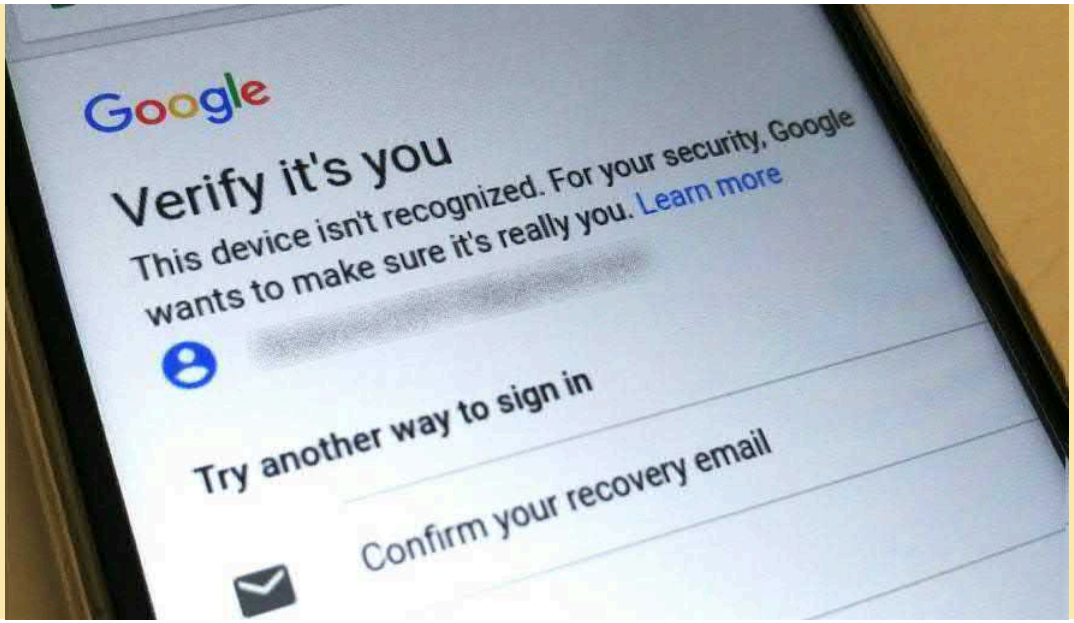


Evaluation of Risk-based Re-Authentication Methods

Stephan Wiefling*#, Tanvi Patil+, Markus Dürmuth#, Luigi Lo Iacono*

H-BRS University of Applied Sciences (*)

Ruhr University Bochum (#)

UNC Charlotte (+)

Current practice*

- Email verification
- Six-digit code
  - Major impact on time exposure and usability
  - But not studied so far!

Service	Requested authentication factors
Amazon	▪ **Verification code** (**email***, text message)
Facebook	▪ Approve login on another computer ▪ Identify photos of friends ▪ Asking friends for help ▪ **Verification code** (text message)
GOG.com	▪ **Verification code** (**email**)*
Google	▪ Enter the city you usually sign in from ▪ **Verification code** (**email**, text message, app, phone call) ▪ Press confirmation button on second device
LinkedIn	▪ **Verification code** (**email**)*

*Wiefling et al.: Is This Really You? An Empirical Study on Risk-Based Authentication Applied in the Wild. In: IFIP SEC '19. Springer (2019)

Study Procedure

- 1. Registration
- 2. Login
- 3. Exit survey

Study Procedure

- 1. Registration
- **2. Login**
  - Re-Authentication requested
  - Method differed in each condition
- 3. Exit survey

Method 1: State of the Art (in use)

- Code-based method
- Code in email body

Your personal security code

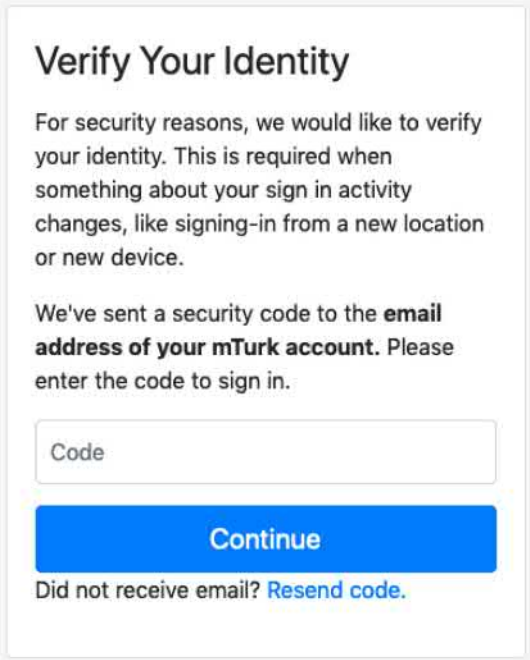
Dear [redacted] user,
Someone just tried to sign in to your [redacted] account.

If you were prompted for a security code, please enter the following to complete your sign in:

166832

If you were not prompted, please change your password immediately in the profile settings of cloust.de.

Thanks, the [redacted] Team



Method 2: Subject Line (new)

- Code-based method
- Code in email body **and subject line**

966601 is your personal security code

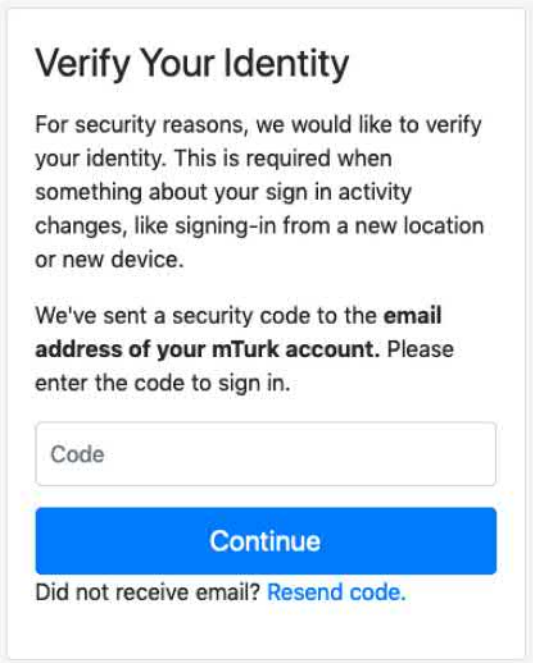
Dear [redacted] user,
Someone just tried to sign in to your [redacted] account.

If you were prompted for a security code, please enter the following to complete your sign in:

166832

If you were not prompted, please change your password immediately in the profile settings of cloust.de.

Thanks, the [redacted] Team



Method 3: Link (new)

- Link-based method
- Verification link in email body

Your personal confirmation link

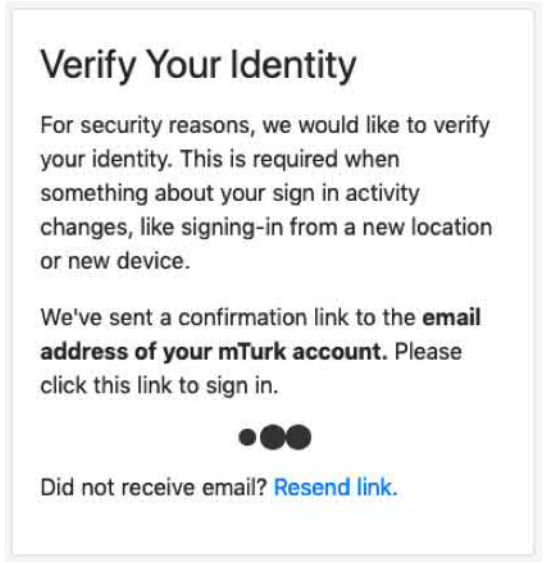
Dear [redacted] user,
Someone just tried to sign in to your [redacted] account.

If you were prompted to open a confirmation link, please click the link below to complete your sign in:

https://[redacted]/verify/vxno8ykjdyabx5zweuvoanqe42vgv0nj

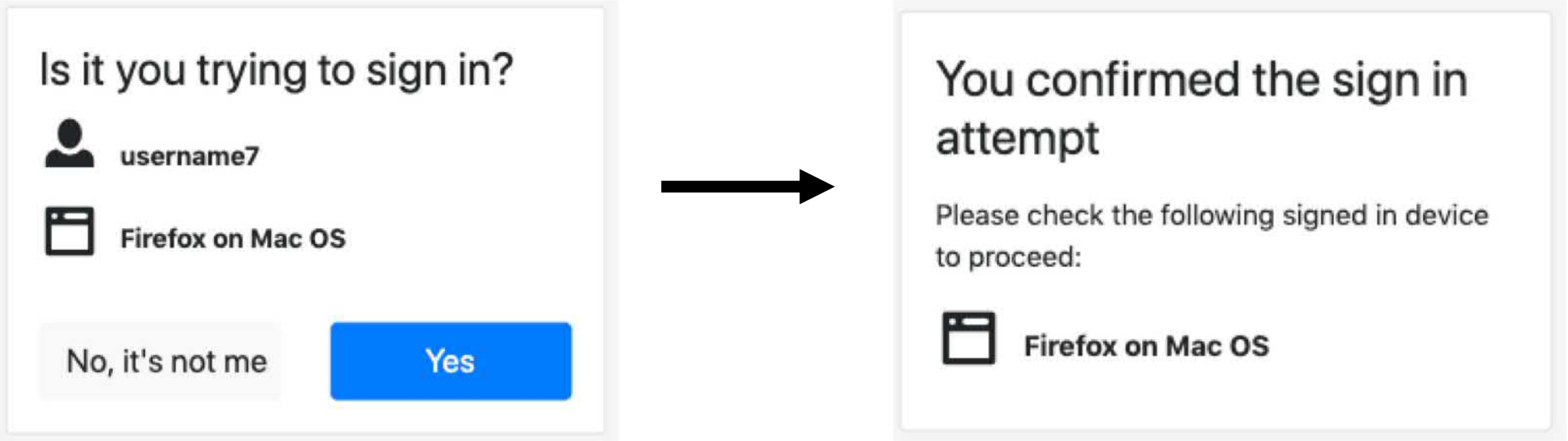
This link expires in 15 minutes.

