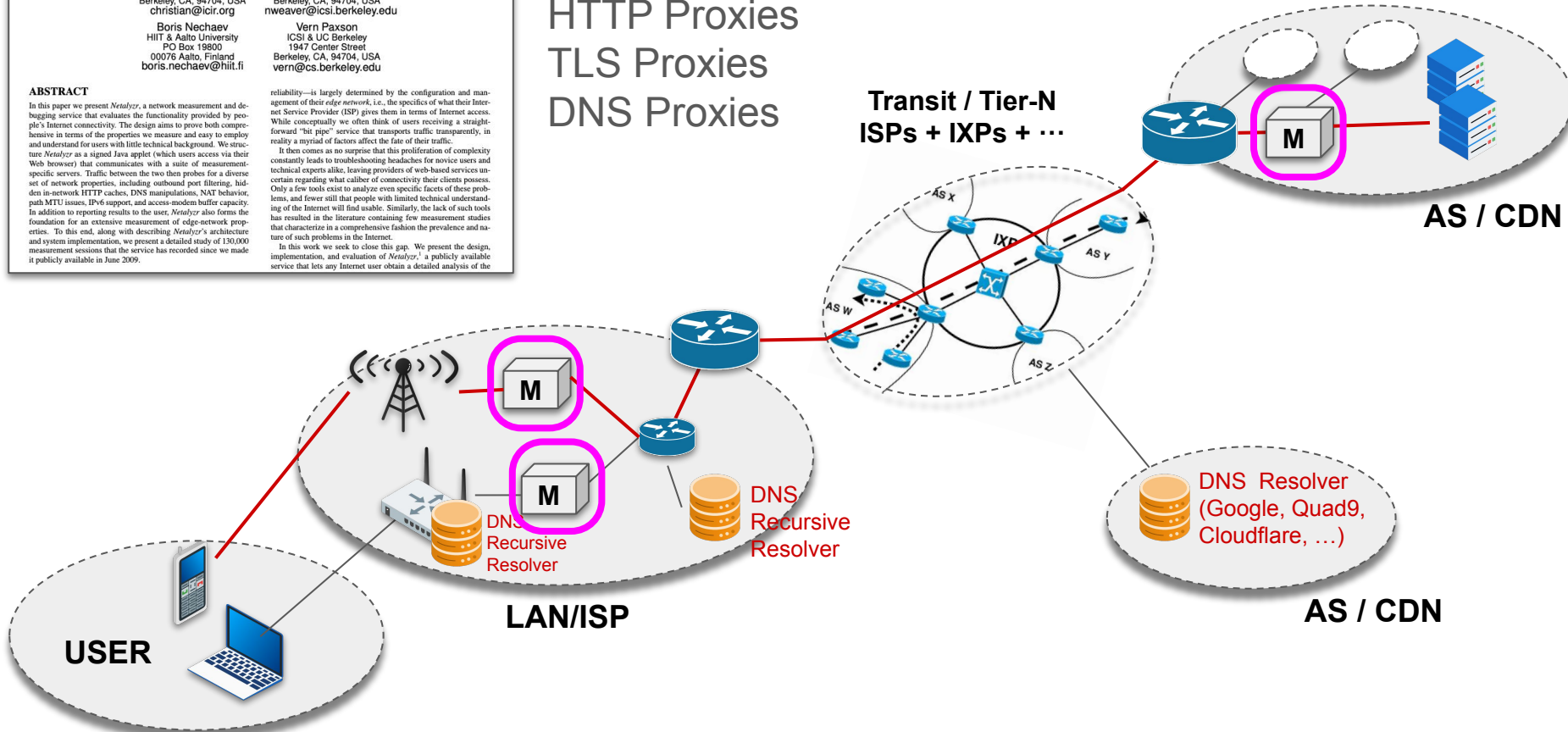
TCP Splitting Proxies

HTTP Proxies

TLS Proxies

DNS Proxies

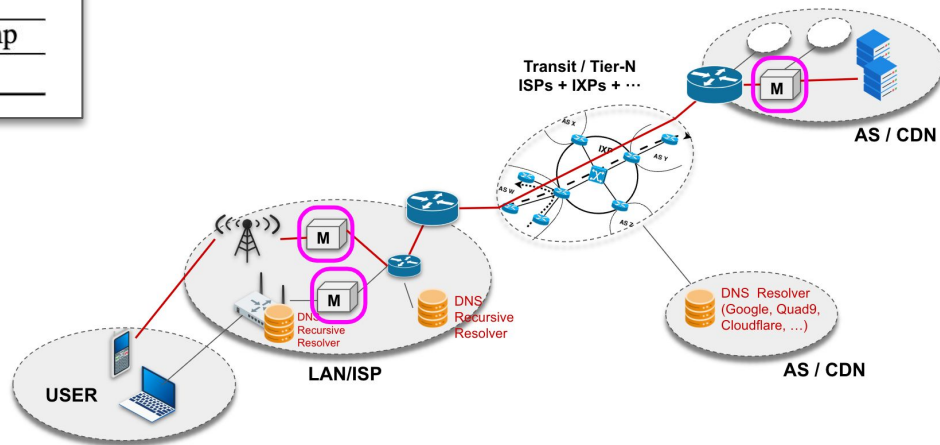
Transit / Tier-N
ISPs + IXPs + ...



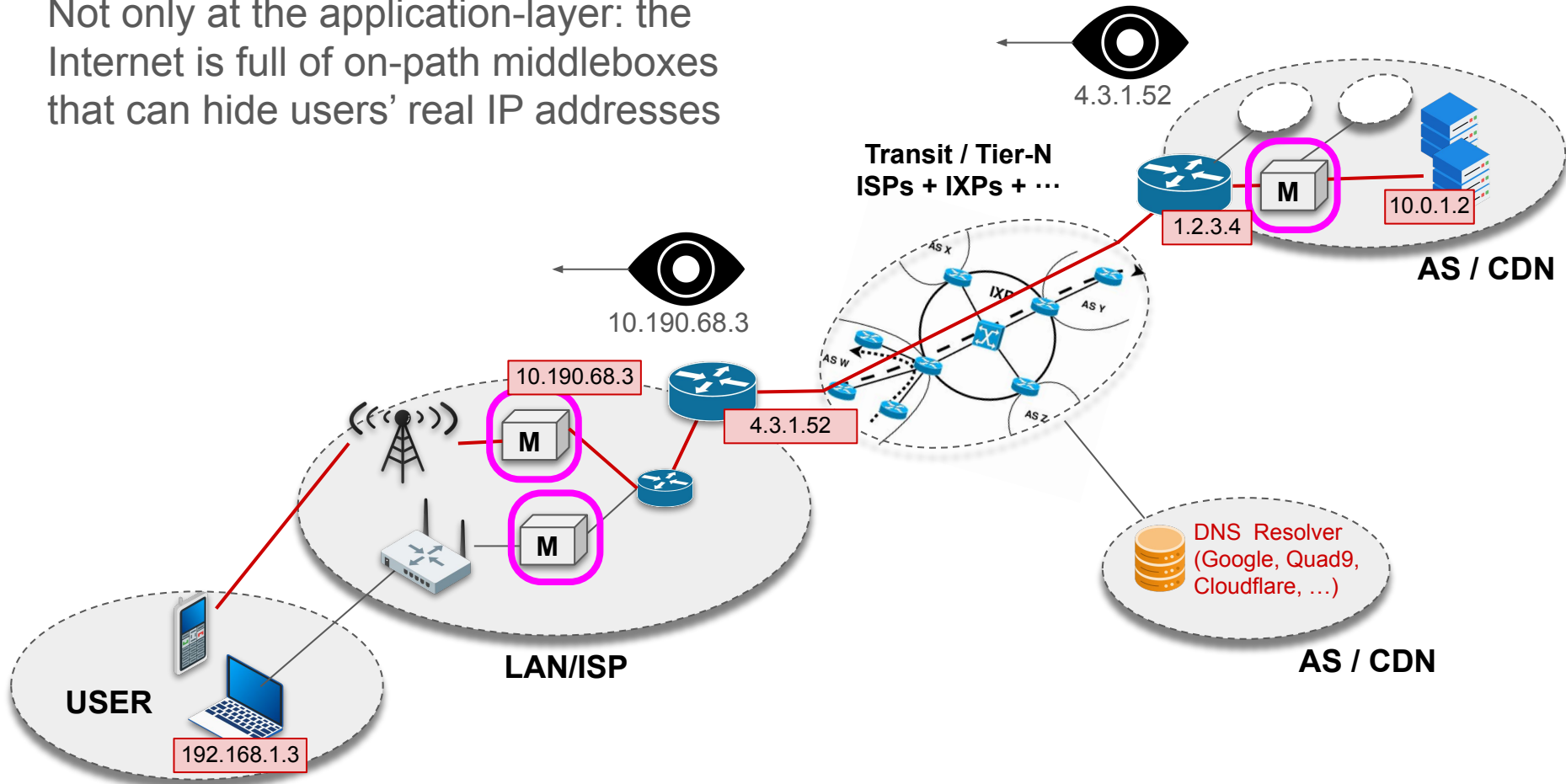
In-path HTTP Proxies can inject **sensitive user data** and **unique identifiers** (perma-cookies) in clear-text HTTP traffic that deanonymize the user


HTTP Header	Operators	Notes
x-up-calling-line-id	Vodacom (ZA)	Phone #
x-up-nai		
x-up-vodacomgw-subid		
msisdn	Orange (JO)	MSISDN
x-nokia-msisdn	Smart (PH)	
tm_user-id	Movistar (ES)	Subscriber ID
x-up-subno		
x-up-3gpp-imei	Vodacom (ZA)	IMEI
lbs-eventtime	Smartone (HK)	Timestamp
lbs-zoneid	Smartone (HK)	Location

HTTP Header	Operator
x-acr	AT&T (US)
x-amobee-1	Airtel (IN)
x-amobee-2	Singtel (SG)
x-uidh	Verizon (US)
x-vf-acr	Vodacom (ZA), Vodafone (NL)



Not only at the application-layer: the Internet is full of on-path middleboxes that can hide users' real IP addresses



ABOUT
EUROPOL

OPERATIONS, SERVICES &
INNOVATION


CRIME
AREAS


PARTNERS &
COLLABORATION


CAREERS &
PROCUREMENT

MEDIA &
PRESS

PUBLICATIONS &
EVENTS

SEARCH

CONTACT

LANGUAGE

Home / Media & Press

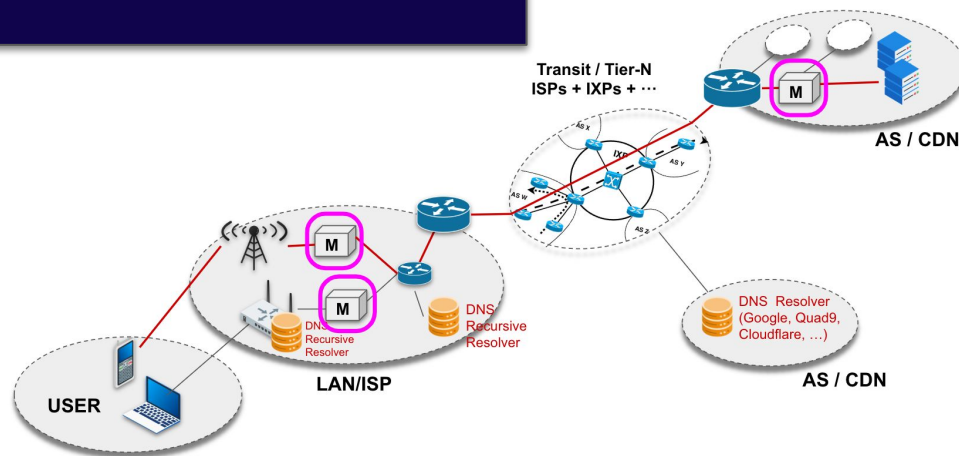
13 October 2017

NEWS

Are you sharing the same IP address as a criminal? Law enforcement call for the end of Carrier Grade NAT (CGN) to increase accountability online

Europol and the Estonian Presidency of the EU Council address the serious online capability gap in law enforcement efforts to investigate and attribute crime created by CGN technologies.

What if your phone or WiFi AP is a VPN server?

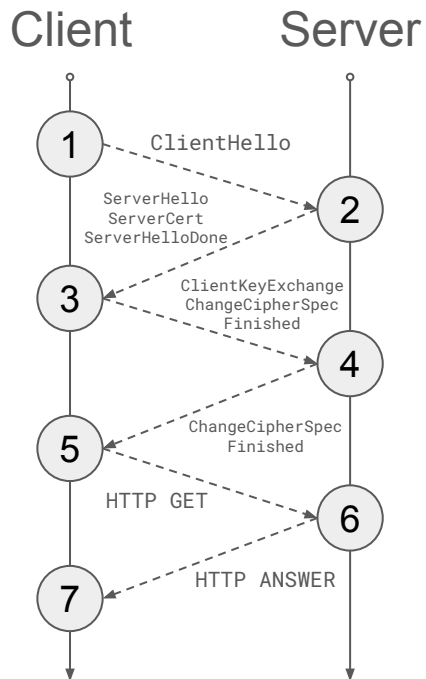


The “good” news

TLS

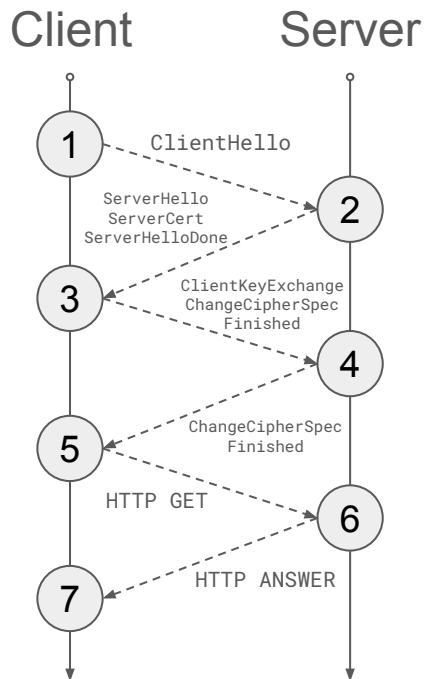
Securing end-to-end communications: **TLS**

TLS 1.2 Handshake

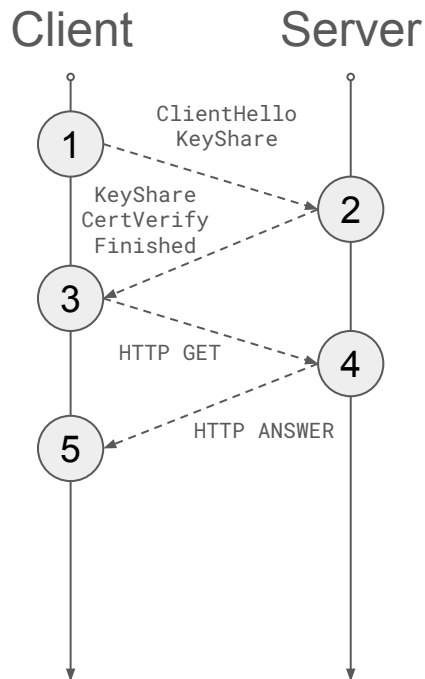


Securing end-to-end communications: **TLS**

TLS 1.2 Handshake

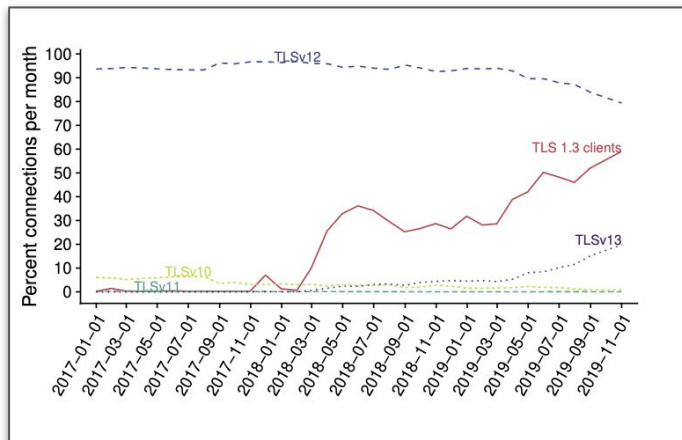
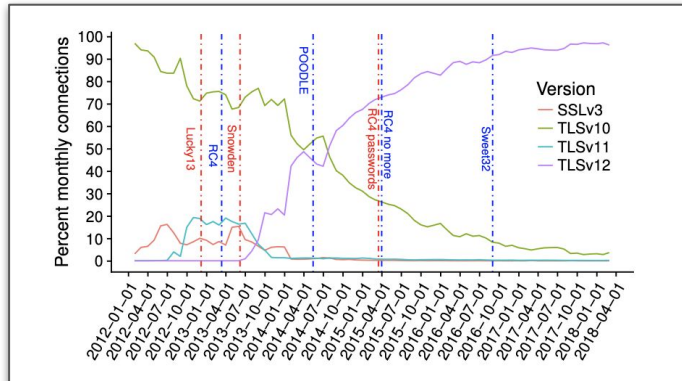


TLS 1.3 Handshake



Securing end-to-end communications: TLS adoption

SSL/TLS Version	Release Date
SSL 2	Feb. 1995
SSL 3	Nov. 1996
TLS 1.0	Jan. 1999
TLS 1.1	Apr. 2006
TLS 1.2	Aug. 2008
TLS 1.3	Aug. 2018



Tracking the deployment of TLS 1.3 on the Web: A story of experimentation and centralization

Ralph Holt^{1,2}, Jens Hiller³, Johanna Amann⁴, Abbas Razaghpanah⁴, Thomas Jost³, Narseo Vallina-Rodriguez⁴, Oliver Hohlfeld⁴

¹University of Twente, ²University of Sydney, ³North Auster University, ⁴IMDEA Networks, ⁵Brandenburg University of Technology

¹raholt@utwente.nl, ²hiller.jens@sydney.edu.au, ³johanna.amann@brandenburg-universitaet.de, ⁴abbas.razaghpanah@imdea.org, ⁵oliver.hohlfeld@tu-braunschweig.de

ABSTRACT

Transport Layer Security (TLS) 1.3 is a redesign of the Web's most important security protocol. It was standardized in August 2018 after a four-year-long, unprecedented design process involving many cryptographers and industry stakeholders. We use the new opportunity to track deployment, uptake, and use of a new mission-critical security protocol from the early design phase until well over a year after standardization. For a practical view, we combine and analyze data from active domain scans, passive monitoring of large networks, and a crowd-sourcing effort on Android devices. In contrast to TLS 1.2, where adoption took more than five years and was prompted by severe attacks on previous versions, TLS 1.3 is deployed surprisingly quickly and without security concerns calling for it. Just 15 months after standardization, it is used in about 20% of connections we observe. Deployment on popular domains is at 30% and at about 10% across the com/net/org top-level domains (TLDs). We show that the development and fast deployment of TLS 1.3 is best understood as a story of experimentation and centralization. Very few giant, global actors drive the development. We show that Cloudflare alone brings deployment to sizable numbers and describe how actors like Facebook and Google use their control over both client and server endpoints to experiment with the protocol and ultimately deploy it at scale. This story cannot be captured by a single dataset alone, highlighting the need for multi-perspective studies on Internet evolution.