If you were not prompted, please change your password immediately in the profile settings of cloust.de.
Thanks, the [redacted] Team



Method 3: Link (new)

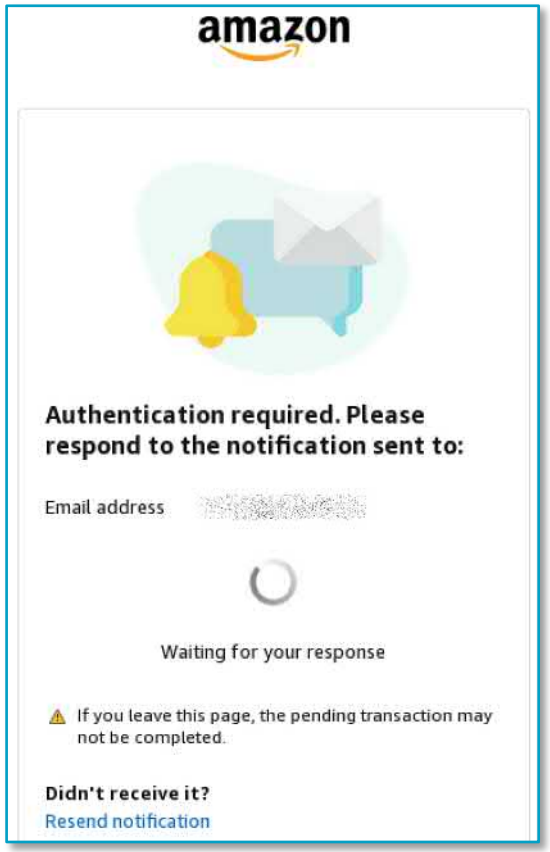
- Extra confirmation when confirmation device is different*

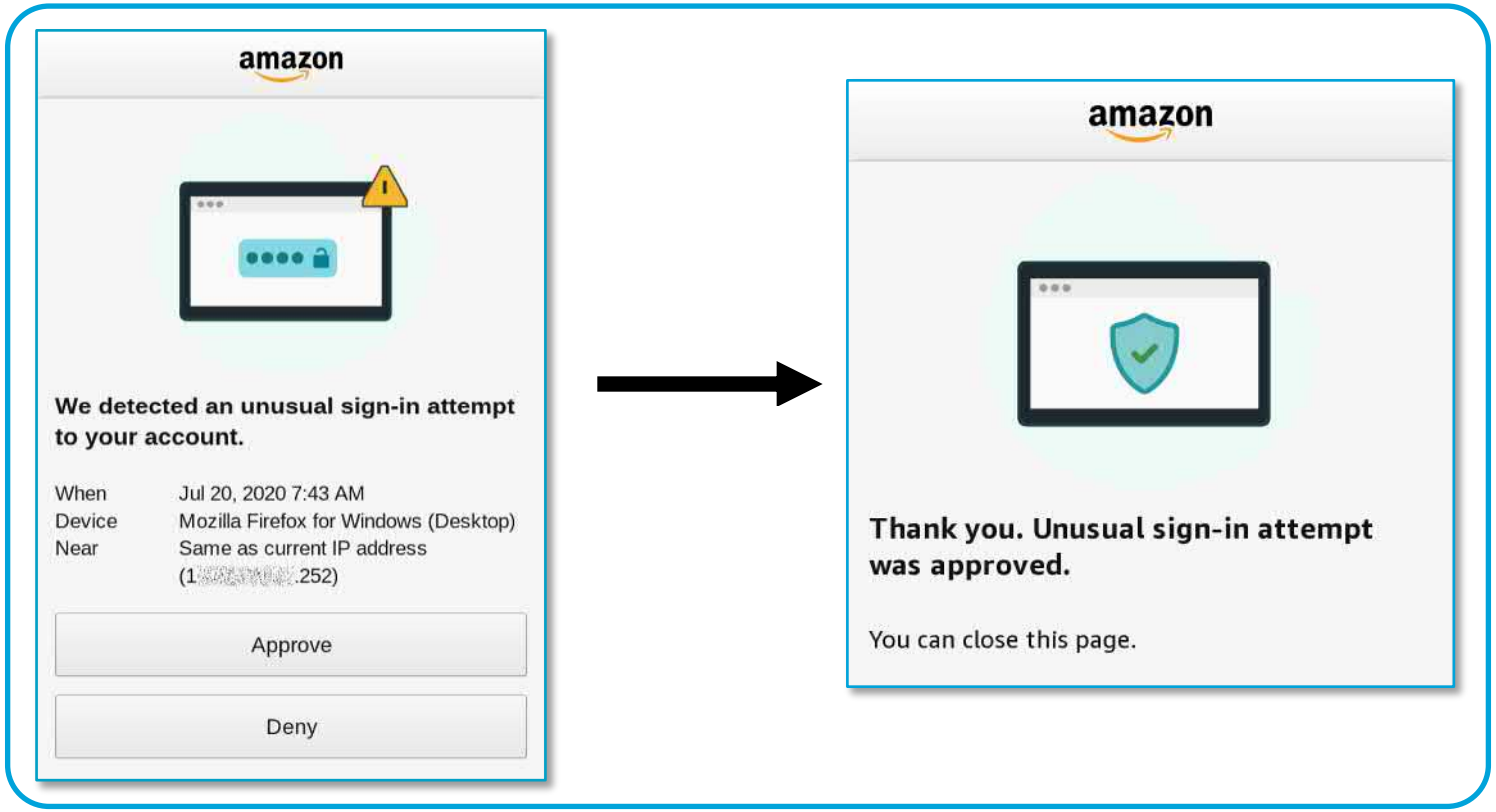


*Based on Google's Android device confirmation dialog

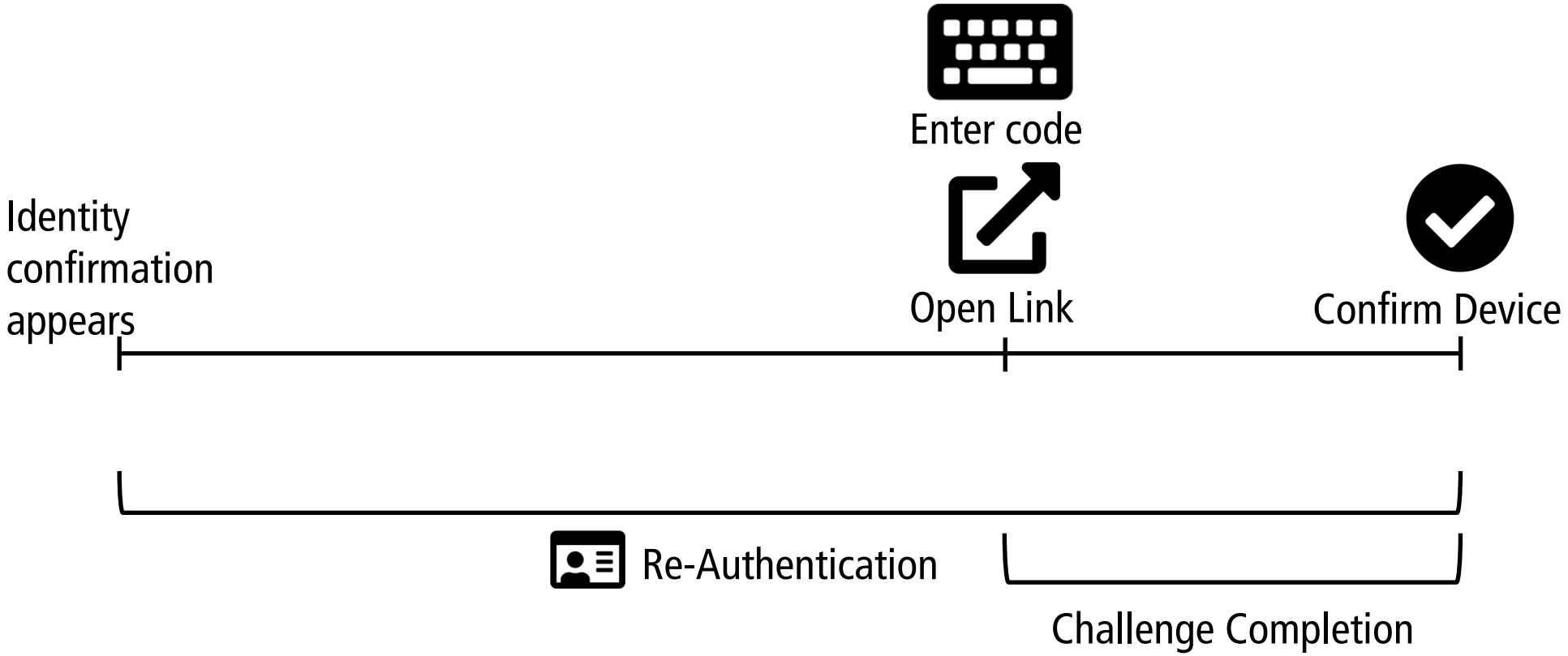
Method 3: Link (new)