Coming of Age: A Longitudinal Study of TLS Deployment

Platon Kotzias
IMDEA Software Institute
Universidad Politécnica de Madrid

Abbas Razaghpanah
Stony Brook University

Johanna Amann
ICSI/Cloudflare/LBNL

Kenneth G. Paterson
Royal Holloway, University of London

Narseo Vallina-Rodriguez
IMDEA Networks Institute
ICSI

Juan Caballero
IMDEA Software Institute

ABSTRACT

The Transport Layer Security (TLS) protocol is the de facto standard for encrypted communication on the Internet. However, it has been plagued by a number of different attacks and security issues over the last years. Addressing these attacks requires changes to the protocol, to server- or client-software, or to all of them. In this paper we conduct the first large-scale longitudinal study examining the evolution of the TLS ecosystem over the last six years. We place a special focus on the ecosystem's evolution in response to high-profile attacks.

For our analysis, we use a passive measurement dataset with more than 319.3B connections since February 2012, and an active dataset that contains TLS and SSL scans of the entire IPv4 address space since August 2013. To identify the evolution of specific clients we also create the -to our knowledge- largest TLS-client fingerprint database to date, consisting of 1,684 fingerprints.

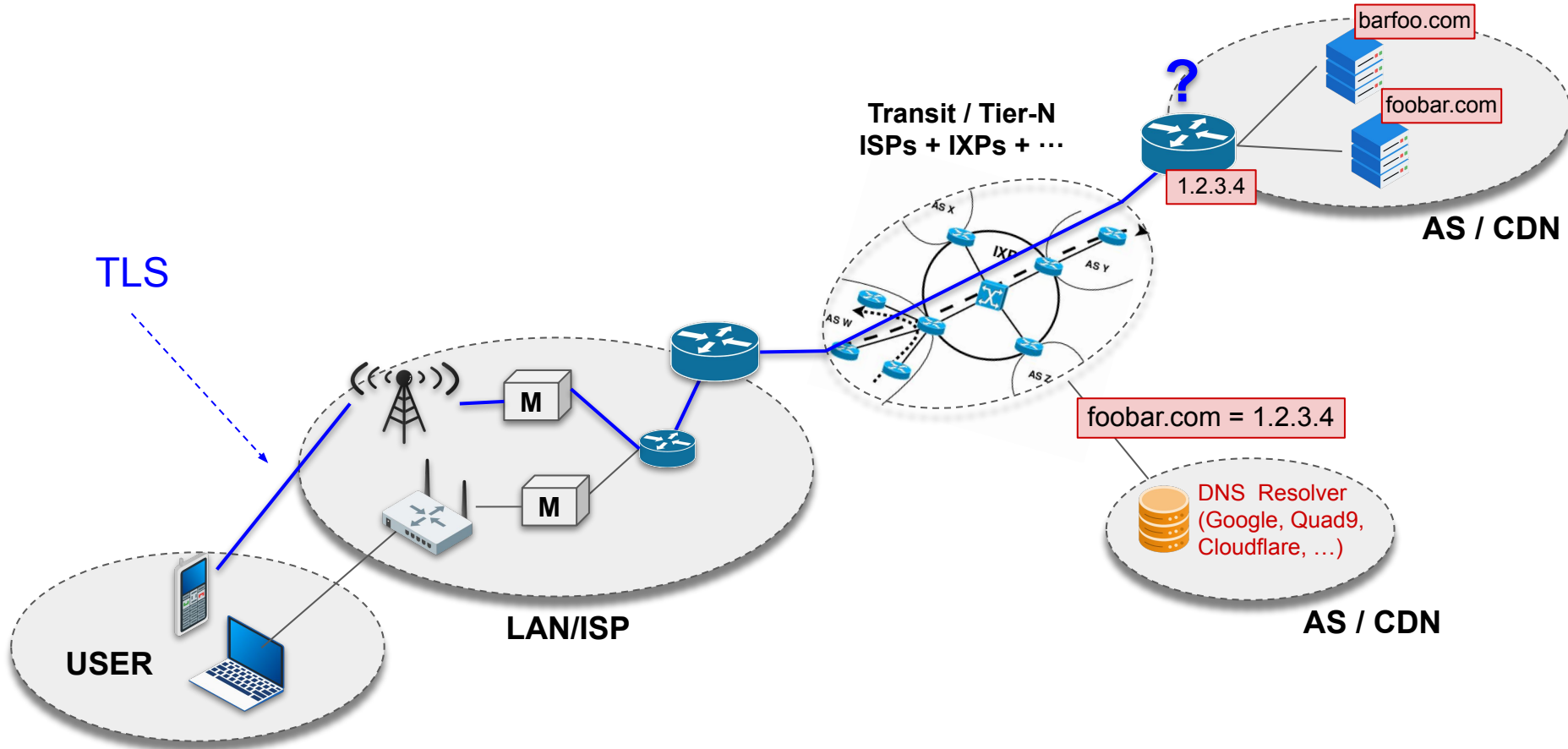
We observe that the ecosystem has shifted significantly since 2012, with major changes in which cipher suites and TLS extensions are offered by clients and accepted by servers having taken place. Where possible, we correlate these with the timing of specific attacks on TLS. At the same time, our results show that while clients, especially browsers, are quick to adopt new algorithms, they are also slow to drop support for older ones. We also encounter significant amounts of client software that probably unwittingly offer unsafe ciphers. We discuss these findings in the context of long tail effects in the TLS ecosystem.

of each new attack and vulnerability that is discovered. Over the last few years various TLS vulnerabilities such as BEAST, Lucky 13, POODLE, Heartbleed, FREAK, Logjam, and multiple attacks against RC4 have been discovered. The Snowden revelations have also highlighted weaknesses in TLS, specifically the reliance on RSA key transport for establishing keying material, a method that can be passively broken by an entity in possession of the server's RSA private key. Addressing these attacks requires changes to the protocol, to server-, or to client-software, or to all of them simultaneously.

Prior work highlights different parts of the TLS ecosystem like specific attacks [6, 9, 10, 17, 41, 44, 44, 63, 74, 82], problems of the PKI [7, 46, 54, 60] or problems of TLS usage in specific areas like on mobile devices [47, 71, 85]. However, to the best of our knowledge, no prior work has examined the specific impact of security issues on protocol deployment.

In this paper, we conduct a large-scale longitudinal study examining the evolution of the TLS ecosystem since 2012 both on the client and on the server side. We analyze trends and evolution of the ecosystem, putting a special focus on changes occurring in response to specific high-profile attacks. For this, we use a combination of passive and active measurement data. Our passive measurements have been running continuously since February 2012 and currently contain protocol information about more than 319.3B TLS connections. The active measurement data provided to us by Censys [42] contains SSL and TLS scans of the entire IPv4 address space starting from August 2013.

TLS Server Name Indication (**SNI**)



TLS Server Name Indication (**SNI**)

```
> Internet Protocol Version 4, Src: 192.168.172.128, Dst: 216.58.223.110
> Transmission Control Protocol, Src Port: 52662, Dst Port: 443, Seq: 1, Ack: 1, Len: 23
√ Secure Sockets Layer
  √ TLSv1 Record Layer: Handshake Protocol: Client Hello
    Content Type: Handshake (22)
    Version: TLS 1.0 (0x0301)
    Length: 229
    √ Handshake Protocol: Client Hello
      Handshake Type: Client Hello (1)
      Length: 225
      Version: TLS 1.2 (0x0303)
      > Random: f4e5aa91712540053d080909719269b21b1d7c0890969a02...
      Session ID Length: 0
      Cipher Suites Length: 40
      > Cipher Suites (20 suites)
      Compression Methods Length: 1
      > Compression Methods (1 method)
      Extensions Length: 144
      > Extension: renegotiation info (len=1)
      √ Extension: server_name (len=20)
        Type: server_name (0)
        Length: 20
        √ Server Name Indication extension
          Server Name list length: 18
          Server Name Type: host_name (0)
          Server Name length: 15
          Server Name: play.google.com
      > Extension: extended_master_secret (len=0)
      > Extension: signature_algorithms (len=20)
      > Extension: status_request (len=5)
      > Extension: next_protocol_negotiation (len=0)
      > Extension: signed_certificate_timestamp (len=0)
```

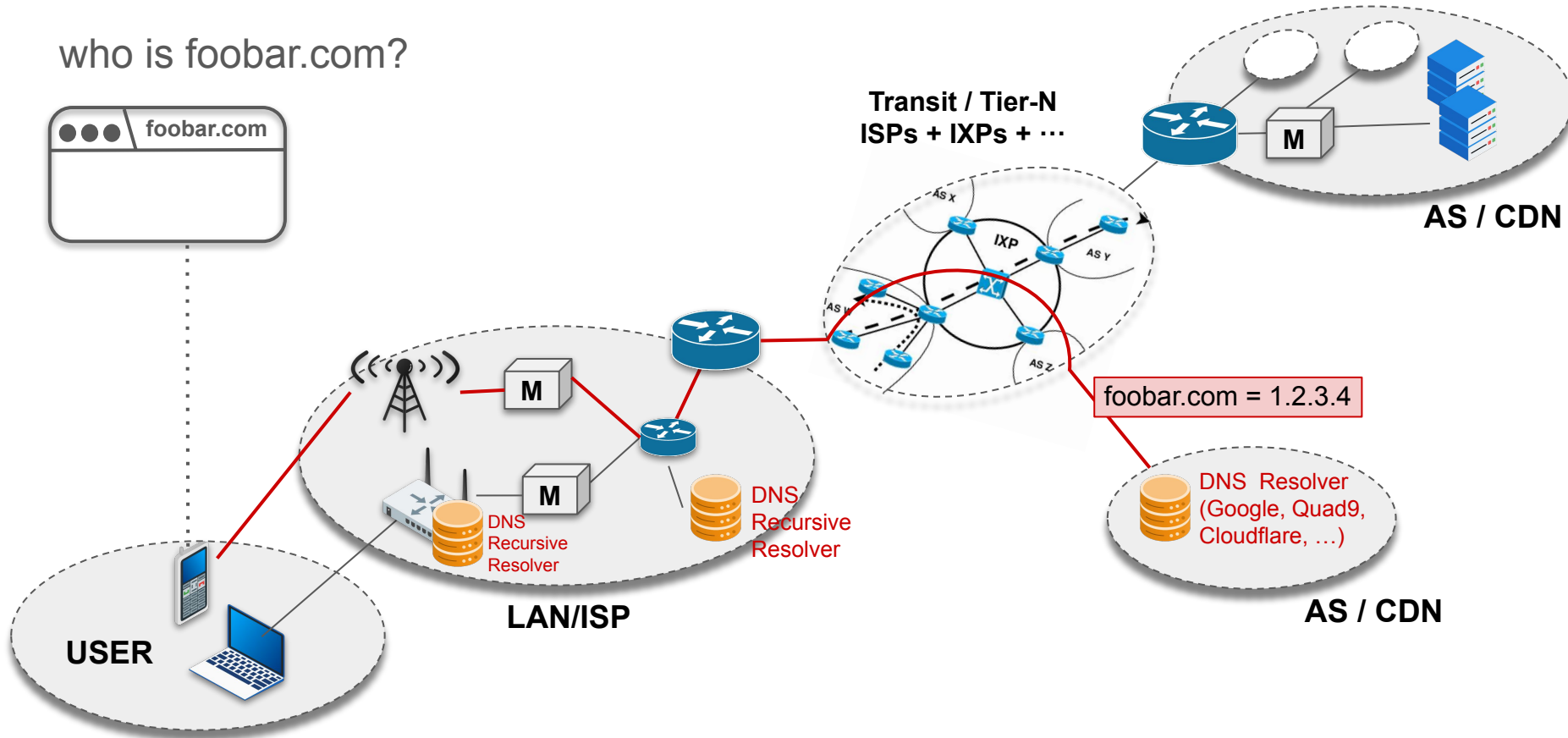
Remark: ClientHello is not encrypted

Securing the DNS

DNS-over-X

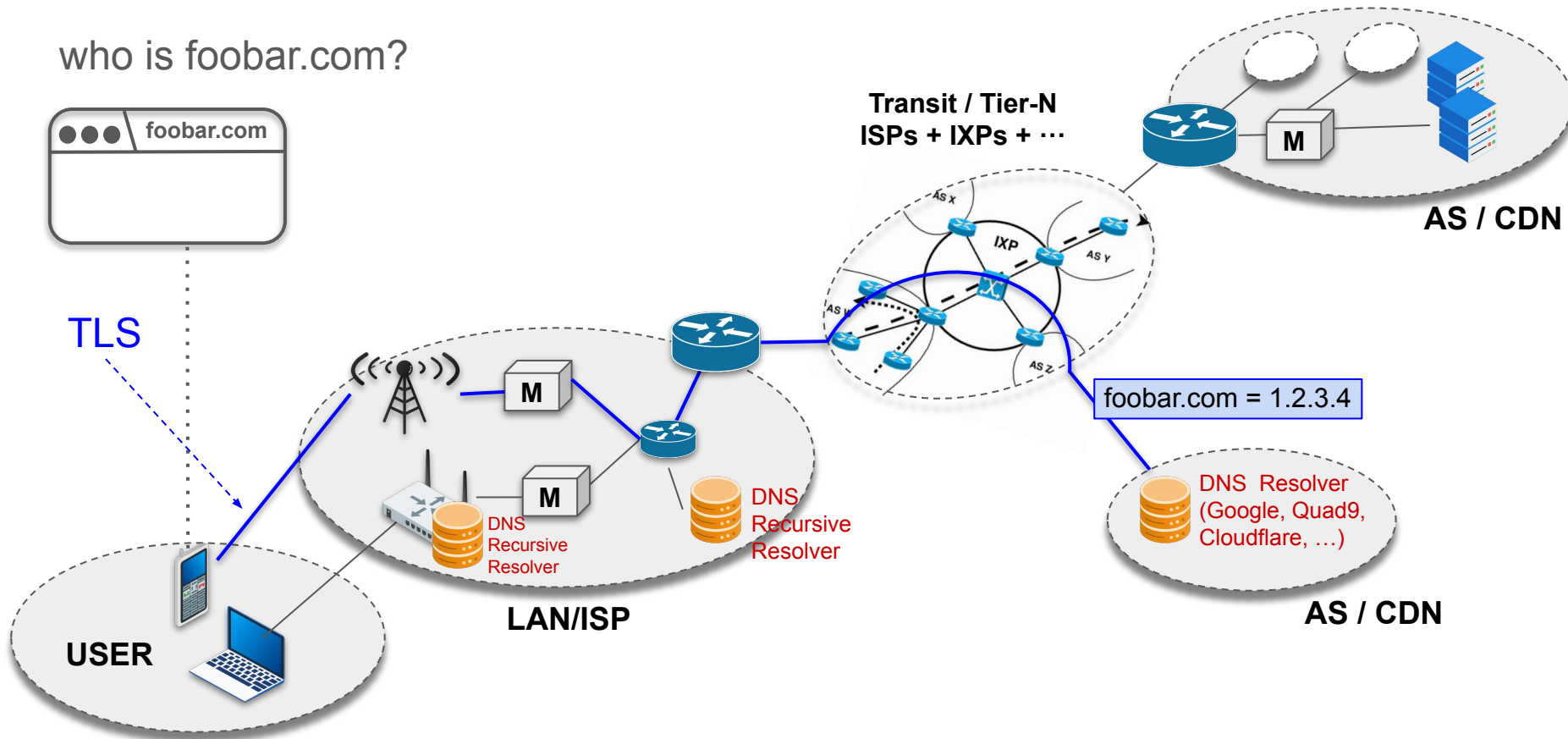
Securing the DNS: DoH/DoT/DoQ

who is foobar.com?



Securing the DNS: DoH/DoT/DoQ

who is foobar.com?



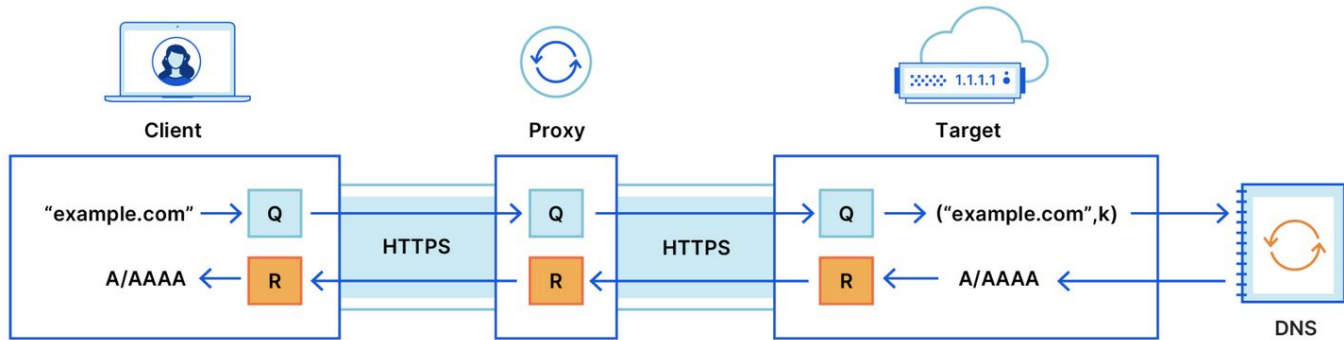
Securing the DNS: **DoH vs. DoT**

Each standard was developed separately and has its own RFC

- DoH (RFC 8484, Oct 2018)
 - HTTP \Rightarrow tcp/443
 - Indistinguishable from regular HTTP traffic \Rightarrow DNS queries and responses are camouflaged within other HTTPS traffic
- DoT (RFC 7858, May 2016)
 - tcp/853
 - Detectable \Rightarrow can be blocked

More DNS privacy: **Oblivious DoH (ODoH)**

- DoH provides confidentiality and authentication for DNS but it is not private
 - Clients reveal their IP addresses
- oDoH (RFC 9230, Experimental) builds on DoH to solve this problem



Source: Cloudflare

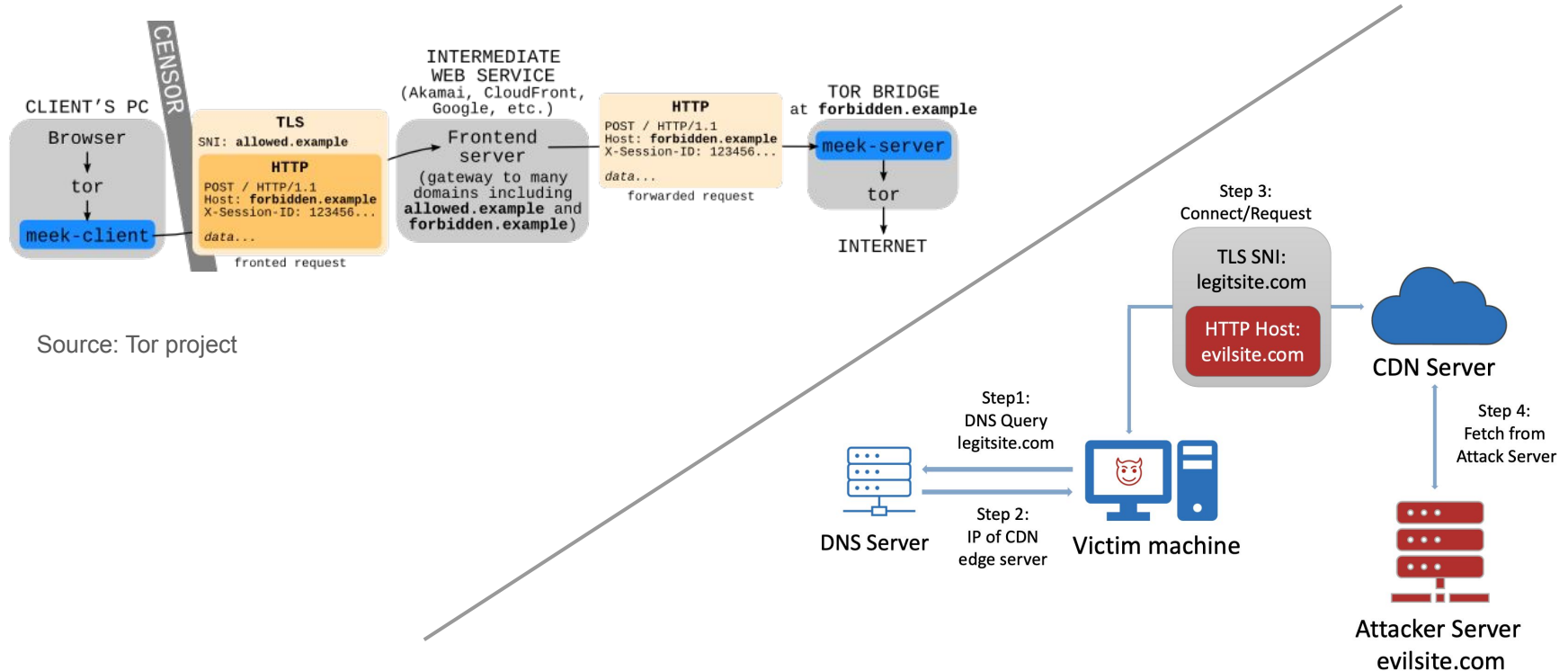
Hiding the destination

Domain Fronting, ESNI, and ECH

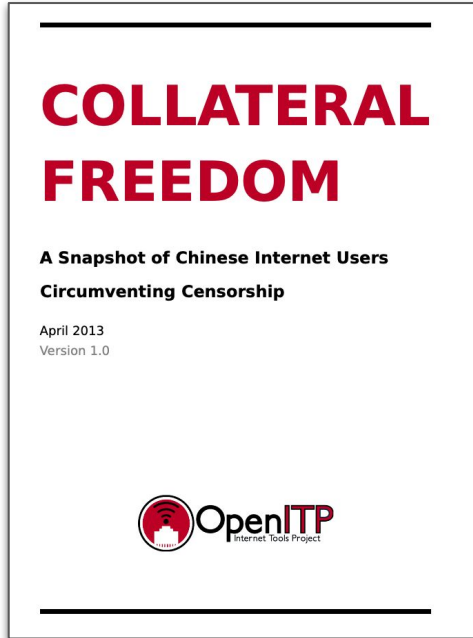
Domain fronting

- Anti-censorship technique
 - Telegram, Signal —raised protests in Russia & China
 - Tor (old meek plugin)
 - Also used by malware
 - Blocking C2 traffic becomes harder
- Exploits discrepancy between the TLS server SNI and the HTTP Host header in the request
 - CDNs typically rely on the **Host header** to identify the server (**encrypted**, not visible)
 - **SNI** used in TLS: **visible** to network traffic
 - Result is true endpoint of the communication is hidden

Domain fronting: the Tor meek plugin & a C2 beacon



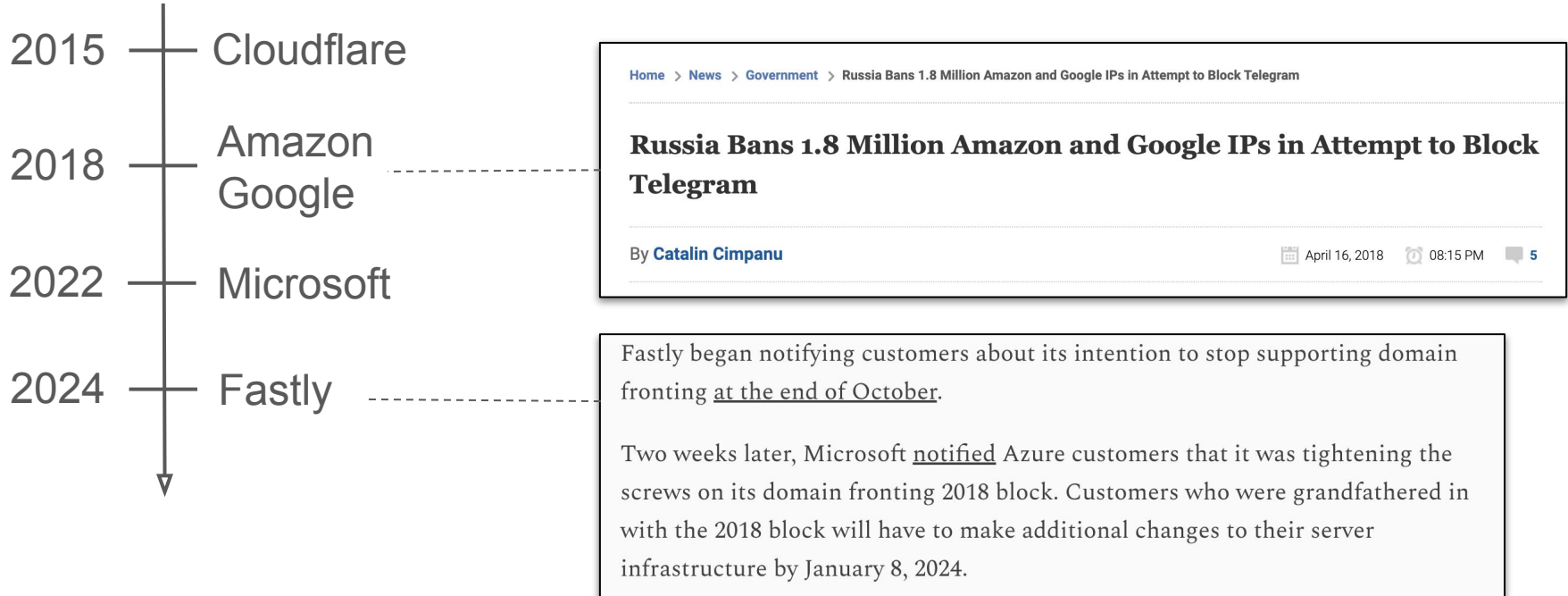
Domain fronting and **collateral freedom**



*“Collateral freedom is an anti-censorship strategy that attempts to make it **economically prohibitive for censors to block content** on the Internet. This is achieved by hosting content on cloud services that are considered by censors to be **“too important to block,”** and then using encryption to prevent censors from identifying requests for censored information that is hosted among other content, **forcing censors to either allow access to the censored information or take down entire services.**”*

Domain fronting and collateral freedom

Domain Fronting Bans Timeline



Domain fronting: is it really dead?

Measuring CDNs susceptible to Domain Fronting

Karthika Subramani
ksubramani@gatech.edu
Georgia Institute of Technology
USA

Roberto Perdisci
perdisci@uga.edu
University of Georgia and Georgia
Institute of Technology
USA

Pierros Skafidas
pskafidas3@gatech.edu
Georgia Institute of Technology
USA

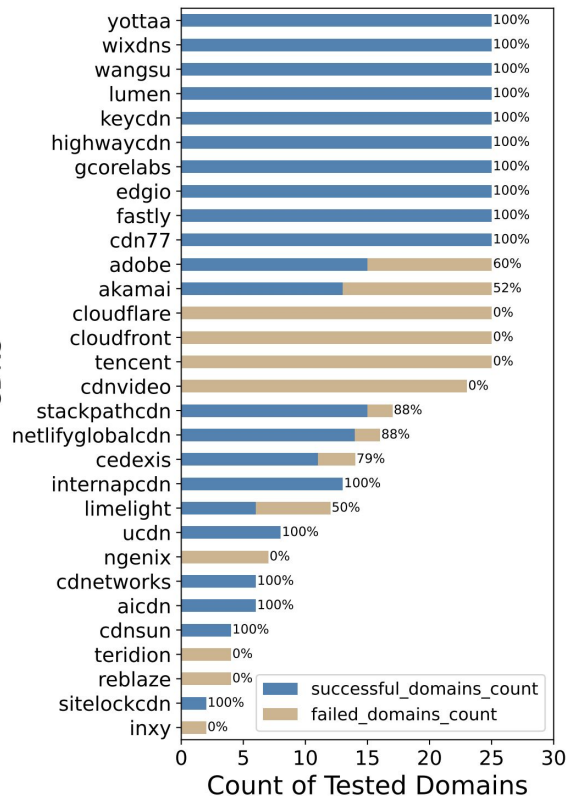
ABSTRACT

Domain fronting is a network communication technique that involves leveraging (or abusing) content delivery networks (CDNs) to disguise the final destination of network packets by presenting them as if they were intended for a different domain than their actual endpoint. This technique can be used for both benign and malicious purposes, such as circumventing censorship or hiding malware-related communications from network security systems. Since domain fronting has been known for a few years, some popular CDN providers have implemented traffic filtering approaches to curb its use at their CDN infrastructure. However, it remains unclear to what extent domain fronting has been mitigated.

To better understand whether domain fronting can still be effectively used, we propose a systematic approach to discover CDNs that are still prone to domain fronting. To this end, we leverage passive and active DNS traffic analysis to pinpoint domain names served by CDNs and build an automated tool that can be used to discover CDNs that allow domain fronting in their infrastructure. Our results reveal that domain fronting is feasible in 22 out of 30 CDNs that we tested, including some major CDN providers like Akamai and Fastly. This indicates that domain fronting remains widely available and can be easily abused for malicious purposes.

with stringent internet restrictions, such as China and Iran, domain fronting has been instrumental for activists and ordinary citizens alike to bypass digital barriers and access platforms like Signal and Telegram [5, 9]. However, the same technique has found favor among malicious actors. For instance, APT29, also known as Cozy Bear, reportedly used domain fronting to camouflage their malware command-and-control (C2) infrastructure, complicating detection and attribution [7]. Furthermore, according to a recent study [10], about 3.5% of all Cobalt Strike Beacons were configured to use domain fronting to effectively evade detection for a prolonged period of time.

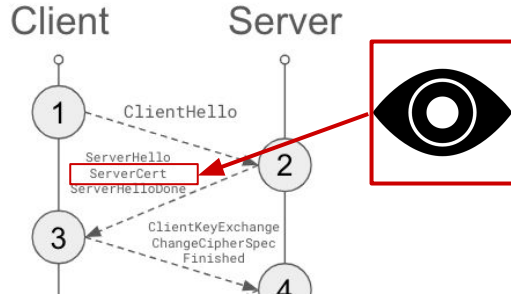
In order to detect or defend against domain fronting, censors and network operators are compelled to adopt drastic CDN traffic blocking measures, often with considerable collateral damage, in an attempt to mitigate the associated risks [20]. Rather than blocking CDN traffic altogether, a more effective approach to counter this threat lies within the infrastructure of CDNs themselves. To prevent unintended consequences from nationwide censorship, few popular CDNs have taken measures to prevent domain fronting on their platforms. For example, Google and Amazon disabled domain fronting in their services in 2018 [1], while Microsoft Azure only disabled it recently in November 2022, following its use by Meek, a Tor plugin for traffic tunneling [4, 7]. Irrespective of these measures,



2024 study finds domain fronting still works in 73% (22/30) of the tested CDNs

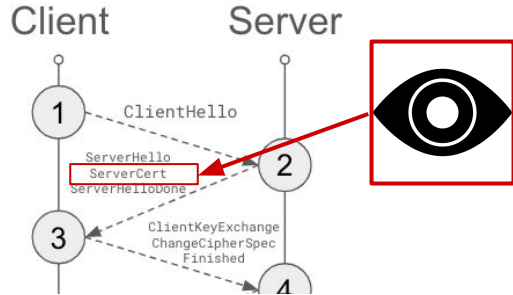
ESNI: Encrypting the SNI

TLS 1.2 handshake revisited

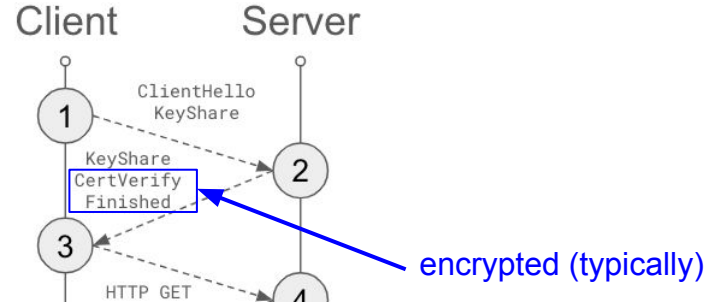


ESNI: Encrypting the SNI

TLS 1.2 handshake revisited

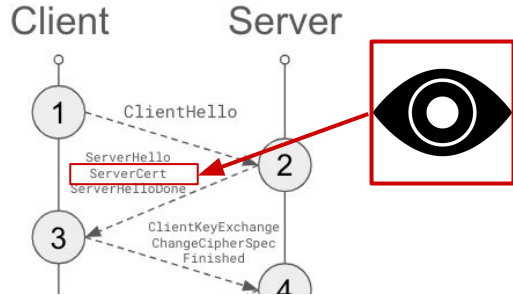


TLS 1.3 handshake revisited

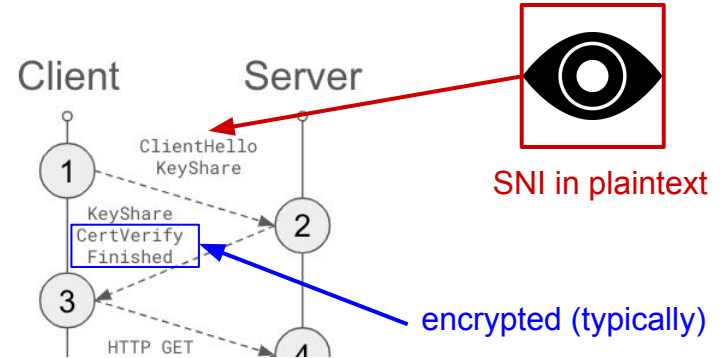


ESNI: Encrypting the SNI

TLS 1.2 handshake revisited



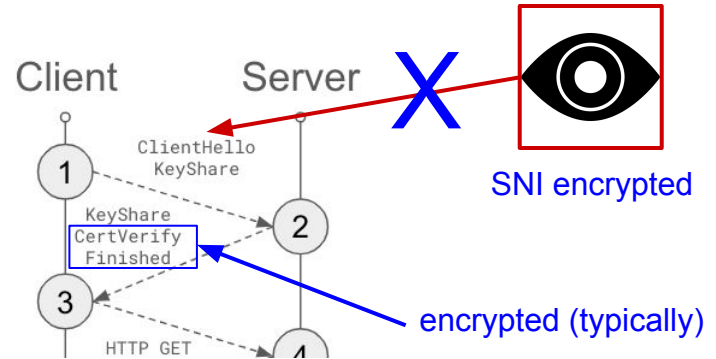
TLS 1.3 handshake revisited



ESNI: Encrypting the SNI

Encrypt the SNI in the ClientHello message

TLS 1.3 handshake revisited

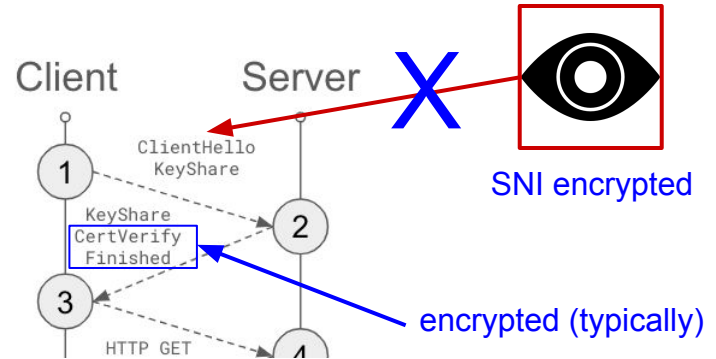


ESNI: Encrypting the SNI

Encrypt the SNI in the ClientHello message

Wait a moment ... using what key?