- Amazon deployed method one year after our study

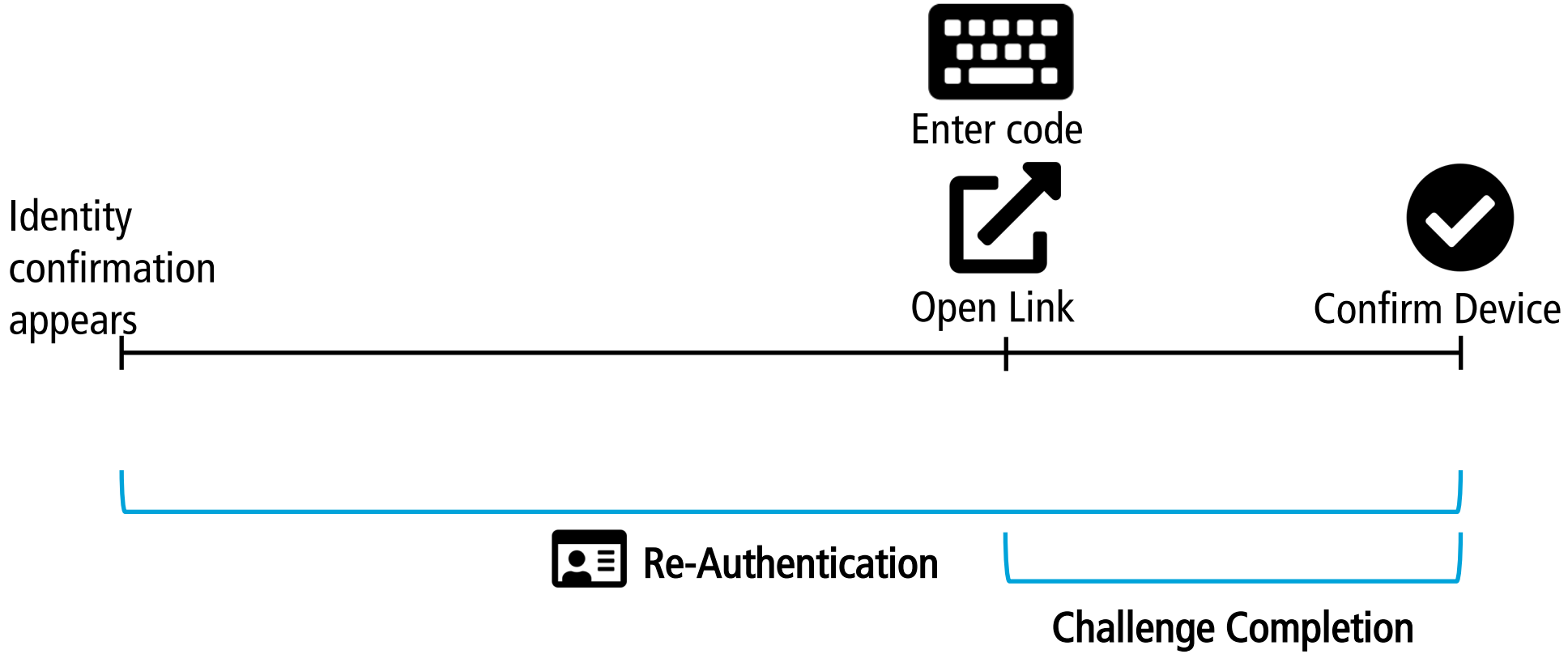




Timings: Measurement



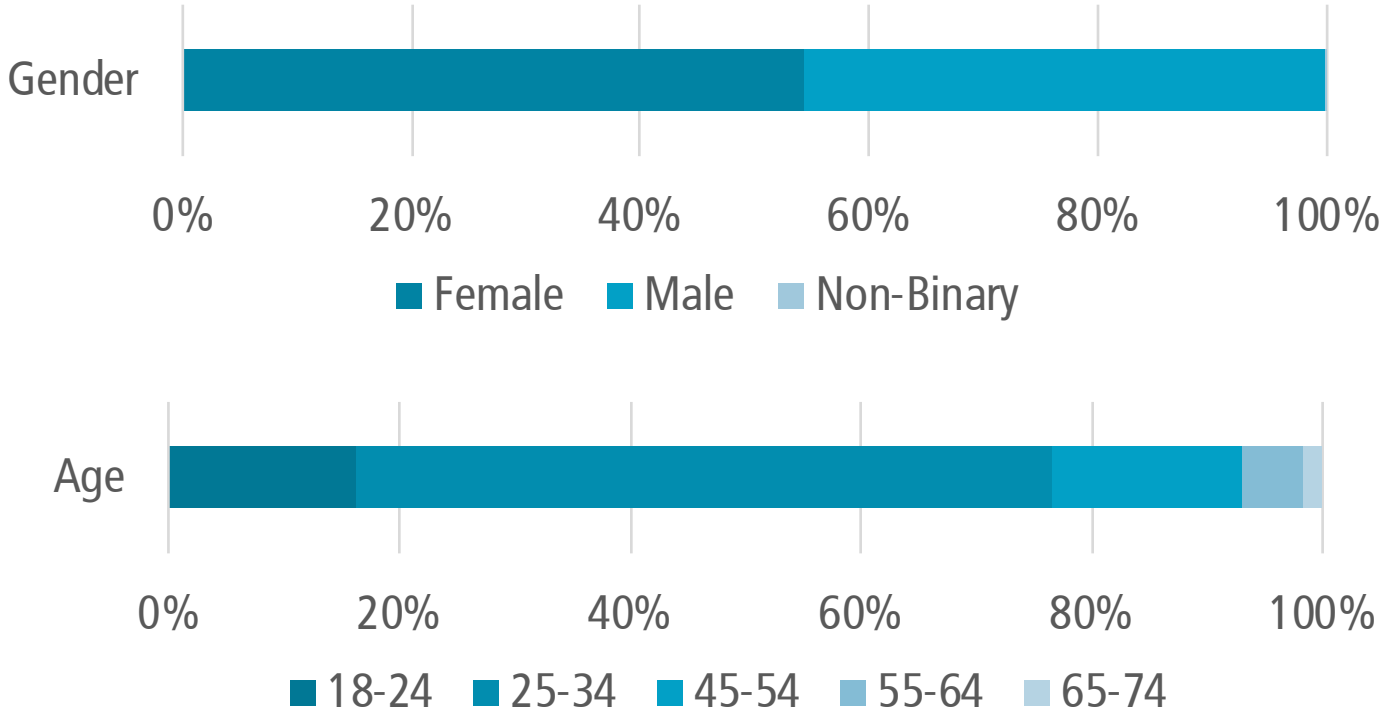
Timings: Measurement



Study Procedure

- 1. Registration
- 2. Login
- **3. Exit survey**

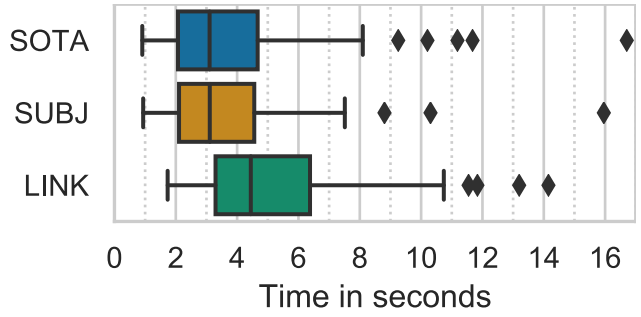
Results: Demographics (n=451)



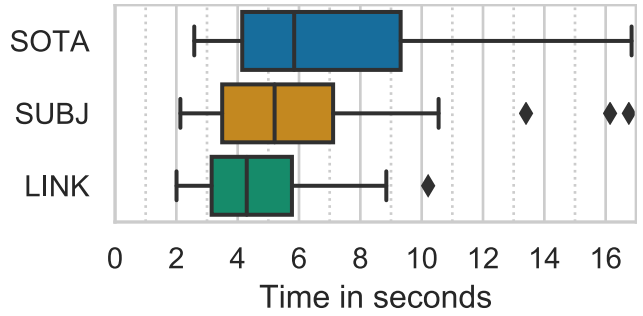
- Associate degree or higher (63%)
- No computer science background (74%)

Results: Challenge Completion Time

- Faster in two cases (each $p<0.01$)
  - Code-based: Desktop PC for login + re-authentication
  - Link-based: Desktop PC for login, mobile device for re-authentication