TLS 1.3 handshake revisited



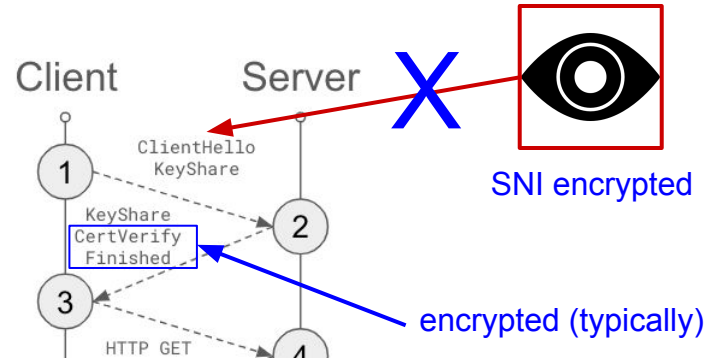
ESNI: Encrypting the SNI

Encrypt the SNI in the ClientHello message

Wait a moment ... using what key?

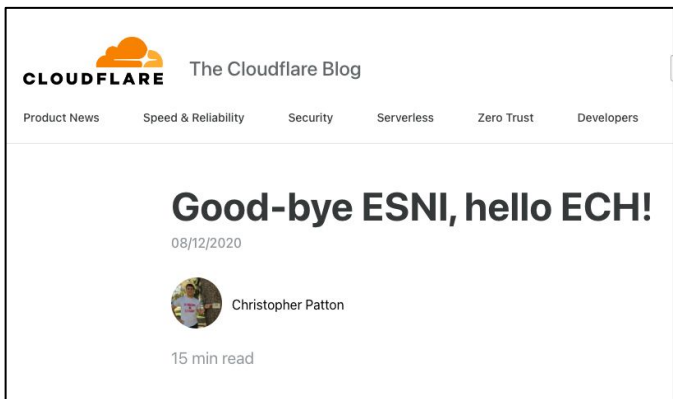
Get server public key through DNS (TXT record), preferably using DoH or DoT

TLS 1.3 handshake revisited




```
$ dig _esni.crypto.dance TXT +short
```



```
"/wGuNThxACQAHQAgXzyda0XSJRQWzDG7lk/r01r1ZQy+MdNxKg/mAqSnt0EAAhMBAQQAAAAAX67XsAAAAABftsCwAAA="
```



ESNI is no more


- Was only adopted by Cloudflare, Mozilla Firefox and a few others in 2018
- Abruptly removed around 2020-21 by all parties
- Alleged reasons include
 - Protection it gives is incomplete because there are other sensitive fields in the ClientHello
 - A bunch of sophisticated attacks proposed
 - Using DNS for key distribution is not as easy as it seems
- Solution?
 - **Encrypt the whole ClientHello message**






Encrypted Client Hello - the last puzzle piece to privacy


2023-09-29




Achiel van der Mandele



Alessandro Ghedini



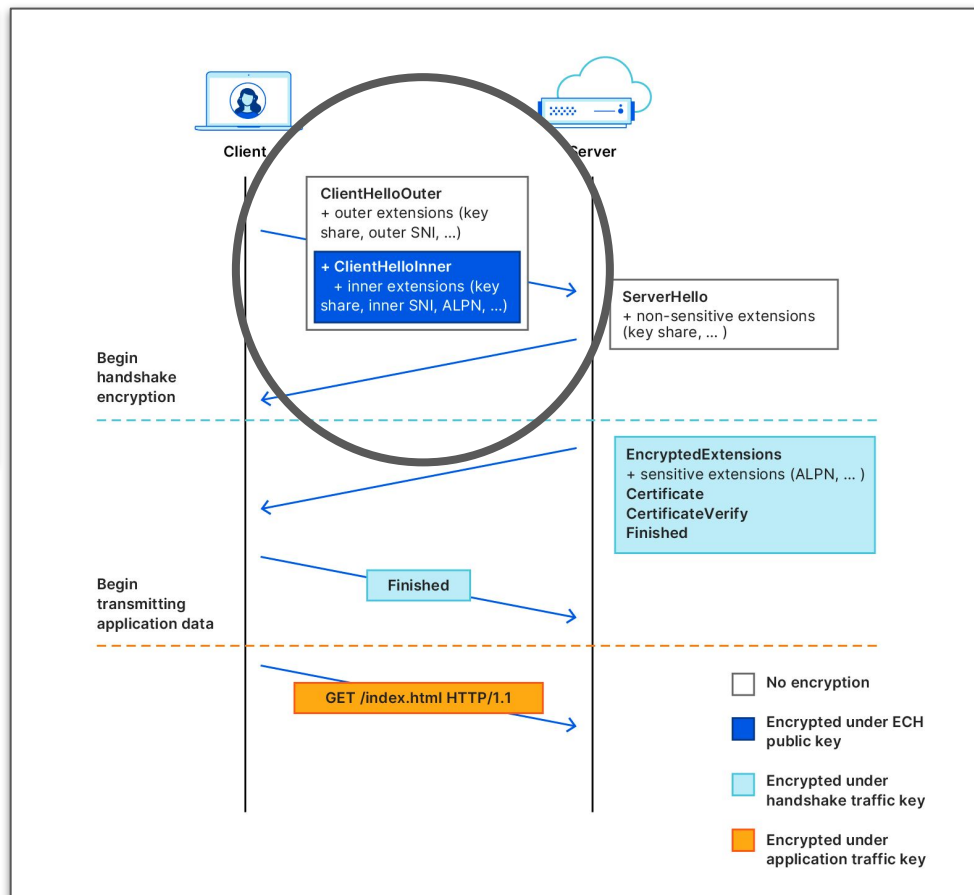
Christopher Wood



Rushil Mehra

4 min read

This post is also available in [简体中文](#), [日本語](#), [한국어](#) and [繁體中文](#).



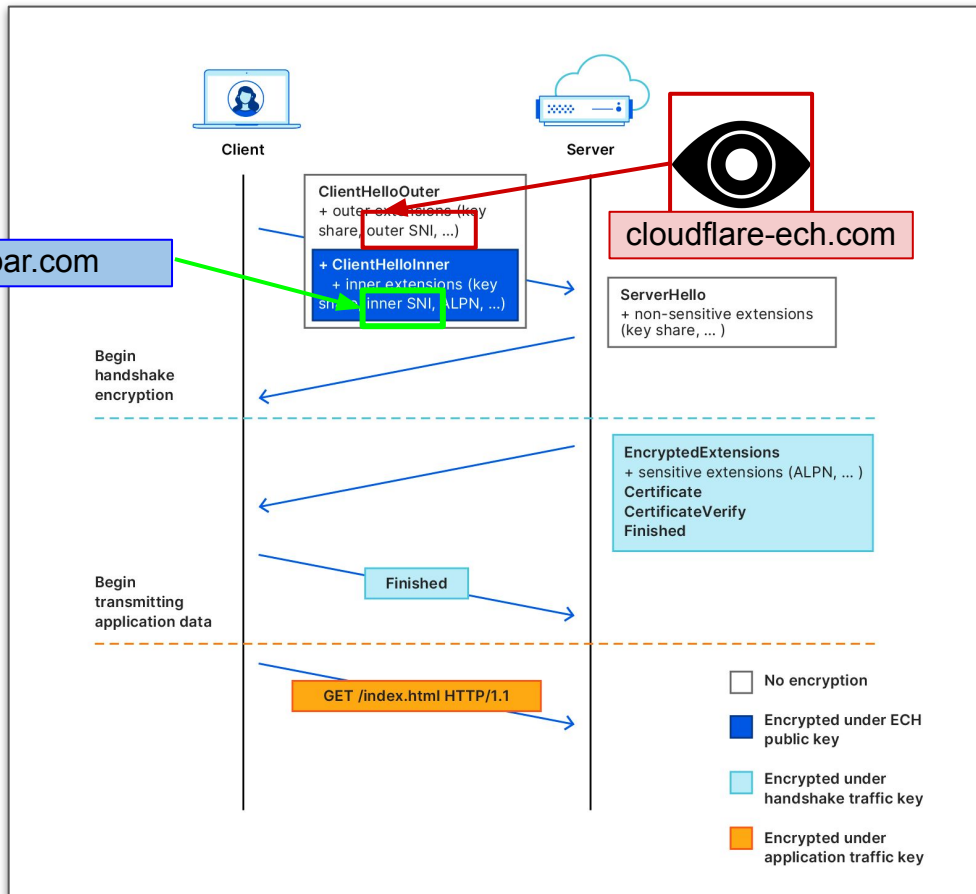
Encrypted Client Hello - the last puzzle piece to privacy


2023-09-29

Achiel van der MandeleAlessandro GhediniChristopher WoodRushil Mehra

4 min read


This post is also available in [简体中文](#), [日本語](#), [한국어](#) and [繁體中文](#).





 **CLOUDFLARE**


Encrypted Client Hello - the last puzzle piece to privacy

2023-09-29

 Achiel van der Mandele

 Alessandro Ghedini

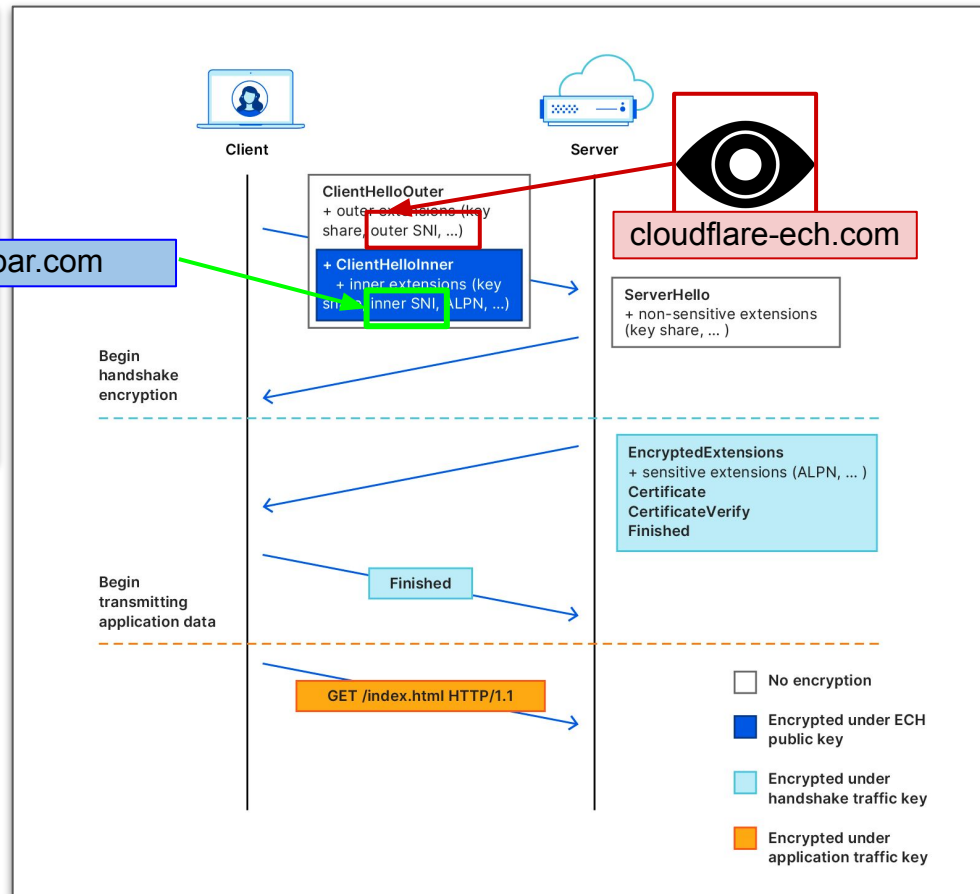
 Christopher Wood

 Rushil Mehra

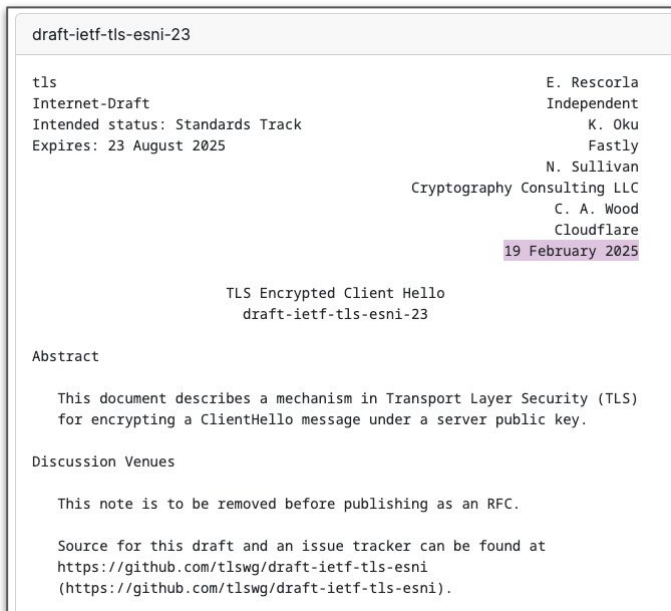
4 min read

This post is also available in [简体中文](#), [日本語](#), [한국어](#) and [繁體中文](#).

Wait a moment ... using what key?



ECH: status & questions



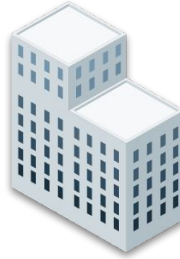
- Developments taking place as we speak
- How about problems with DNS-based key distribution?
 - Kinda solved with the newly introduced HTTPS DNS record type. See <https://blog.cloudflare.com/speeding-up-https-and-http-3-negotiation-with-dns/>
- Will it be broadly adopted? Ever? Soon? Nobody knows. Obstacles:
 - “Network ossification”: larger-than-expected TLS connection failures because of middleboxes not supporting it
 - Some countries (usual suspects) threaten to block all known client-facing servers (e.g., cloudflare-ech.com)
 - Realistic threat as there are only a bunch and are easy to enumerate
 - Could (1) break the Internet for many and (2) hurt key stakeholders
 - Unclear how CDNs and browsers would react
- Sovereignty reasons

The “bad” news

The Web PKI

The Web PKI

- Web Public Key Infrastructure (PKI) enables TLS **server authentication** by linking an identity (DNS name or IP address) to a cryptography public key
- Web PKI is ...
 - ... the most widely deployed PKI
 - ... foundational to the security of the web
 - ... rapidly changing for technical and political reasons
 - ... fragile, complicated, sometimes dirty



Certificate Authority

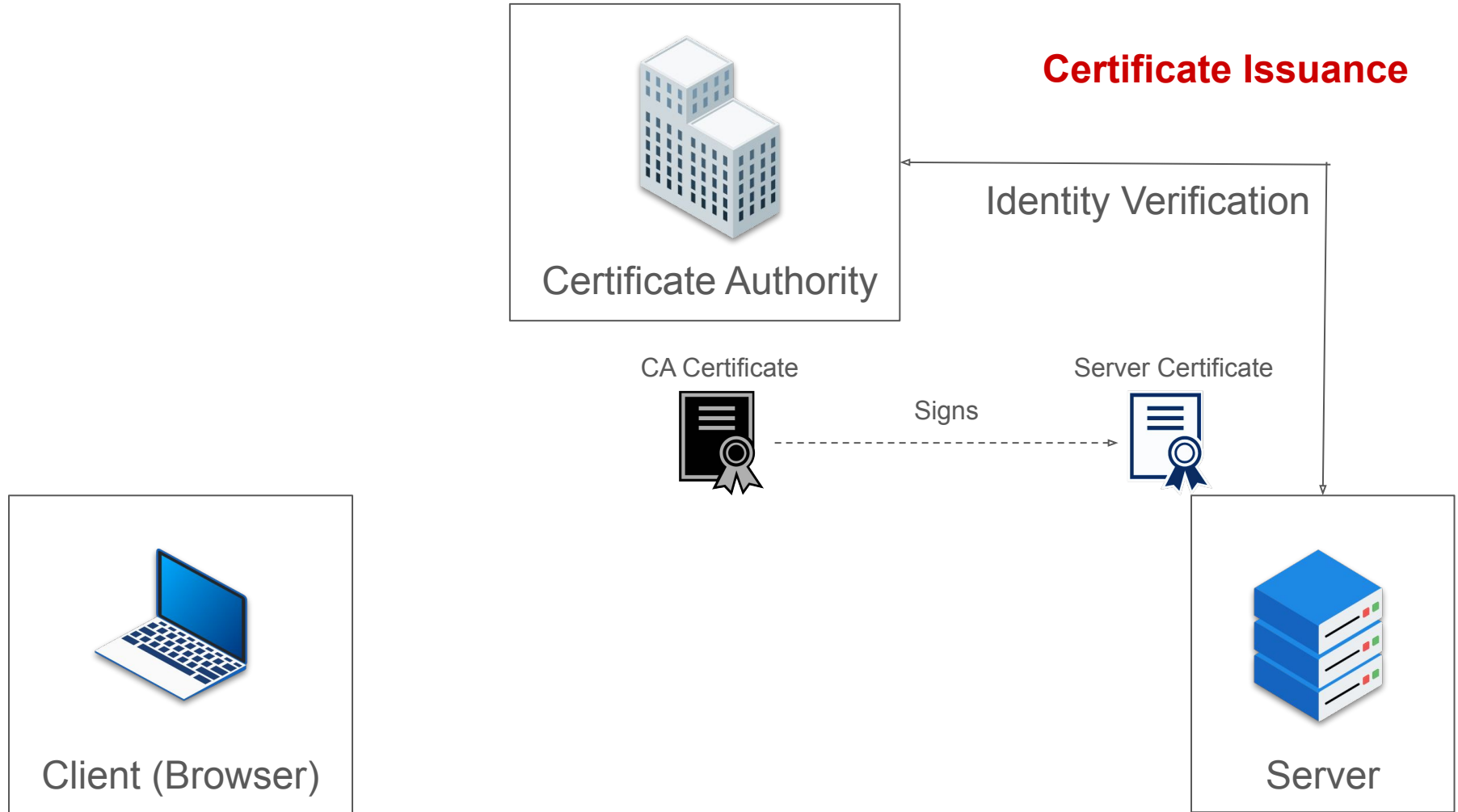


Client (Browser)



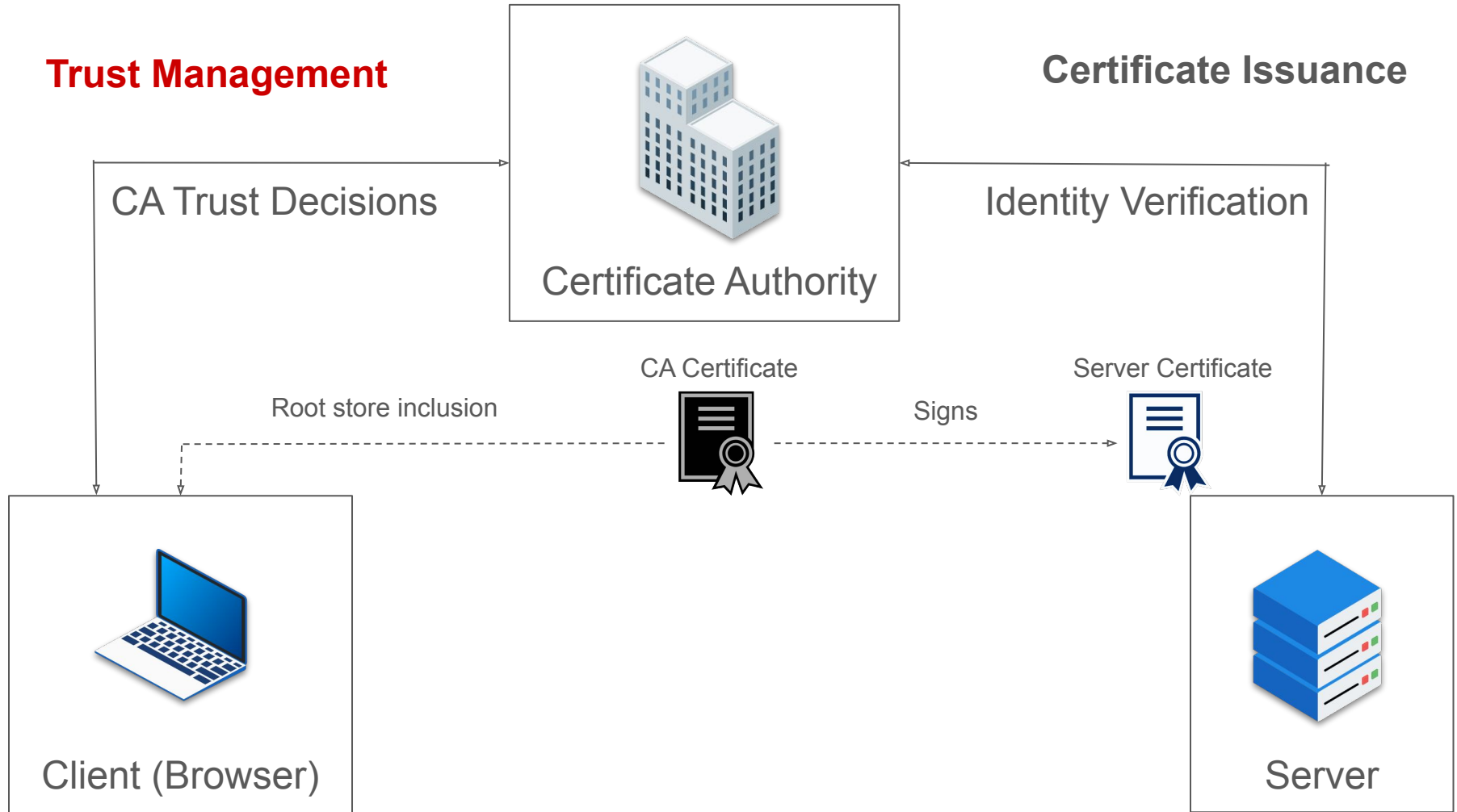
Server

Certificate Issuance



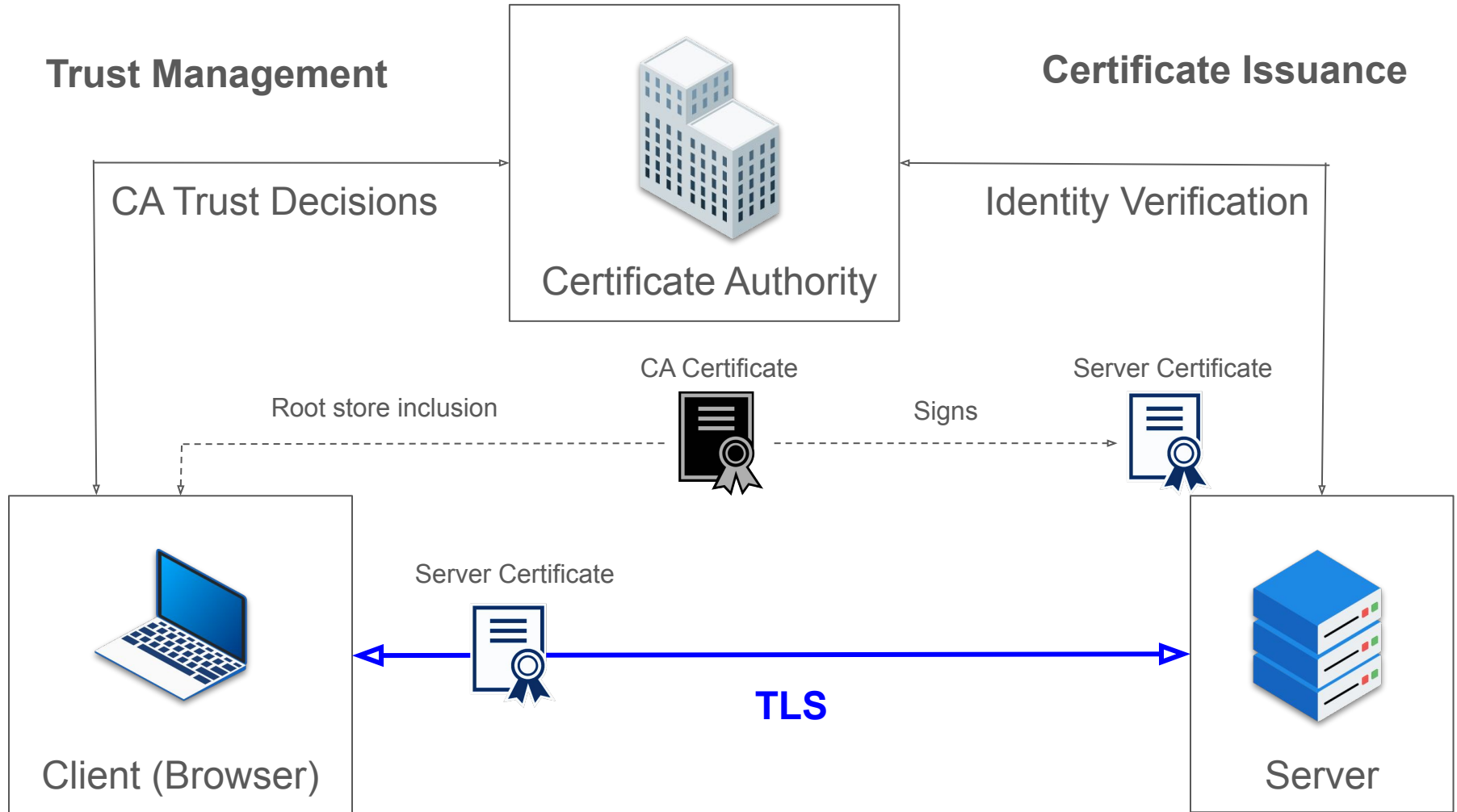
Trust Management

Certificate Issuance



Trust Management

Certificate Issuance

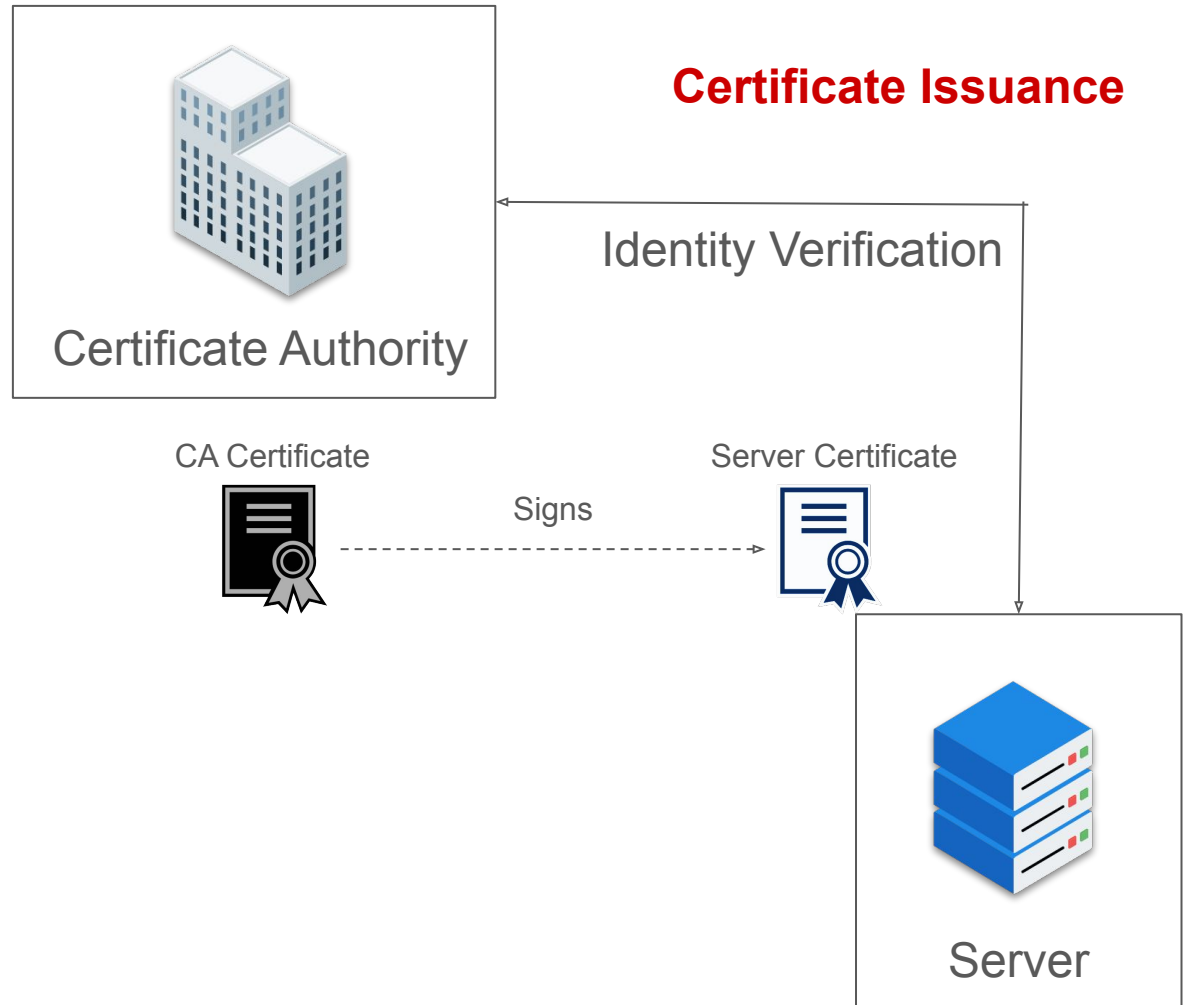


Goal

- Verify that a network identifier controls some cryptographic public key

Problem

- How to verify?
- What does control mean?

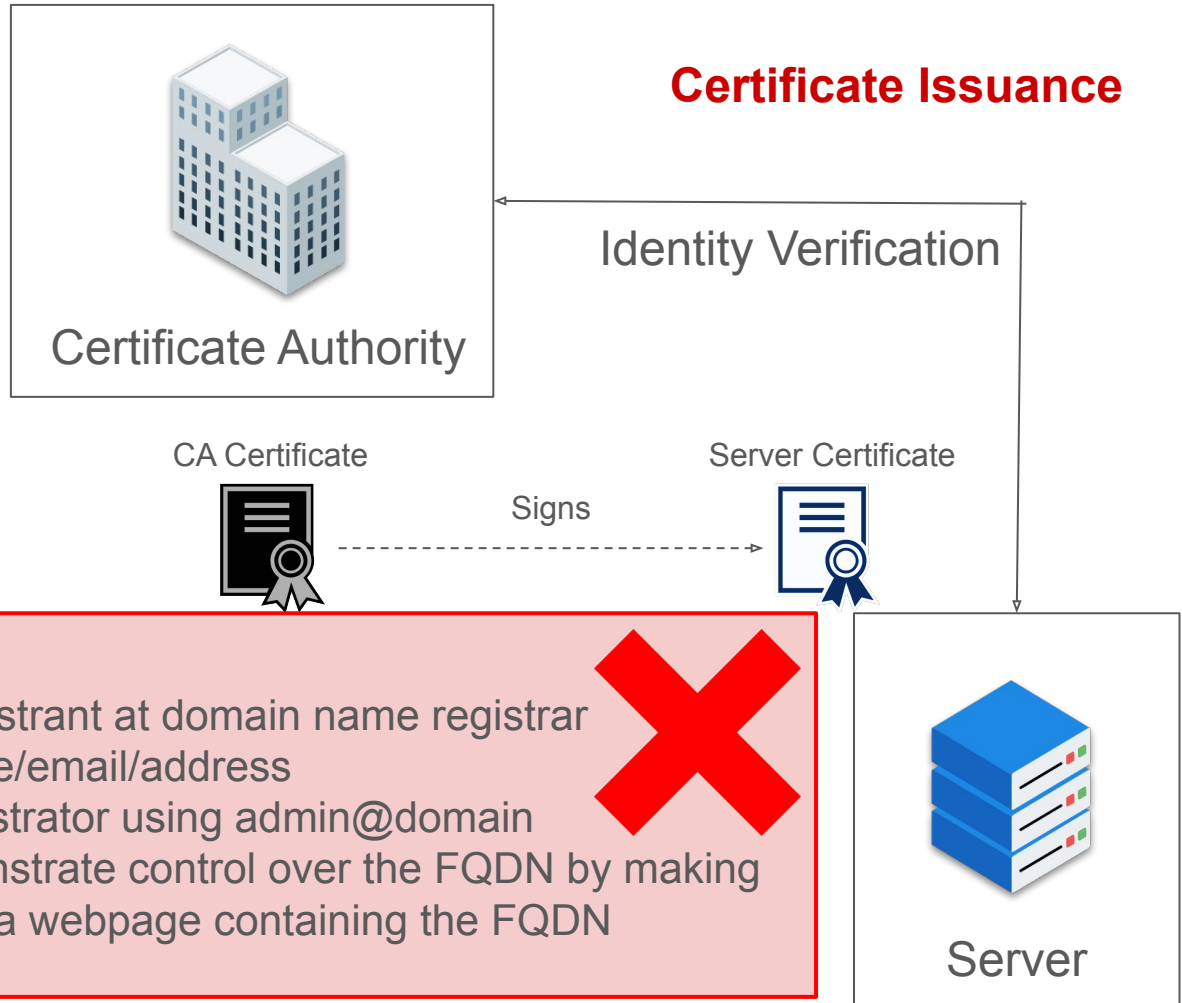


Goal

- Verify that a network identifier controls some cryptographic public key

Problem

- How to verify?
- What does control mean?