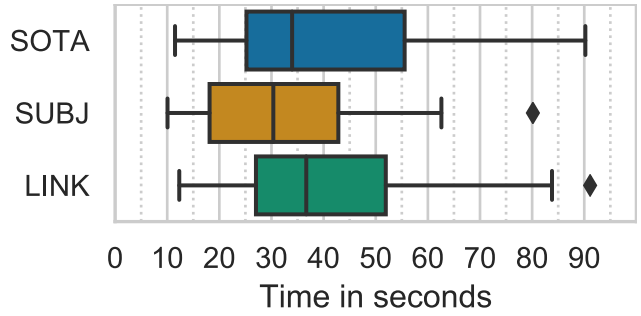
Desktop/Desktop



Desktop/Mobile

Results: Re-Authentication Time

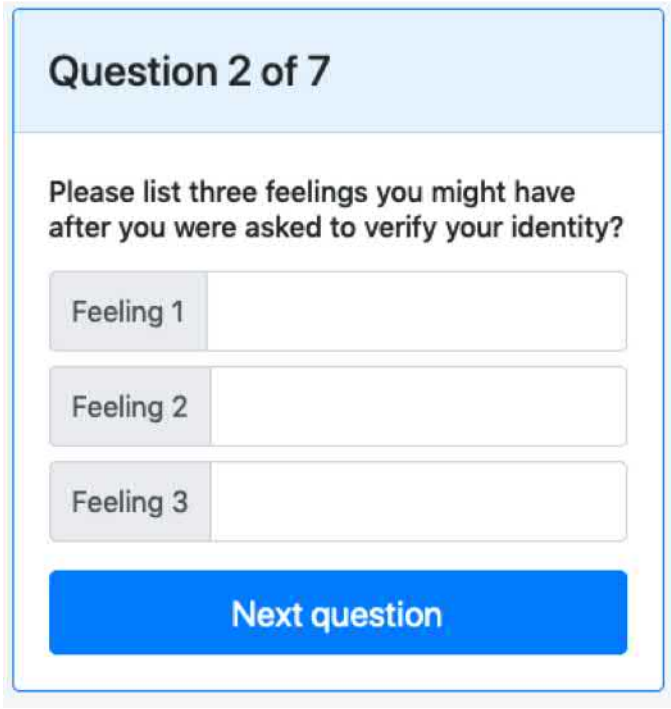
- Faster with code in subject line and body
  - Desktop PC for login + re-authentication (p=0.02)



Desktop/Desktop

Results: Feelings

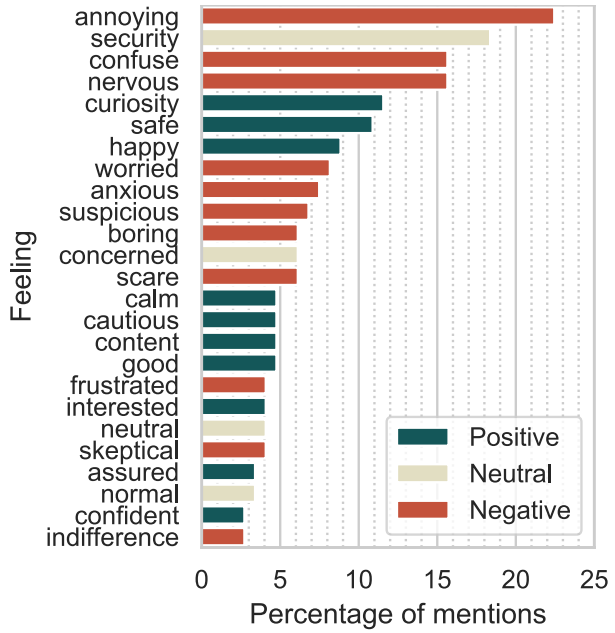
- Question in exit survey*



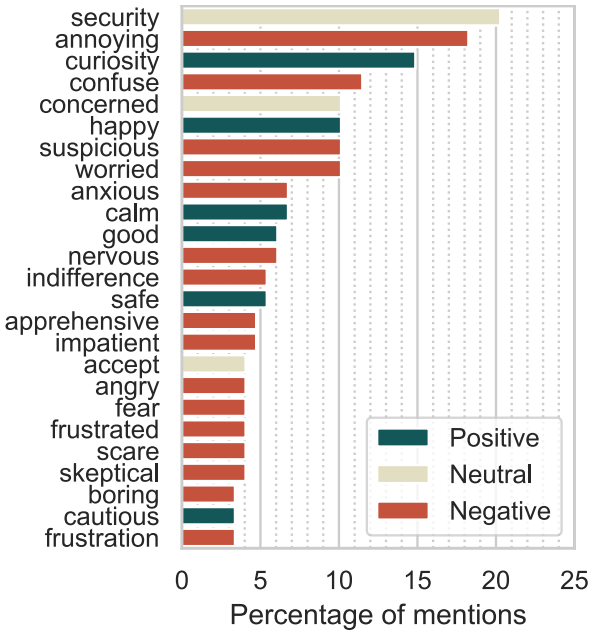
*Question similar to Golla et al. (CCS '18)

Results: Feelings

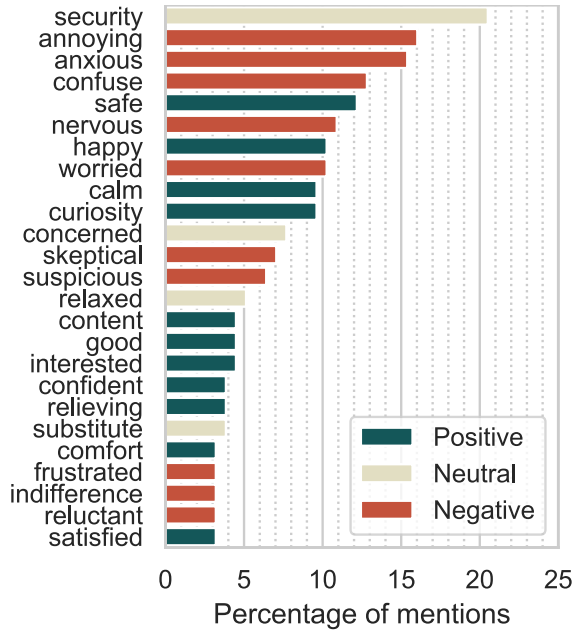
- Similar number of mentions in all conditions
- With exceptions



State of the art (Code in body)