Historically (ca. 2012)

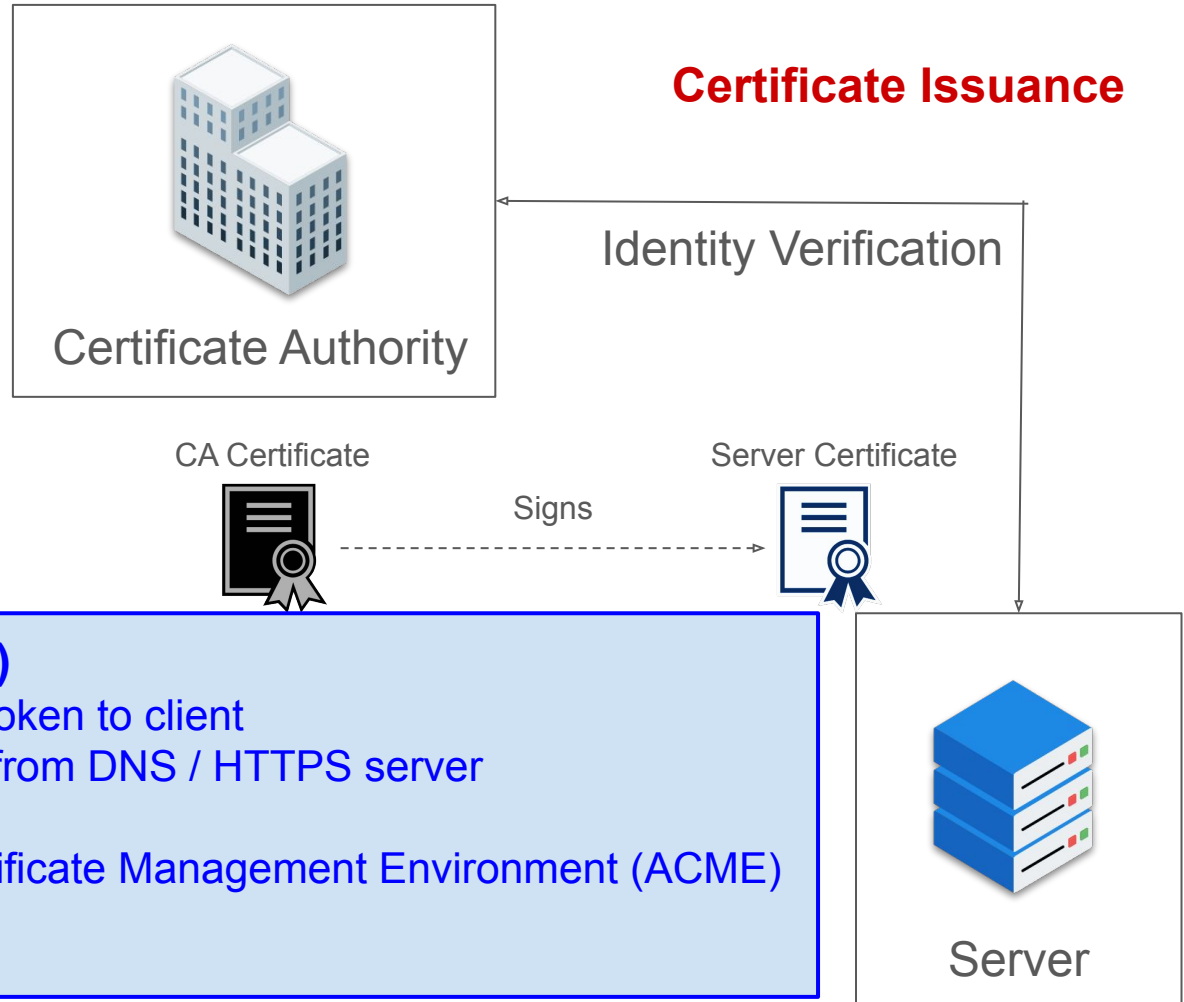
- Confirming applicant is registrant at domain name registrar
- Contact registrant via phone/email/address
- Contact the domain administrator using admin@domain
- Having the applicant demonstrate control over the FQDN by making an agreed-upon change to a webpage containing the FQDN
- ...

Goal

- Verify that a network identifier controls some cryptographic public key

Problem

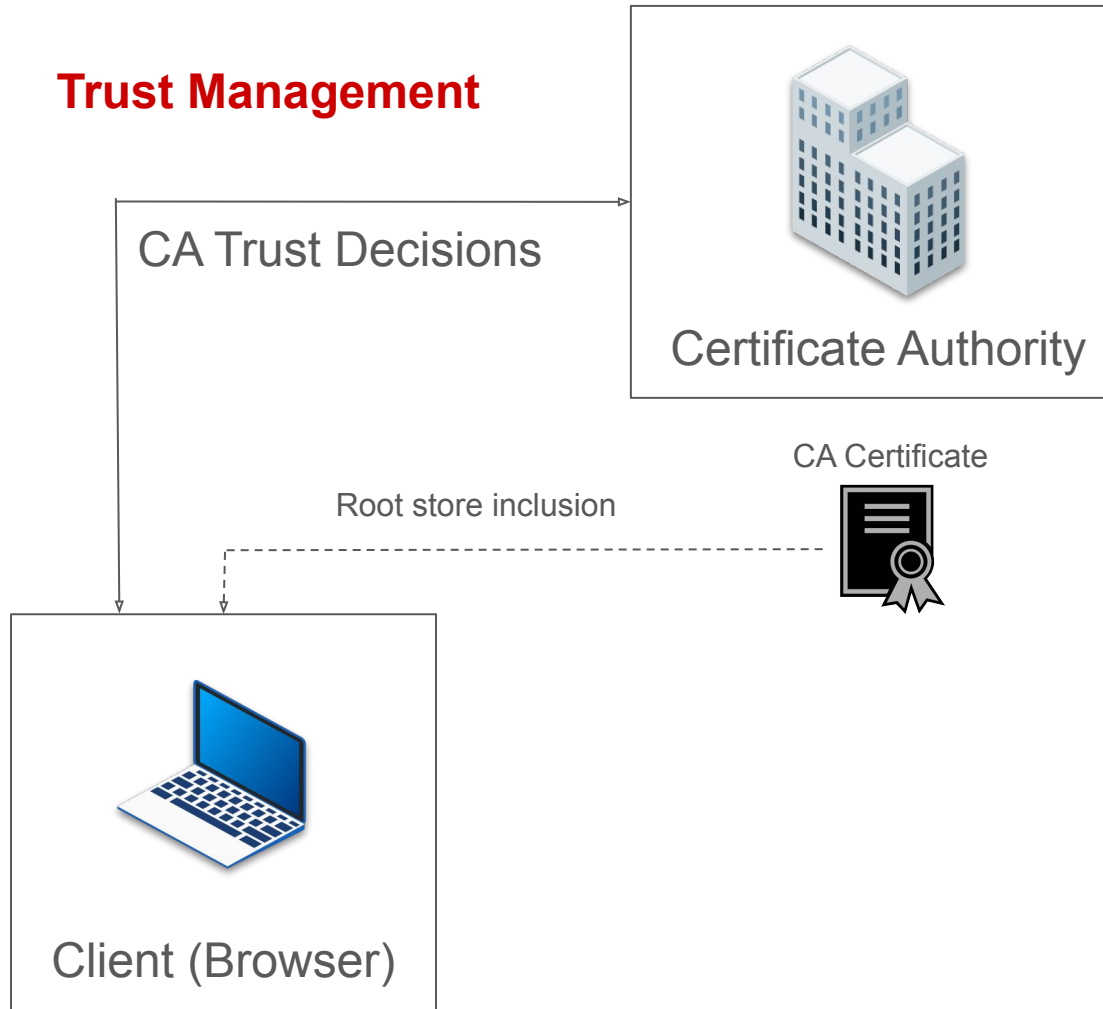
- How to verify?
- What does control mean?



Modern issuance (ca. 2021)

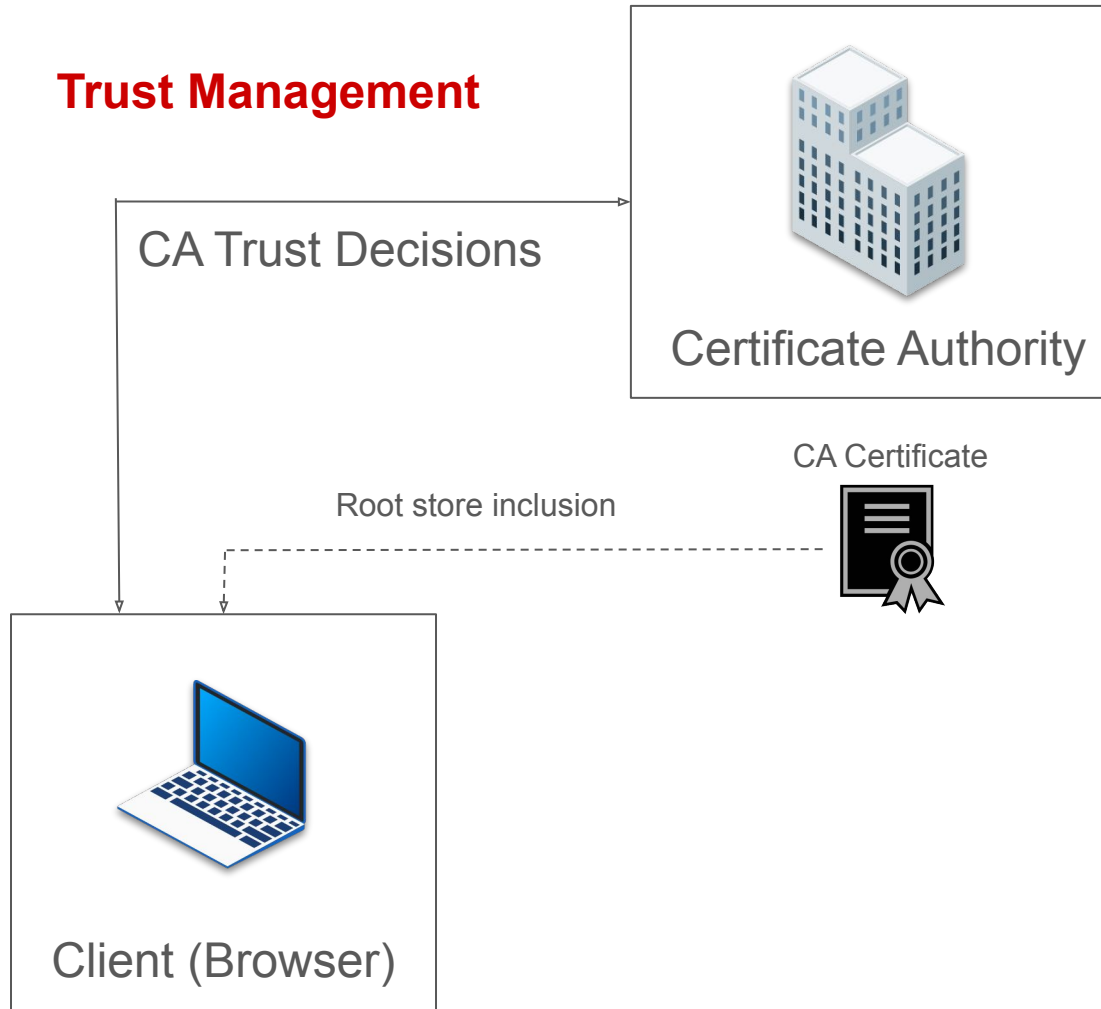
- Step 1. CA sends random token to client
- Step 2. CA retrieves token from DNS / HTTPS server
- Automatable!
- RFC 8555 - Automatic Certificate Management Environment (ACME)
- Let's Encrypt

Trust Management



In the current Web PKI design, every trust anchor is a single point of failure

Trust Management



In the current Web PKI design, every trust anchor is a single point of failure

KIM ZETTER SECURITY SEP 28, 2011 3:05 PM

DigiNotar Files for Bankruptcy in Wake of Devastating Hack

A Dutch certificate authority that suffered a major hack attack this summer has been unable to recover from the blow and filed for bankruptcy this week.

SAVE

A hand holding a laptop screen showing a blue background with a white circle.

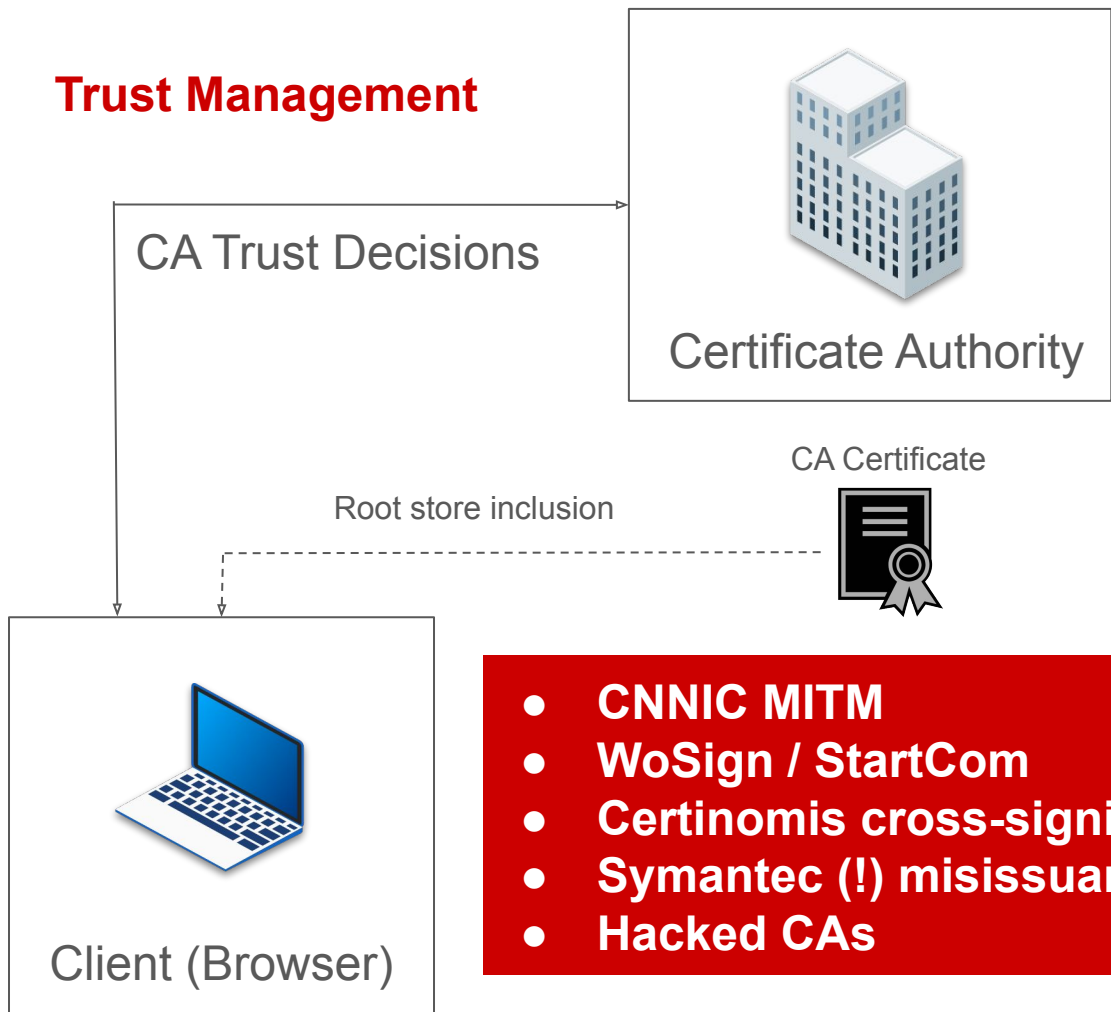
Go to ...

A Dutch certificate authority that suffered a major hack attack this summer has been unable to recover from the blow and filed for bankruptcy this week.

DigiNotar, which is owned by Illinois-based Vasco Data Security and was the primary provider of digital security certificates for domains owned by the Dutch government, was breached in early June due to lax security.

The breach allowed the intruder to trick DigiNotar's system into issuing him more than 500 fraudulent digital certificates for top internet companies like Google, Mozilla, and Skype. This meant that users who went to a supposedly secure page such as

Trust Management



In the current Web PKI design, every trust anchor is a single point of failure

- **CNNIC MITM**
- **WoSign / StartCom**
- **Certinomis cross-signing**
- **Symantec (!) misissuance**
- **Hacked CAs**

KIM ZETTER SECURITY SEP 28, 2011 3:05 PM

DigiNotar Files for Bankruptcy in Wake of Devastating Hack

A Dutch certificate authority that suffered a major hack attack this summer has been unable to recover from the blow and filed for bankruptcy this week.

SAVE

A Dutch certificate authority that suffered a major hack attack this summer has been unable to recover from the blow and filed for bankruptcy this week.

Go to ...

A Dutch certificate authority that suffered a major hack attack this summer has been unable to recover from the blow and filed for bankruptcy this week.

DigiNotar, which is owned by Illinois-based Vasco Data Security and was the primary provider of digital security certificates for domains owned by the Dutch government, was breached in early June due to lax security.

The breach allowed the intruder to trick DigiNotar's system into issuing him more than 500 fraudulent digital certificates for top internet companies like Google, Mozilla, and Skype. This meant that users who went to a supposedly secure page such as

How to evaluate CA trustworthiness?

Current practices

1. Bizz / Gov relationships
2. Independent audits
3. Certificate transparency: CT logs + signed cert timestamps (SCT) verification during TLS

But still huge problem

How to evaluate CA trustworthiness?

Current practices

1. Bizz / Gov relationships
2. Independent audits
3. Certificate transparency: CT logs + signed cert timestamps (SCT) verification during TLS

But still huge problem

THE DIRTY LAUNDRY OF THE WEB PKI

Note: Presentation times are in Pacific Standard Time (PST).

Tuesday, January 24, 2023 - 4:10 pm-4:40 pm

Emily Stark, Google

Abstract:

When you type "<https://example.com>" in your web browser, how do you know that you're establishing a secure connection? The question is foundational to the web security model, and the answer rests in the web public key infrastructure (PKI). Certificate Authorities (CAs) issue certificates that authenticate websites. Sadly, the web PKI – which is so foundational to the common web – is shockingly antiquated, overcomplicated, and cruddy. In this talk, Emily will explain how the web PKI works, exposing the fragile security infrastructure on which the web is built. I'll also outline some ideas for a next-generation server authentication model for the web.

Emily is a software engineer and manager working on the Google Chrome web browser. She leads Chrome's secure connections team, responsible for trustworthy, understandable encrypted and authenticated connections for the web. She works on HTTPS adoption, Certificate Transparency, the TLS stack, and connection security UX (such as site identity in the address bar and certificate errors). Emily is also a frequent speaker at security conferences and has been a member of the Open Web Foundation's Usable Security Experts group. Emily holds a bachelor's degree from MIT, both in computer science.

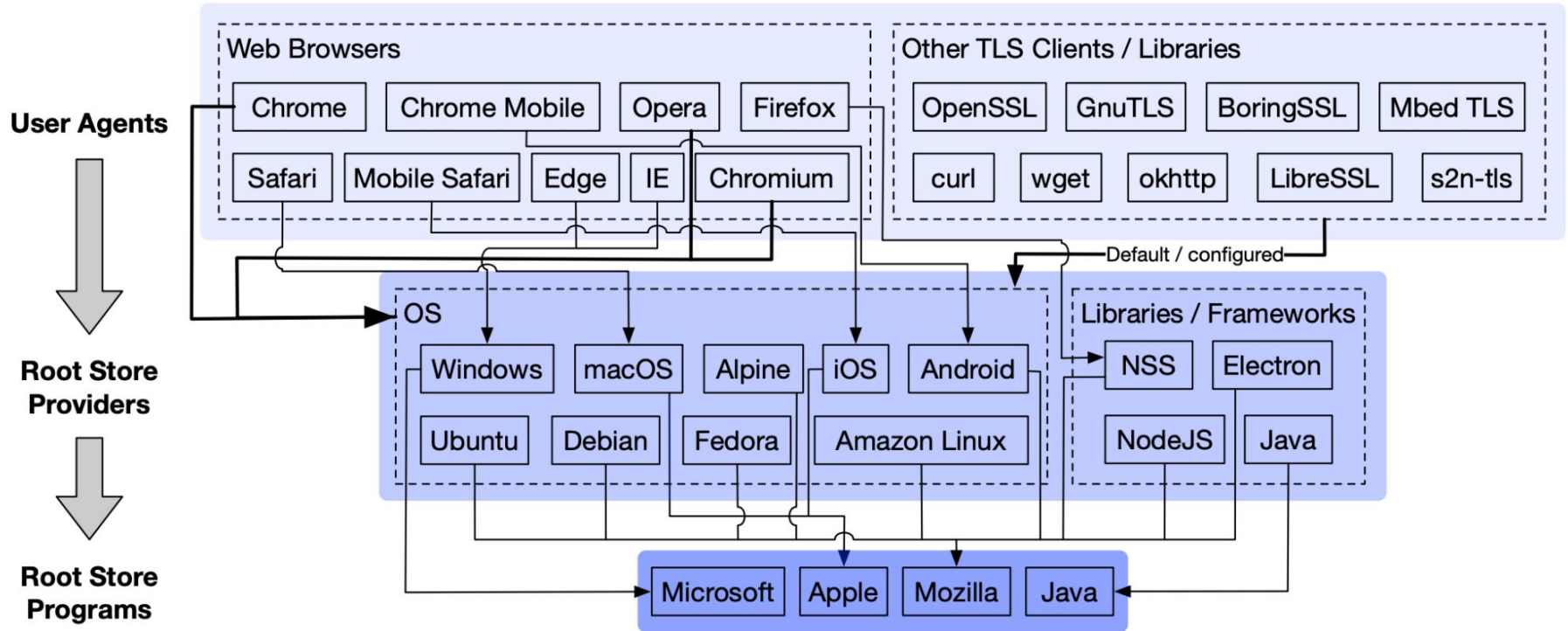
Connect: [@estark37](#)

BibTeX

PRESENTATION VIDEO



TLS root store ecosystem (circa 2021)



TPRC47 (2019)

A complete study of P.K.I. (PKI's Known Incidents)

Nicolas Serrano
School of Informatics,
Computing & Engineering
Indiana University
Bloomington
nicserr@iu.edu

Hilda Hadan
School of Informatics,
Computing & Engineering
Indiana University
Bloomington
hhadan@iu.edu

L. Jean Camp
School of Informatics,
Computing & Engineering
Indiana University
Bloomington
ljcamp@indiana.edu

Abstract—In this work, we report on a comprehensive analysis of PKI resulting from Certificate Authorities' (CAs) behavior using over 1300 instances. We found several cases where CAs designed business models that favored the issuance of digital certificates over the guidelines of the CA Forum, root management programs, and other PKI requirements. Examining PKI from the perspective of business practices, we identify a taxonomy of failures and identify systemic vulnerabilities in the governance and practices in PKI. Notorious cases include the “backdating” of digital certificates, the issuance of these for MITM attempts, the lack of verification of a requester's identity, and the unscrupulous issuance of rogue certificates. We performed a detailed study of 379 of these 1300 incidents. Using this sample, we developed a taxonomy of the different types of incidents and their causes. For each incident, we determined if the incident was disclosed by the problematic CA. We also noted the Root CA and the year of the incident. We identify the failures in terms of business practices, geography, and outcomes from CAs.

We analyzed the role of Root Program Owners (RPOs) and differentiated their policies. We identified serial and chronic offenders in the PKI trusted root programs. Some of these were distrusted by RPOs, while others remain being trusted despite failures. We also identified cases where the concentration of power of RPOs was arguably a contributing factor in the incident. We identify these cases where there is a risk of concentration of power and the resulting conflict of interests.

Our research is the first comprehensive academic study addressing all verified reported incidents. We approach this not from a machine learning or statistical perspective but, rather, we identify each reported public incident with a focus on identifying patterns of individual lapses. Here we also have a specific focus on the role of CAs and RPOs. Building on this study, we identify the issues in incentive structures that are contributors to the problems.

used. However, there have been problems with PKI. There are reasons to reconsider this trust. For example, while the mathematical foundations of the cryptography used in PKI have been studied and demonstrated to be complex to crack, advances in hardware have turned computationally secure algorithms into breakable ones. In addition, sometimes the implementation of these cryptographic algorithms introduces flaws or vulnerabilities that are external to the core crypto-mathematical function, and that can be exploited by attackers.

Sometimes, the vulnerabilities are not in the cryptographic protocols, implementing code or hardware, but in the business systems or processes that support the operations of PKI, for example, in the issuance of digital certificates. Certificates above all are a good sold in the PKI world. These miscellaneous but necessary steps that are required to obtain a digital certificate have proven to be sometimes hazardous.

Here we address the business component of PKI, examining the organizations that are the issuers of the certificates. The goal of a business is to be competitive and to make profit. The goal of a digital certificate is to bring security to its user. Therefore, digital certificates are private goods that offer security to its users and that are sold by some companies for a profit. These companies may be interested in ensuring security to people interacting with their customers after the sale, but the goal of a certificate authority (CA) is to profit from selling as many certificates as possible. It would be possible to make a theoretical argument that this is a moral hazard¹, but here we take an empirical approach to document the questionable behaviors of these companies. One common

- CA vulnerabilities are typically in the **business processes** supporting operations
 - Human error, improper security controls, misinterpretation/unaware, infrastructure problem, etc.
- Often because of their for-profit nature

Incident	#No	Total	Percentage
Fields in certificates not compliant to BR	112	146	38.52%
Non-BR-compliant ³¹ or problematic OCSP responder or CRL	33	39	10.29%
Erroneous/Misleading/Late/Lacking Audit report	24	25	6.60%
Repeated/Lacking appropriate entropy Serial Numbers	19	22	5.80%
Undisclosed SubCA	15	19	5.01%
512/1024 bits key	16	18	4.75%
Possible issuance of rogue certificates	13	18	4.75%
Use of SHA-1/MD5 hashing algorithm	13	15	3.96%
CAA ³² mis-issuance	12	14	3.69%
Rogue certificate	12	12	3.17%
CA/RA/SubCA/Reseller hacked	8	11	2.90%
Other	35	40	10.55%

TPRC47 (2019)

A complete study of P.K.I. (PKI's Known Incidents)

Nicolas Serrano
School of Informatics,
Computing & Engineering
Indiana University
Bloomington
nicserr@iu.edu

Hilda Hadan
School of Informatics,
Computing & Engineering
Indiana University
Bloomington
hhadan@iu.edu

L Jean Camp
School of Informatics,
Computing & Engineering
Indiana University
Bloomington
ljcamp@indiana.edu

Abstract—In this work, we report on a comprehensive analysis of PKI resulting from Certificate Authorities' (CAs) behavior using over 1300 instances. We found several cases where CAs designed business models that favored the issuance of digital certificates over the guidelines of the CA Forum, root management programs, and other PKI requirements. Examining PKI from the perspective of business practices, we identify a taxonomy of failures and identify systemic vulnerabilities in the governance and practices in PKI. Notorious cases include the "backdating" of digital certificates, the issuance of these for MITM attempts, the lack of verification of a requester's identity, and the unscrupulous issuance of rogue certificates. We performed a detailed study of 379 of these 1300 incidents. Using this sample, we developed a taxonomy of the different types of incidents and their causes. For each incident, we determined if the incident was disclosed by the problematic CA. We also noted the Root CA and the year of the incident. We identify the failures in terms of business practices, geography, and outcomes from CAs. We analyzed the role of Root Program Owners (RPOs) and differentiated their policies. We identified serial and chronic offenders in the PKI trusted root programs. Some of these were distrusted by RPOs, while others were not. We also identified failures. We also identified power of RPOs was an incident. We identify the concentration of power at

used. However, there have been problems with PKI. There are reasons to reconsider this trust. For example, while the mathematical foundations of the cryptography used in PKI have been studied and demonstrated to be complex to crack, advances in hardware have turned computationally secure algorithms into breakable ones. In addition, sometimes the implementation of these cryptographic algorithms introduces flaws or vulnerabilities that are external to the core crypto-mathematical function, and that can be exploited by attackers.

Sometimes, the vulnerabilities are not in the cryptographic protocols, implementing code or hardware, but in the business systems or processes that support the operations of PKI, for example, in the issuance of digital certificates. Certificates above all are a good sold in the PKI world. These miscellaneous but necessary steps that are required to obtain a digital certificate have proven to be sometimes hazardous.

Here we address the business component of PKI, examining the organizations that are the issuers of the certificates.

Country

#CAs

1

2

3

4

5

7

12

Countries with problematic root CAs

- CA vulnerabilities are typically in the **business processes** supporting operations
 - Human error, improper security controls, misinterpretation/unaware, infrastructure problem, etc.
- Often because of their for-profit nature

Incident	#No	Total	Percentage
Fields in certificates not compliant to BR	112	146	38.52%
Non-BR-compliant ³¹ or problematic OCSP responder or CRL	33	39	10.29%
Erroneous/Misleading/Late/Lacking Audit report	24	25	6.60%
Repeated/Lacking appropriate entropy Serial Numbers	19	22	5.80%
Undisclosed SubCA	15	19	5.01%
512/1024 bits key	16	18	4.75%
Possible issuance of rogue certificates	13	18	4.75%
Use of SHA-1/MD5 hashing algorithm	13	15	3.96%
CAA ³² mis-issuance	12	14	3.69%
Rogue certificate	12	12	3.17%
CA/RA/SubCA/Reseller hacked	8	11	2.90%
Other	35	40	10.55%

TPRC47 (2019)

A complete study of P.K.I. (PKI's Known Incidents)

Nicolas Serrano
School of Informatics,
Computing & Engineering
Indiana University
Bloomington
nicserr@iu.edu

Hilda Hadan
School of Informatics,
Computing & Engineering
Indiana University
Bloomington
hhadan@iu.edu

L Jean Camp
School of Informatics,
Computing & Engineering
Indiana University
Bloomington
ljcamp@indiana.edu

Abstract—In this work, we report on a comprehensive analysis of PKI resulting from Certificate Authorities' (CAs) behavior using over 1300 instances. We found several cases where CAs designed business models that favored the issuance of digital certificates over the guidelines of the CA Forum, root management programs, and other PKI requirements. Examining PKI from the perspective of business practices, we identify a taxonomy of failures and identify systemic vulnerabilities in the governance and practices in PKI. Notorious cases include the "backdating" of digital certificates, the issuance of these for MITM attempts, the lack of verification of a requester's identity, and the unscrupulous issuance of rogue certificates. We performed a detailed study of 379 of these 1300 incidents. Using this sample, we developed a taxonomy of the different types of incidents and their causes. For each incident, we determined if the incident was disclosed by the problematic CA. We also noted the Root CA and the year of the incident. We identify the failures in terms of business practices, geography, and outcomes from CAs. We analyzed the role of Root Program Owners (RPOs) and differentiated their policies. We identified serial and chronic offenders in the PKI trusted root programs. Some of these were distrusted by RPOs, while others were not. We also identified failures. We also identified power of RPOs was an incident. We identify the concentration of power at

Our research is the addressing all verified this not from a machine but, rather, we identify a focus on identifying patterns have a specific focus on on this study, we identify are contributors to the p

used. However, there have been problems with PKI. There are reasons to reconsider this trust. For example, while the mathematical foundations of the cryptography used in PKI have been studied and demonstrated to be complex to crack, advances in hardware have turned computationally secure algorithms into breakable ones. In addition, sometimes the implementation of these cryptographic algorithms introduces flaws or vulnerabilities that are external to the core crypto-mathematical function, and that can be exploited by attackers.

Sometimes, the vulnerabilities are not in the cryptographic protocols, implementing code or hardware, but in the business systems or processes that support the operations of PKI, for example, in the issuance of digital certificates. Certificates above all are a good sold in the PKI world. These miscellaneous but necessary steps that are required to obtain a digital certificate have proven to be sometimes hazardous.

Here we address the business component of PKI, examining the organizations that are the issuers of the certificates.

Country	#CAs
	1
	2
	3
	4
	5
	7
USA	12

Countries with problematic root CAs

- CA vulnerabilities are typically in the **business processes** supporting operations
 - Human error, improper security controls, misinterpretation/unaware, infrastructure problem, etc.
- Often because of their for-profit nature

Incident	#No	Total	Percentage
Fields in certificates not compliant to BR	112	146	38.52%
Non-BR-compliant ³¹ or problematic OCSP responder or CRL	33	39	10.29%
Erroneous/Misleading/Late/Lacking Audit report	24	25	6.60%
Repeated/Lacking appropriate entropy Serial Numbers	19	22	5.80%
Undisclosed SubCA	15	19	5.01%
512/1024 bits key	16	18	4.75%
Possible issuance of rogue certificates	13	18	4.75%
Use of SHA-1/MD5 hashing algorithm	13	15	3.96%
CAA ³² mis-issuance	12	14	3.69%
Rogue certificate	12	12	3.17%
CA/RA/SubCA/Reseller hacked	8	11	2.90%
Other	35	40	10.55%

TPRC47 (2019)

A complete study of P.K.I. (PKI's Known Incidents)

Nicolas Serrano
School of Informatics,
Computing & Engineering
Indiana University
Bloomington
nicserr@iu.edu

Hilda Hadan
School of Informatics,
Computing & Engineering
Indiana University
Bloomington
hhadan@iu.edu

L Jean Camp
School of Informatics,
Computing & Engineering
Indiana University
Bloomington
ljcamp@indiana.edu

Abstract—In this work, we report on a comprehensive analysis of PKI resulting from Certificate Authorities' (CAs) behavior using over 1300 instances. We found several cases where CAs designed business models that favored the issuance of digital certificates over the guidelines of the CA Forum, root management programs, and other PKI requirements. Examining PKI from the perspective of business practices, we identify a taxonomy of failures and identify systemic vulnerabilities in the governance and practices in PKI. Notorious cases include the "backdating" of digital certificates, the issuance of these for MITM attempts, the lack of verification of a requester's identity, and the unscrupulous issuance of rogue certificates. We performed a detailed study of 379 of these 1300 incidents. Using this sample, we developed a taxonomy of the different types of incidents and their causes. For each incident, we determined if the incident was disclosed by the problematic CA. We also noted the Root CA and the year of the incident. We identify the failures in terms of business practices, geography, and outcomes from CAs. We analyzed the role of Root Program Owners (RPOs) and differentiated their policies. We identified serial and chronic offenders in the PKI trusted root programs. Some of these were distrusted by RPOs, while others were not. We also identified failures. We also identified power of RPOs was an incident. We identify the concentration of power at

used. However, there have been problems with PKI. There are reasons to reconsider this trust. For example, while the mathematical foundations of the cryptography used in PKI have been studied and demonstrated to be complex to crack, advances in hardware have turned computationally secure algorithms into breakable ones. In addition, sometimes the implementation of these cryptographic algorithms introduces flaws or vulnerabilities that are external to the core crypto-mathematical function, and that can be exploited by attackers.

Sometimes, the vulnerabilities are not in the cryptographic protocols, implementing code or hardware, but in the business systems or processes that support the operations of PKI, for example, in the issuance of digital certificates. Certificates above all are a good sold in the PKI world. These miscellaneous but necessary steps that are required to obtain a digital certificate have proven to be sometimes hazardous.

Here we address the business component of PKI, examining the organizations that are the issuers of the certificates.

Country	#CAs
	1
	2
	3
	4
	5
Spain	7
USA	12

Countries with problematic root CAs

- CA vulnerabilities are typically in the **business processes** supporting operations
 - Human error, improper security controls, misinterpretation/unaware, infrastructure problem, etc.
- Often because of their for-profit nature

Incident	#No	Total	Percentage
Fields in certificates not compliant to BR	112	146	38.52%
Non-BR-compliant ³¹ or problematic OCSP responder or CRL	33	39	10.29%
Erroneous/Misleading/Late/Lacking Audit report	24	25	6.60%
Repeated/Lacking appropriate entropy Serial Numbers	19	22	5.80%
Undisclosed SubCA	15	19	5.01%
512/1024 bits key	16	18	4.75%
Possible issuance of rogue certificates	13	18	4.75%
Use of SHA-1/MD5 hashing algorithm	13	15	3.96%
CAA ³² mis-issuance	12	14	3.69%
Rogue certificate	12	12	3.17%
CA/RA/SubCA/Reseller hacked	8	11	2.90%
Other	35	40	10.55%

TPRC47 (2019)

A complete study of P.K.I. (PKI's Known Incidents)

Nicolas Serrano
School of Informatics,
Computing & Engineering
Indiana University
Bloomington
nicserr@iu.edu

Hilda Hadan
School of Informatics,
Computing & Engineering
Indiana University
Bloomington
hhadan@iu.edu

L Jean Camp
School of Informatics,
Computing & Engineering
Indiana University
Bloomington
ljcamp@indiana.edu

Abstract—In this work, we report on a comprehensive analysis of PKI resulting from Certificate Authorities' (CAs) behavior using over 1300 instances. We found several cases where CAs designed business models that favored the issuance of digital certificates over the guidelines of the CA Forum, root management programs, and other PKI requirements. Examining PKI from the perspective of business practices, we identify a taxonomy of failures and identify systemic vulnerabilities in the governance and practices in PKI. Notorious cases include the "backdating" of digital certificates, the issuance of these for MITM attempts, the lack of verification of a requester's identity, and the unscrupulous issuance of rogue certificates. We performed a detailed study of 379 of these 1300 incidents. Using this sample, we developed a taxonomy of the different types of incidents and their causes. For each incident, we determined if the incident was disclosed by the problematic CA. We also noted the Root CA and the year of the incident. We identify the failures in terms of business practices, geography, and outcomes from CAs.