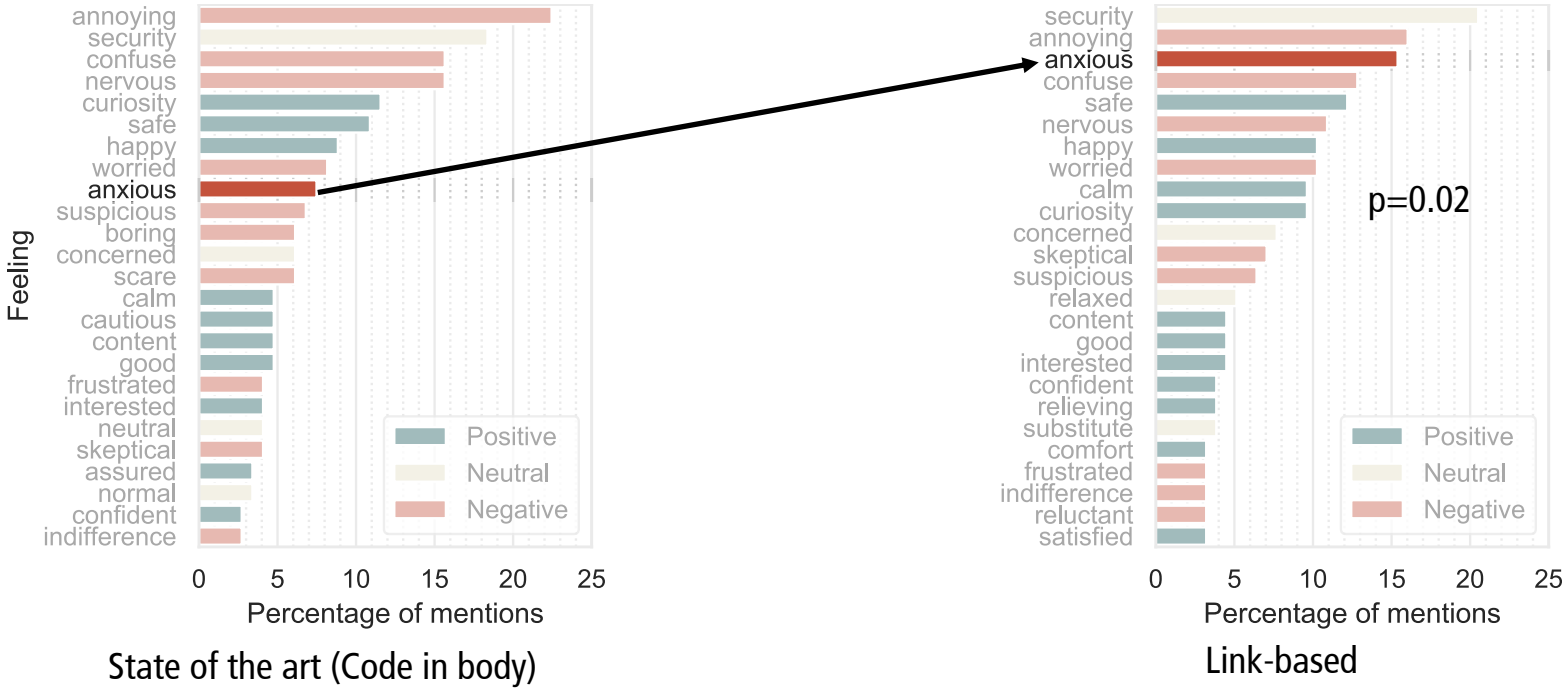
Code in body + subject line



Link-based

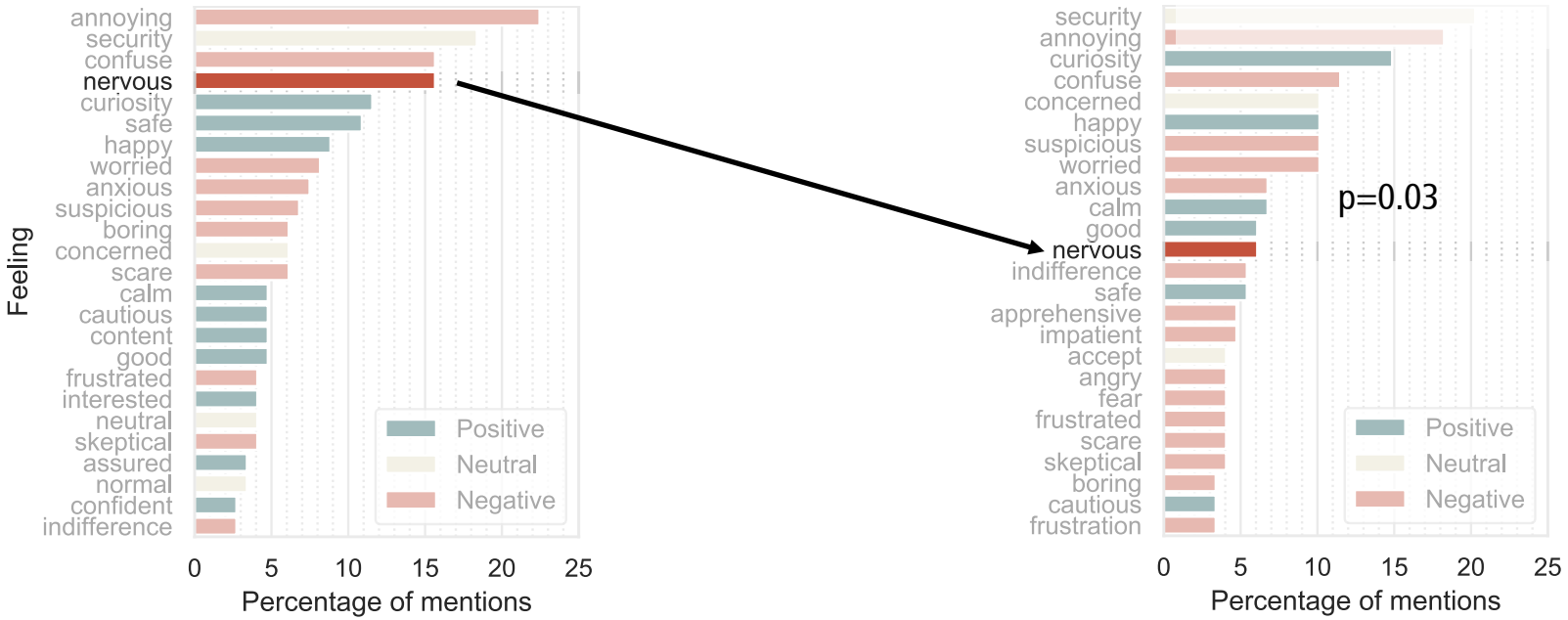
Results: Feelings

- Link-based method made users significantly more anxious than code-based methods



Results: Feelings

- Code in subject line and body made significantly less nervous



Conclusion

- Consider RBA on websites with sensitive data involved
  - Exception: Online banking

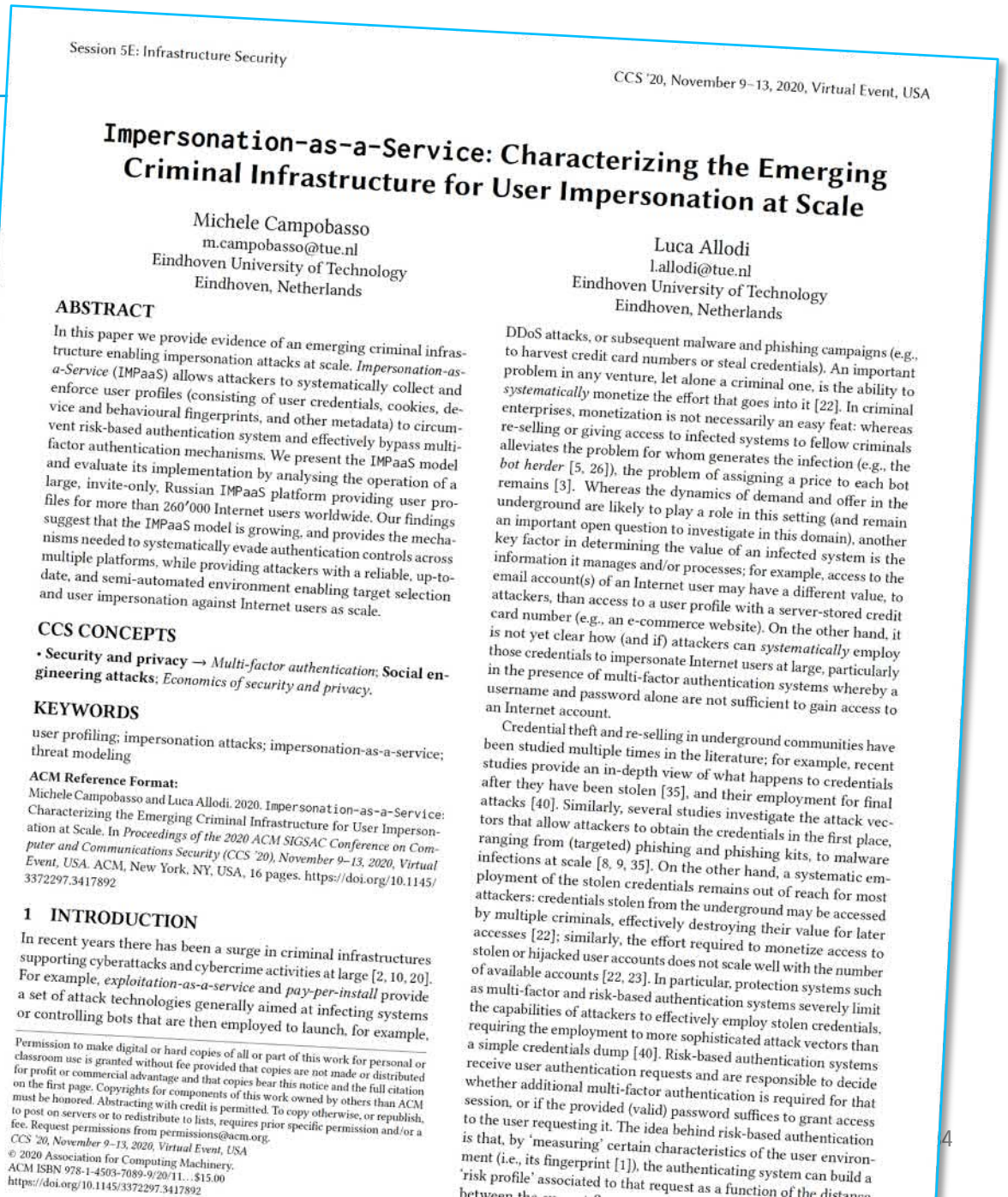
- RBA using email mostly accepted

- Beware of deadlocks

Hackers adapt*

- Digital fingerprints for sale
  - Malware infected devices
  - IP address spoofing via proxy
  - Aim: Bypass RBA
- Can we protect our users against these type of attacks?

* Campobasso et al..: Impersonation-as-a-Service: Characterizing the Emerging Criminal Infrastructure for User Impersonation at Scale. In: CCS '20. ACM (2020)



Session 5E: Infrastructure Security CCS '20, November 9–13, 2020, Virtual Event, USA

Impersonation-as-a-Service: Characterizing the Emerging Criminal Infrastructure for User Impersonation at Scale

Michele Campobasso
m.campobasso@tue.nl
Eindhoven University of Technology
Eindhoven, Netherlands

Luca Allodi
l.allodi@tue.nl
Eindhoven University of Technology
Eindhoven, Netherlands

ABSTRACT

In this paper we provide evidence of an emerging criminal infrastructure enabling impersonation attacks at scale. *Impersonation-as-a-Service* (IMPaaS) allows attackers to systematically collect and enforce user profiles (consisting of user credentials, cookies, device and behavioural fingerprints, and other metadata) to circumvent risk-based authentication system and effectively bypass multi-factor authentication mechanisms. We present the IMPaaS model and evaluate its implementation by analysing the operation of a large, invite-only, Russian IMPaaS platform providing user profiles for more than 260'000 Internet users worldwide. Our findings suggest that the IMPaaS model is growing, and provides the mechanisms needed to systematically evade authentication controls across multiple platforms, while providing attackers with a reliable, up-to-date, and semi-automated environment enabling target selection and user impersonation against Internet users as scale.

CCS CONCEPTS

• **Security and privacy** → *Multi-factor authentication*; **Social engineering attacks**; *Economics of security and privacy*.