We analyzed the role of Root Program Owners (RPOs) and differentiated their policies. We identified serial and chronic offenders in the PKI trusted root programs. Some of these were distrusted by RPOs, while others were not. We also identified failures. We also identified power of RPOs was an incident. We identify the concentration of power at

Our research is the addressing all verified this not from a machine but, rather, we identify focus on identifying patterns have a specific focus on on this study, we identify are contributors to the p

used. However, there have been problems with PKI. There are reasons to reconsider this trust. For example, while the mathematical foundations of the cryptography used in PKI have been studied and demonstrated to be complex to crack, advances in hardware have turned computationally secure algorithms into breakable ones. In addition, sometimes the implementation of these cryptographic algorithms introduces flaws or vulnerabilities that are external to the core crypto-mathematical function, and that can be exploited by attackers.

Sometimes, the vulnerabilities are not in the cryptographic protocols, implementing code or hardware, but in the business systems or processes that support the operations of PKI, for example, in the issuance of digital certificates. Certificates above all are a good sold in the PKI world. These miscellaneous but necessary steps that are required to obtain a digital certificate have proven to be sometimes hazardous.

Here we address the business component of PKI, examining the organizations that are the issuers of the certificates.

Country	#CAs
	1
	2
	3
	4
France, Turkey	5
Spain	7
USA	12

Countries with problematic root CAs

- CA vulnerabilities are typically in the **business processes** supporting operations
 - Human error, improper security controls, misinterpretation/unaware, infrastructure problem, etc.
- Often because of their for-profit nature

Incident	#No	Total	Percentage
Fields in certificates not compliant to BR	112	146	38.52%
Non-BR-compliant ³¹ or problematic OCSP responder or CRL	33	39	10.29%
Erroneous/Misleading/Late/Lacking Audit report	24	25	6.60%
Repeated/Lacking appropriate entropy Serial Numbers	19	22	5.80%
Undisclosed SubCA	15	19	5.01%
512/1024 bits key	16	18	4.75%
Possible issuance of rogue certificates	13	18	4.75%
Use of SHA-1/MD5 hashing algorithm	13	15	3.96%
CAA ³² mis-issuance	12	14	3.69%
Rogue certificate	12	12	3.17%
CA/RA/SubCA/Reseller hacked	8	11	2.90%
Other	35	40	10.55%

TPRC47 (2019)

A complete study of P.K.I. (PKI's Known Incidents)

Nicolas Serrano
School of Informatics,
Computing & Engineering
Indiana University
Bloomington
nicserran@iu.edu

Hilda Hadan
School of Informatics,
Computing & Engineering
Indiana University
Bloomington
hhadan@iu.edu

L Jean Camp
School of Informatics,
Computing & Engineering
Indiana University
Bloomington
ljcamp@indiana.edu

Abstract—In this work, we report on a comprehensive analysis of PKI resulting from Certificate Authorities' (CAs) behavior using over 1300 instances. We found several cases where CAs designed business models that favored the issuance of digital certificates over the guidelines of the CA Forum, root management programs, and other PKI requirements. Examining PKI from the perspective of business practices, we identify a taxonomy of failures and identify systemic vulnerabilities in the governance and practices in PKI. Notorious cases include the "backdating" of digital certificates, the issuance of these for MITM attempts, the lack of verification of a requester's identity, and the unscrupulous issuance of rogue certificates. We performed a detailed study of 379 of these 1300 incidents. Using this sample, we developed a taxonomy of the different types of incidents and their causes. For each incident, we determined if the incident was disclosed by the problematic CA. We also noted the Root CA and the year of the incident. We identify the failures in terms of business practices, geography, and outcomes from CAs. We analyzed the role of Root Program Owners (RPOs) and differentiated their policies. We identified serial and chronic offenders in the PKI trusted root programs. Some of these were distrusted by RPOs, while others were not. We also identified failures. We also identified power of RPOs was an incident. We identify the concentration of power at

used. However, there have been problems with PKI. There are reasons to reconsider this trust. For example, while the mathematical foundations of the cryptography used in PKI have been studied and demonstrated to be complex to crack, advances in hardware have turned computationally secure algorithms into breakable ones. In addition, sometimes the implementation of these cryptographic algorithms introduces flaws or vulnerabilities that are external to the core crypto-mathematical function, and that can be exploited by attackers.

Sometimes, the vulnerabilities are not in the cryptographic protocols, implementing code or hardware, but in the business systems or processes that support the operations of PKI, for example, in the issuance of digital certificates. Certificates above all are a good sold in the PKI world. These miscellaneous but necessary steps that are required to obtain a digital certificate have proven to be sometimes hazardous. Here we address the business component of PKI, examining the organizations that are the issuers of the certificates.

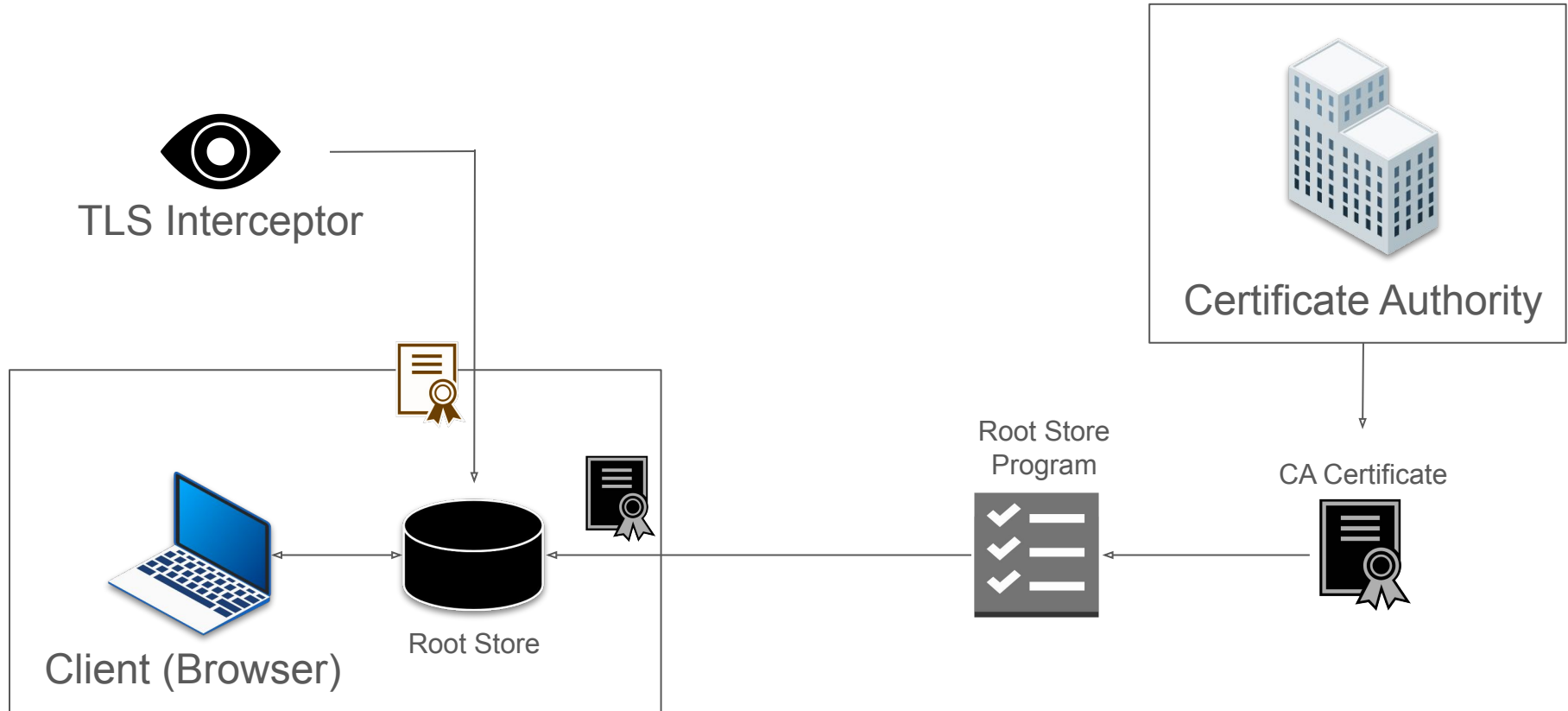
Country	#CAs
Belgium, Bermuda, Canada, Colombia, Estonia, Finland, Hong Kong, India, Ireland, Italy, Kazakhstan, Korea, Romania, Slovak Republic, South Africa, Venezuela	1
Hungary, Japan, Poland, Taiwan	2
Germany, Netherlands, Switzerland, UK	3
China	4
France, Turkey	5
Spain	7
USA	12

Countries with problematic root CAs

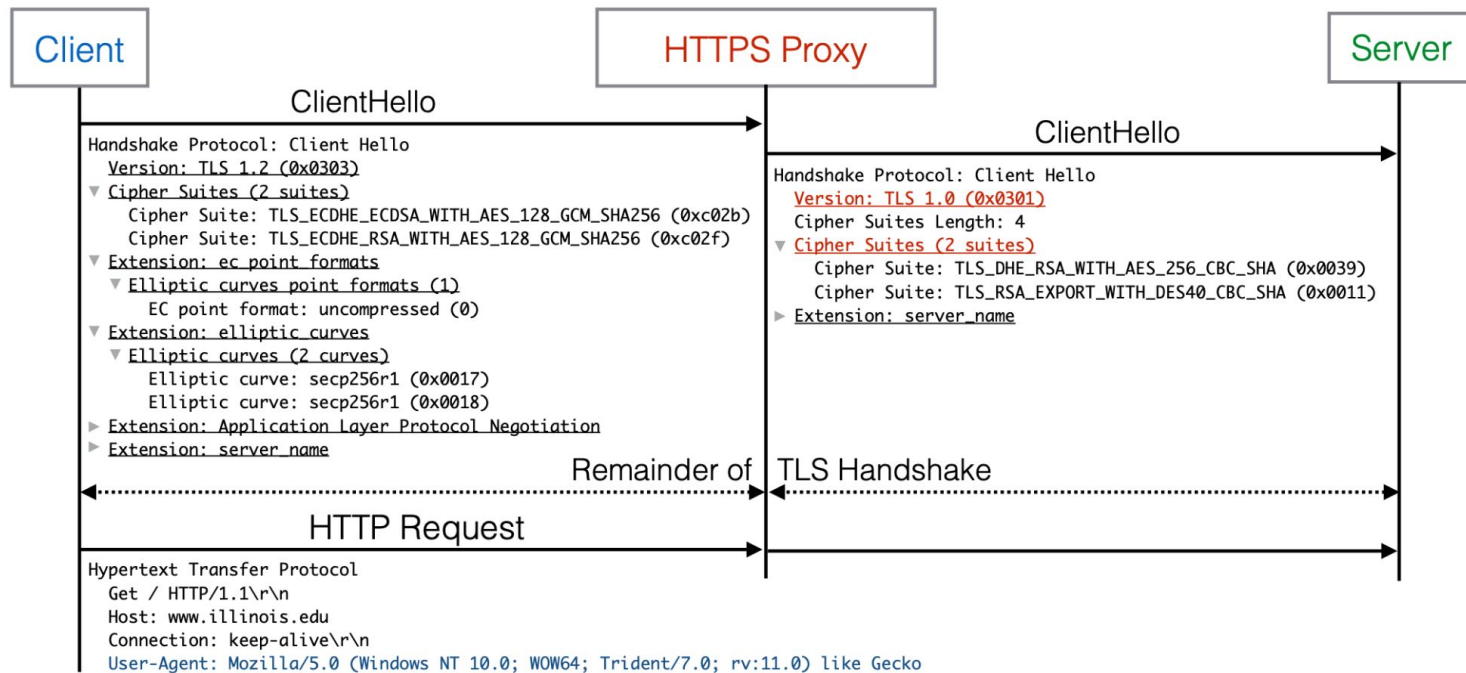
- CA vulnerabilities are typically in the **business processes** supporting operations
 - Human error, improper security controls, misinterpretation/unaware, infrastructure problem, etc.
- Often because of their for-profit nature

Incident	#No	Total	Percentage
Fields in certificates not compliant to BR	112	146	38.52%
Non-BR-compliant ³¹ or problematic OCSP responder or CRL	33	39	10.29%
Erroneous/Misleading/Late/Lacking Audit report	24	25	6.60%
Repeated/Lacking appropriate entropy Serial Numbers	19	22	5.80%
Undisclosed SubCA	15	19	5.01%
512/1024 bits key	16	18	4.75%
Possible issuance of rogue certificates	13	18	4.75%
Use of SHA-1/MD5 hashing algorithm	13	15	3.96%
CAA ³² mis-issuance	12	14	3.69%
Rogue certificate	12	12	3.17%
CA/RA/SubCA/Reseller hacked	8	11	2.90%
Other	35	40	10.55%

Trust Management Revisited: **TLS interception**



Trust Management Revisited: **TLS interception**



Trust Management Revisited: **TLS interception**

Country	MITM %	Country	MITM %
Guatemala	15.0%	Kiribati	8.2%
Greenland	9.9%	Iran	8.1%
South Korea	8.8%	Tanzania	7.3%
Kuwait	8.5%	Bahrain	7.3%
Qatar	8.4%	Afghanistan	6.7%

Fig. 10: **Countries with Highest Firefox Interception**—We show the ten countries with the highest interception rates when connecting to the Mozilla update server. Countries with above average interception rates generally have a large amount of traffic intercepted by a single, dominant mobile provider.

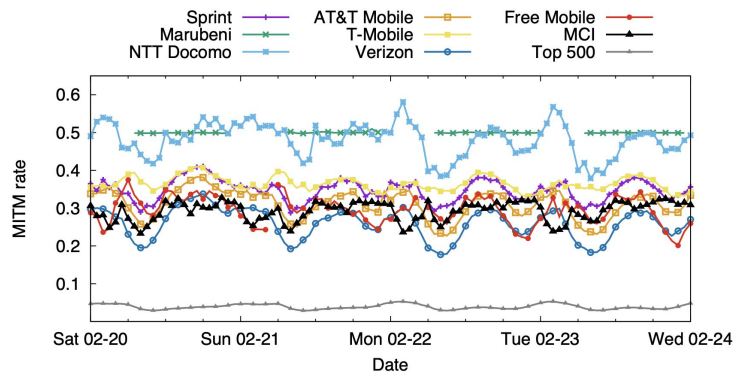


Fig. 9: **ASes with Highest Firefox Interception**—We find that 8 ASes have significantly higher interception rates within the top 500 ASes. All but one are mobile providers.

Trust Management Revisited: **TLS interception**

- 4%-10% global HTTPS traffic intercepted
- Nearly all interception products introduce vulnerabilities
- Injected roots are common and operated by CAs with poor security

Product	Grade	Validates Certificates	Modern Ciphers	Advertises RC4	TLS Version	Grading Notes
A10 vThunder SSL Insight	F	✓	✓	Yes	1.2	Advertises export ciphers
Blue Coat ProxySG 6642	A*	✓	✓	No	1.2	Mirrors client ciphers
Barracuda 610Vx Web Filter	C	✓	✗	Yes	1.0	Vulnerable to Logjam attack
Checkpoint Threat Prevention	F	✓	✗	Yes	1.0	Allows expired certificates
Cisco IronPort Web Security	F	✓	✓	Yes	1.2	Advertises export ciphers
Forcepoint TRITON AP-WEB Cloud	C	✓	✓	No	1.2	Accepts RC4 ciphers
Fortinet FortiGate 5.4.0	C	✓	✓	No	1.2	Vulnerable to Logjam attack
Juniper SRX Forward SSL Proxy	C	✓	✗	Yes	1.2	Advertises RC4 ciphers
Microsoft Threat Mgmt. Gateway	F	✗	✗	Yes	SSLv2	No certificate validation
Sophos SSL Inspection	C	✓	✓	Yes	1.2	Advertises RC4 ciphers
Untangle NG Firewall	C	✓	✗	Yes	1.2	Advertises RC4 ciphers
WebTitan Gateway	F	✗	✓	Yes	1.2	Broken certificate validation

Z. Durumeric, Z. Ma, D. Springall, R. Barnes, N. Sullivan, E. Burszstein, M. Bailey, J. Alex Halderman, V. Paxson. The Security Impact of HTTPS Interception. NDSS 2017

Parting thoughts

Recap: **putting it all together**

1. Observability of networking metadata is a moving target because of the adoption of newer privacy technologies
2. Different privacy technologies focus on different technical goals. It's important to know the strengths and weaknesses of each technology
3. Tracking moved to other (both up and down) layers
 - a. Commercial surveillance
 - b. User & device fingerprinting

From the early manifestos ...

- Many of the designers of the Internet held strong views about cyberspace and what it should be
- Themes pervasive in hacker culture: unrestricted exploration of the possibilities of technology, freedom of information, anti-authoritarianism, etc.
- Read, e.g.:
 - John Perry Barlow's "Declaration of Independence of Cyberspace"
 - The Mentor's "Hacker Manifesto"
- Cypherpunks and the Crypto Wars

... to regulating cyberspace

- But many believe that cyberspace marked the beginning of a new era with more and more human activities taking place there
- Significant challenges for states to exercise control and practice sovereignty
 - In part rooted in the technology itself
 - In part rooted in the privately-owned nature of cyberspace
- Non-Western countries such as Russia, China or NK have a fundamentally different approach to dealing with these issues
- In the US and the EU: waves of regulatory efforts
 - NIS2, DSA, EIDAS, CRA, DMA, Chat Control, Age Verification, DNS4EU
 - National Security laws (e.g., the Patriot Act)

Thank you for listening.

Questions? Comments? Thoughts?