KEYWORDS

user profiling; impersonation attacks; impersonation-as-a-service; threat modeling

ACM Reference Format:
Michele Campobasso and Luca Allodi. 2020. Impersonation-as-a-Service: Characterizing the Emerging Criminal Infrastructure for User Impersonation at Scale. In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security (CCS '20), November 9–13, 2020, Virtual Event, USA.* ACM, New York, NY, USA, 16 pages. https://doi.org/10.1145/3372297.3417892

1 INTRODUCTION

In recent years there has been a surge in criminal infrastructures supporting cyberattacks and cybercrime activities at large [2, 10, 20]. For example, *exploitation-as-a-service* and *pay-per-install* provide a set of attack technologies generally aimed at infecting systems or controlling bots that are then employed to launch, for example, DDoS attacks, or subsequent malware and phishing campaigns (e.g., to harvest credit card numbers or steal credentials). An important problem in any venture, let alone a criminal one, is the ability to *systematically* monetize the effort that goes into it [22]. In criminal enterprises, monetization is not necessarily an easy feat: whereas re-selling or giving access to infected systems to fellow criminals alleviates the problem for whom generates the infection (e.g., the *bot herder* [5, 26]), the problem of assigning a price to each bot remains [3]. Whereas the dynamics of demand and offer in the underground are likely to play a role in this setting (and remain an important open question to investigate in this domain), another key factor in determining the value of an infected system is the information it manages and/or processes; for example, access to the email account(s) of an Internet user may have a different value, to attackers, than access to a user profile with a server-stored credit card number (e.g., an e-commerce website). On the other hand, it is not yet clear how (and if) attackers can *systematically* employ those credentials to impersonate Internet users at large, particularly in the presence of multi-factor authentication systems whereby a username and password alone are not sufficient to gain access to an Internet account.

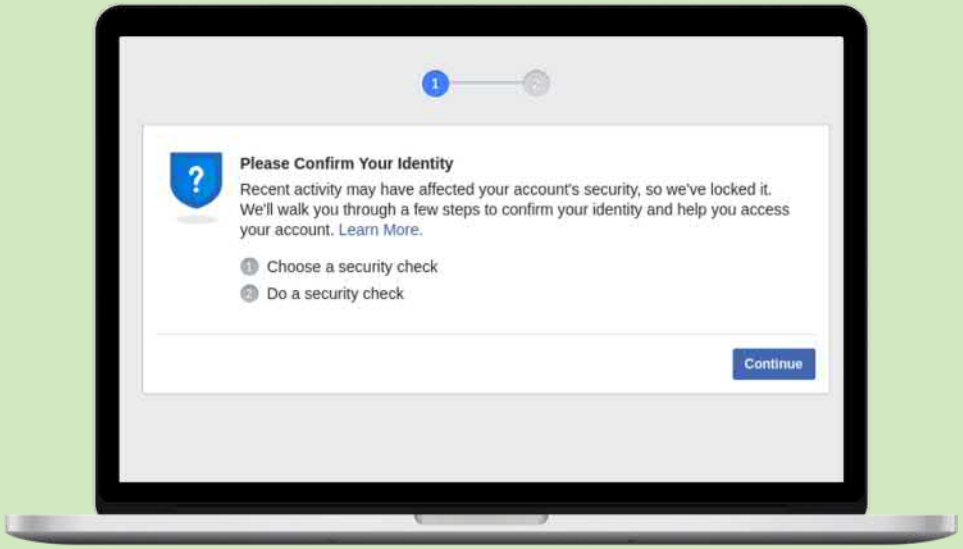
Credential theft and re-selling in underground communities have been studied multiple times in the literature; for example, recent studies provide an in-depth view of what happens to credentials after they have been stolen [35], and their employment for final attacks [40]. Similarly, several studies investigate the attack vectors that allow attackers to obtain the credentials in the first place, ranging from (targeted) phishing and phishing kits, to malware infections at scale [8, 9, 35]. On the other hand, a systematic employment of the stolen credentials remains out of reach for most attackers: credentials stolen from the underground may be accessed by multiple criminals, effectively destroying their value for later accesses [22]; similarly, the effort required to monetize access to stolen or hijacked user accounts does not scale well with the number of available accounts [22, 23]. In particular, protection systems such as multi-factor and risk-based authentication systems severely limit the capabilities of attackers to effectively employ stolen credentials, requiring the employment to more sophisticated attack vectors than a simple credentials dump [40]. Risk-based authentication systems receive user authentication requests and are responsible to decide whether additional multi-factor authentication is required for that session, or if the provided (valid) password suffices to grant access to the user requesting it. The idea behind risk-based authentication is that, by 'measuring' certain characteristics of the user environment (i.e., its fingerprint [1]), the authenticating system can build a 'risk profile' associated to that request as a function of the distance

CCS '20, November 9–13, 2020, Virtual Event, USA

ACM ISBN 978-1-4503-7089-9/20/11…$15.00
https://doi.org/10.1145/3372297.3417892

Hochschule
Bonn-Rhein-Sieg
University of Applied Sciences

/DAS>
Data and Application Security Group



What's in Score for Website Users:

A Data-driven Long-term Study on Risk-based Authentication Characteristics

Stephan Wiefling*#, Markus Dürmuth#, Luigi Lo Iacono*

H-BRS University of Applied Sciences (*)

Ruhr University Bochum (#)

Data Set

August ‘18 - June ‘20

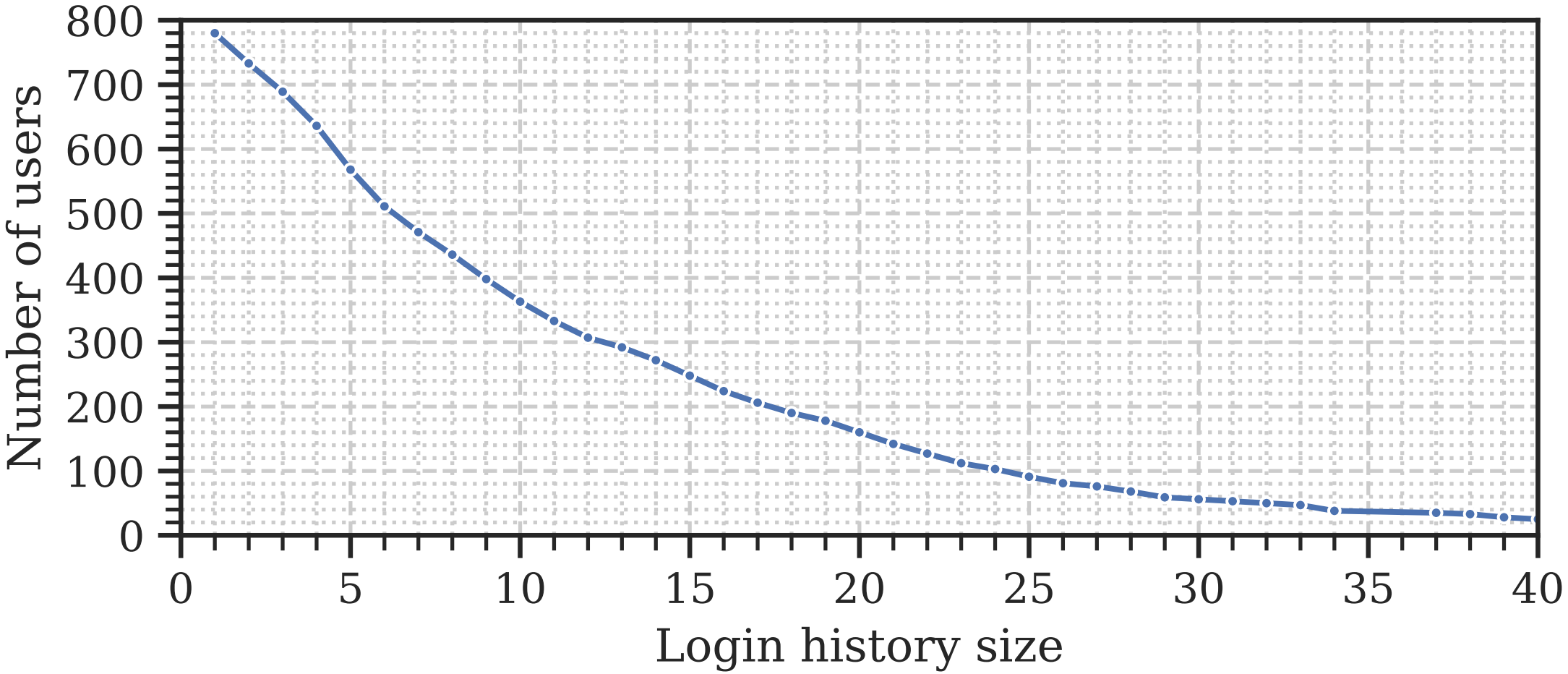
780 Users

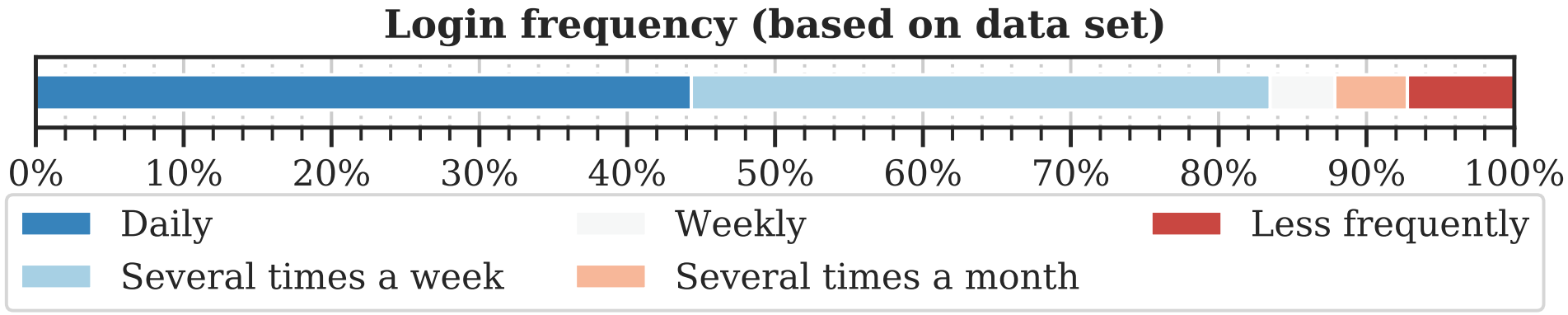
≈250 Features

New Feature!

≈9500 Login Sessions

- 81.1% desktop
- 18.9% mobile


Number of users
800
700
600
500
400
300
200
100
0
0
5
10
15
20
25
30
35
40
Login history size

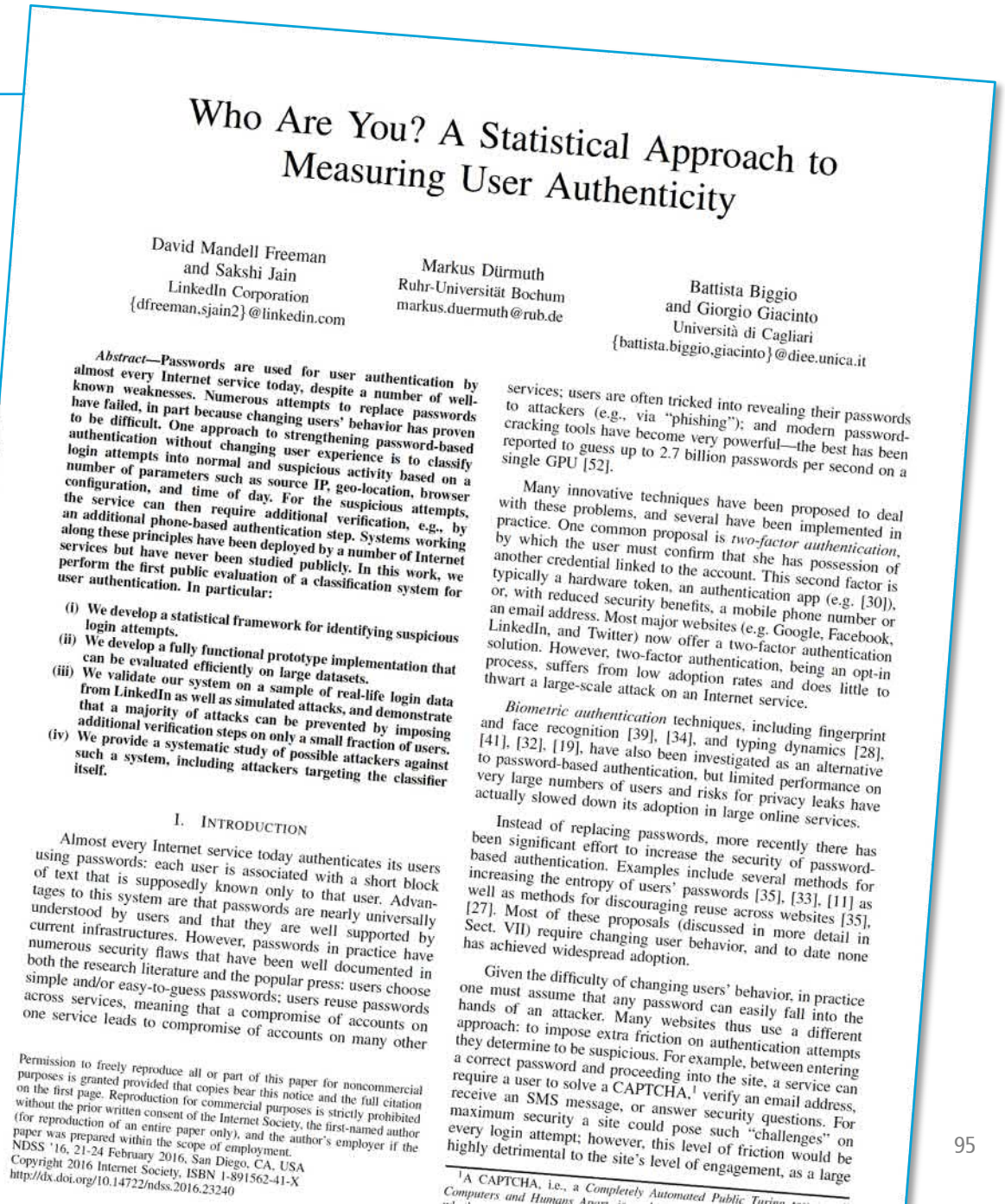

Login frequency (based on data set)
0%
10%
20%
30%
40%
50%
60%
70%
80%
90%
100%
Daily
Several times a week
Weekly
Several times a month
Less frequently

RBA Model

Extended Model (EXTEND)*

- Comparable to models apparently used by Google, Amazon, and LinkedIn

* Based on Freeman et al.: Who Are You? A Statistical Approach to Measuring User Authenticity. NDSS (2016).



Who Are You? A Statistical Approach to Measuring User Authenticity

David Mandell Freeman and Sakshi Jain
LinkedIn Corporation
{dfreeman,sjain2}@linkedin.com

Markus Dürmuth
Ruhr-Universität Bochum
markus.duermuth@rub.de

Battista Biggio and Giorgio Giacinto
Università di Cagliari
{battista.biggio,giacinto}@diee.unica.it

Abstract—**Passwords are used for user authentication by almost every Internet service today, despite a number of well-known weaknesses. Numerous attempts to replace passwords have failed, in part because changing users' behavior has proven to be difficult. One approach to strengthening password-based authentication without changing user experience is to classify login attempts into normal and suspicious activity based on a number of parameters such as source IP, geo-location, browser configuration, and time of day. For the suspicious attempts, the service can then require additional verification, e.g., by an additional phone-based authentication step. Systems working along these principles have been deployed by a number of Internet services but have never been studied publicly. In this work, we perform the first public evaluation of a classification system for user authentication. In particular:**

(i) We develop a statistical framework for identifying suspicious login attempts.
(ii) We develop a fully functional prototype implementation that can be evaluated efficiently on large datasets.
(iii) We validate our system on a sample of real-life login data from LinkedIn as well as simulated attacks, and demonstrate that a majority of attacks can be prevented by imposing additional verification steps on only a small fraction of users.
(iv) We provide a systematic study of possible attackers against such a system, including attackers targeting the classifier itself.

I. Introduction

Almost every Internet service today authenticates its users using passwords: each user is associated with a short block of text that is supposedly known only to that user. Advantages to this system are that passwords are nearly universally understood by users and that they are well supported by current infrastructures. However, passwords in practice have numerous security flaws that have been well documented in both the research literature and the popular press: users choose simple and/or easy-to-guess passwords; users reuse passwords across services, meaning that a compromise of accounts on one service leads to compromise of accounts on many other services; users are often tricked into revealing their passwords to attackers (e.g., via "phishing"); and modern password-cracking tools have become very powerful—the best has been reported to guess up to 2.7 billion passwords per second on a single GPU [52].

Many innovative techniques have been proposed to deal with these problems, and several have been implemented in practice. One common proposal is *two-factor authentication*, by which the user must confirm that she has possession of another credential linked to the account. This second factor is typically a hardware token, an authentication app (e.g. [30]), or, with reduced security benefits, a mobile phone number or an email address. Most major websites (e.g. Google, Facebook, LinkedIn, and Twitter) now offer a two-factor authentication solution. However, two-factor authentication, being an opt-in process, suffers from low adoption rates and does little to thwart a large-scale attack on an Internet service.

Biometric authentication techniques, including fingerprint and face recognition [39], [34], and typing dynamics [28], [41], [32], [19], have also been investigated as an alternative to password-based authentication, but limited performance on very large numbers of users and risks for privacy leaks have actually slowed down its adoption in large online services.

Instead of replacing passwords, more recently there has been significant effort to increase the security of password-based authentication. Examples include several methods for increasing the entropy of users' passwords [35], [33], [11] as well as methods for discouraging reuse across websites [35], [27]. Most of these proposals (discussed in more detail in Sect. VII) require changing user behavior, and to date none has achieved widespread adoption.

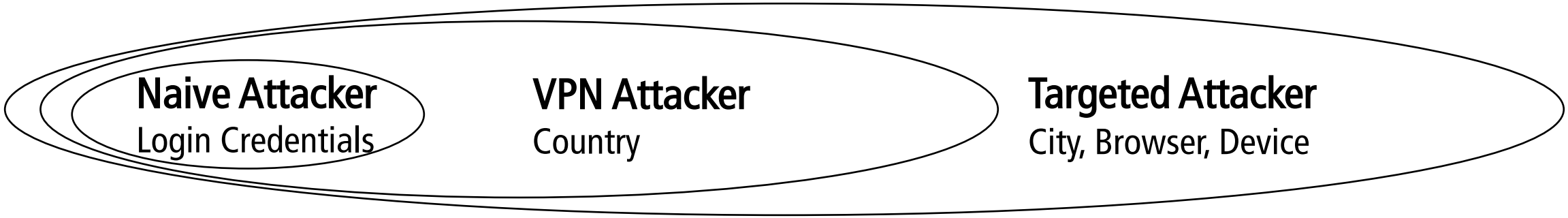
Given the difficulty of changing users' behavior, in practice one must assume that any password can easily fall into the hands of an attacker. Many websites thus use a different approach: to impose extra friction on authentication attempts they determine to be suspicious. For example, between entering a correct password and proceeding into the site, a service can require a user to solve a CAPTCHA,[1] verify an email address, receive an SMS message, or answer security questions. For maximum security a site could pose such "challenges" on every login attempt; however, this level of friction would be highly detrimental to the site's level of engagement, as a large

NDSS '16, 21-24 February 2016, San Diego, CA, USA

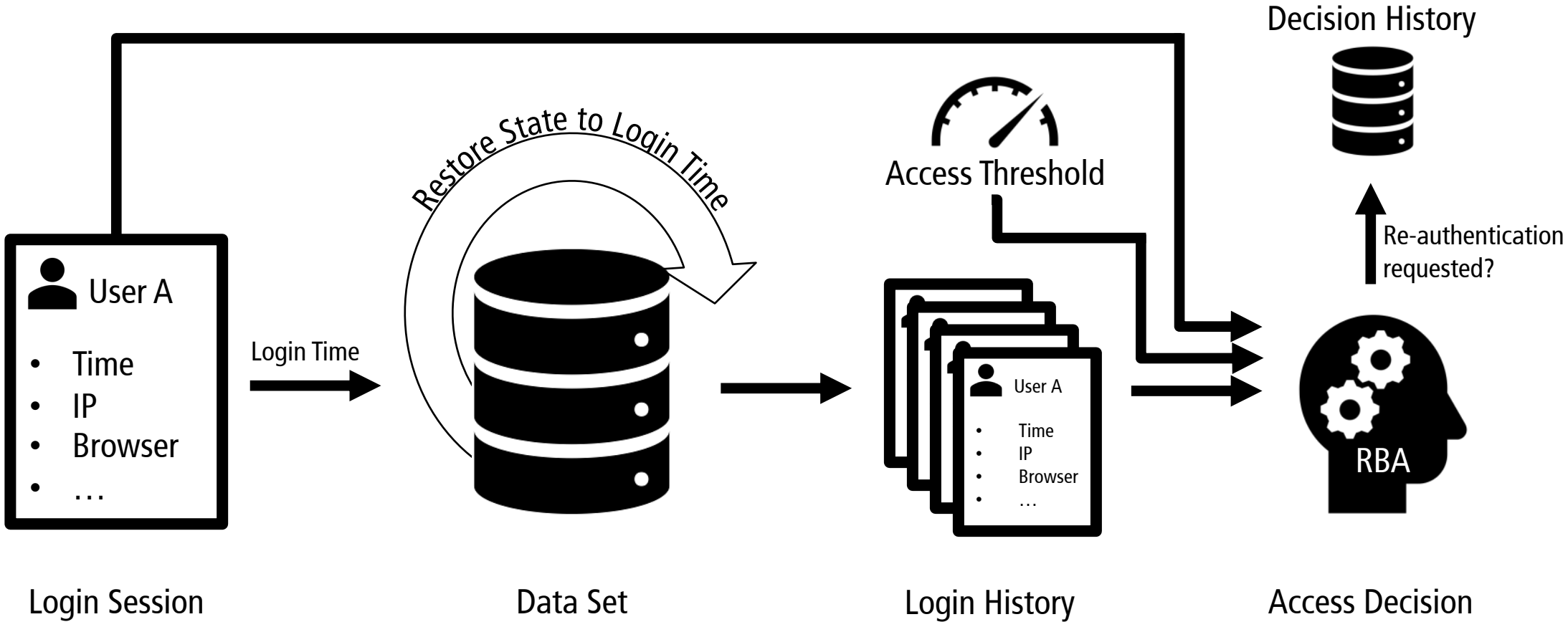
http://dx.doi.org/10.14722/ndss.2016.23240

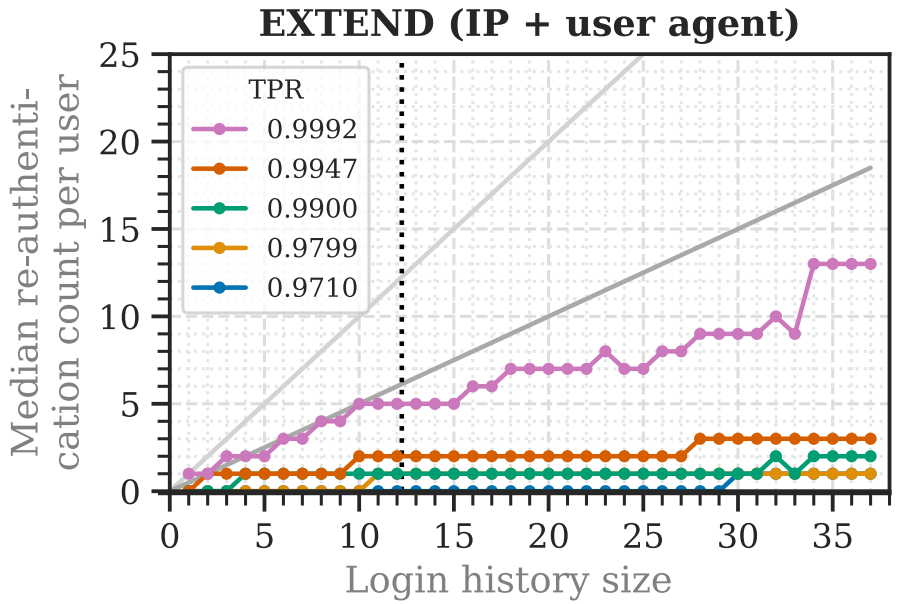
[1] A CAPTCHA, i.e., a *Completely Automated Public Turing* ... *Computers and Humans Apart*

Attacker Models


Naive Attacker
Login Credentials
VPN Attacker
Country
Targeted Attacker
City, Browser, Device

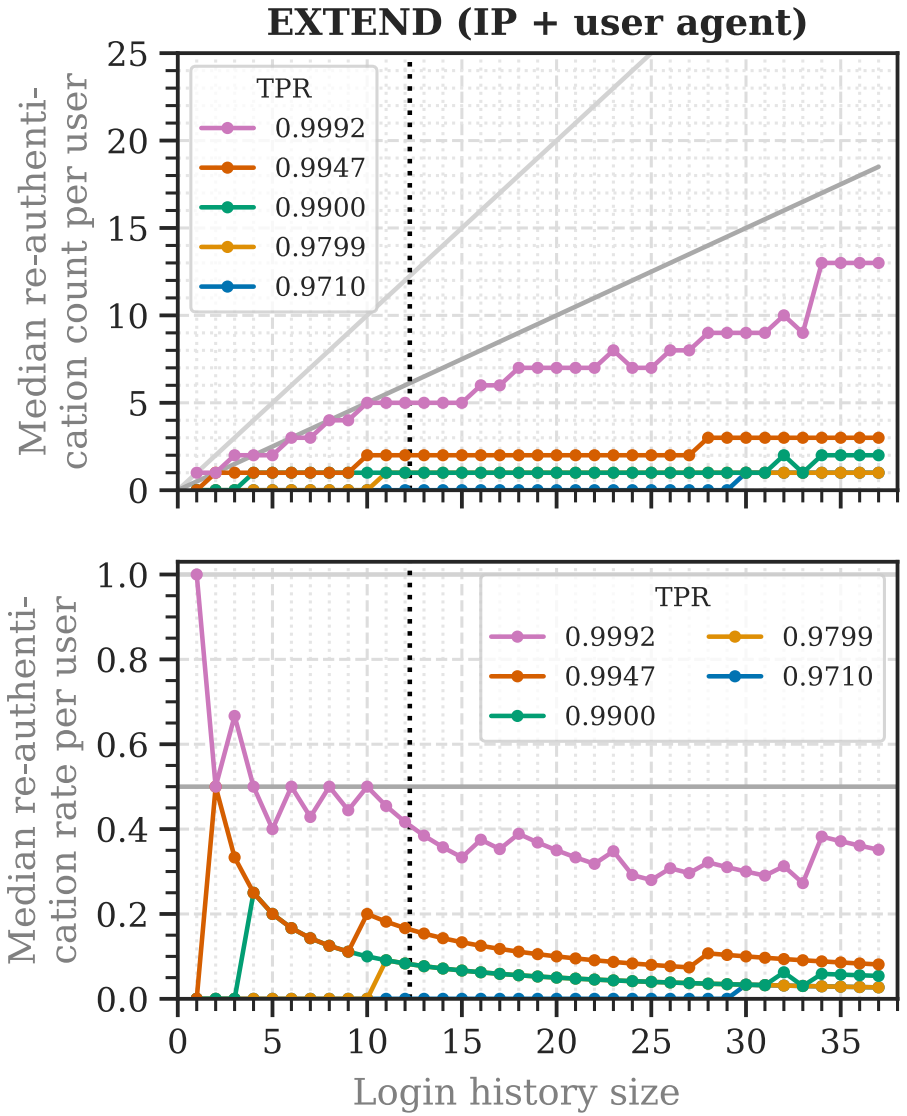
How often will RBA request re-authentication in practice?


Decision History
Restore State to Login Time
Access Threshold
Re-authentication requested?
User A
• Time
• IP
• Browser
• …
Login Time
User A
• Time
• IP
• Browser
• …
RBA
Login Session
Data Set
Login History
Access Decision



Results (Targeted Attacker)

- RBA asks less for re-authentication with increasing number of logins



Results (Targeted Attacker)

- Only one entry required for stable and reliable RBA setup in most cases

Analyzing RBA Features

Single Features

- High reliability
- Good RBA performance on their own

Feature	JavaScript not required	Risk Score Ratio	Unique values	Median logins until re-authentication
IP address	●	1.20	●●●●●	2.00

Single Add-on Features

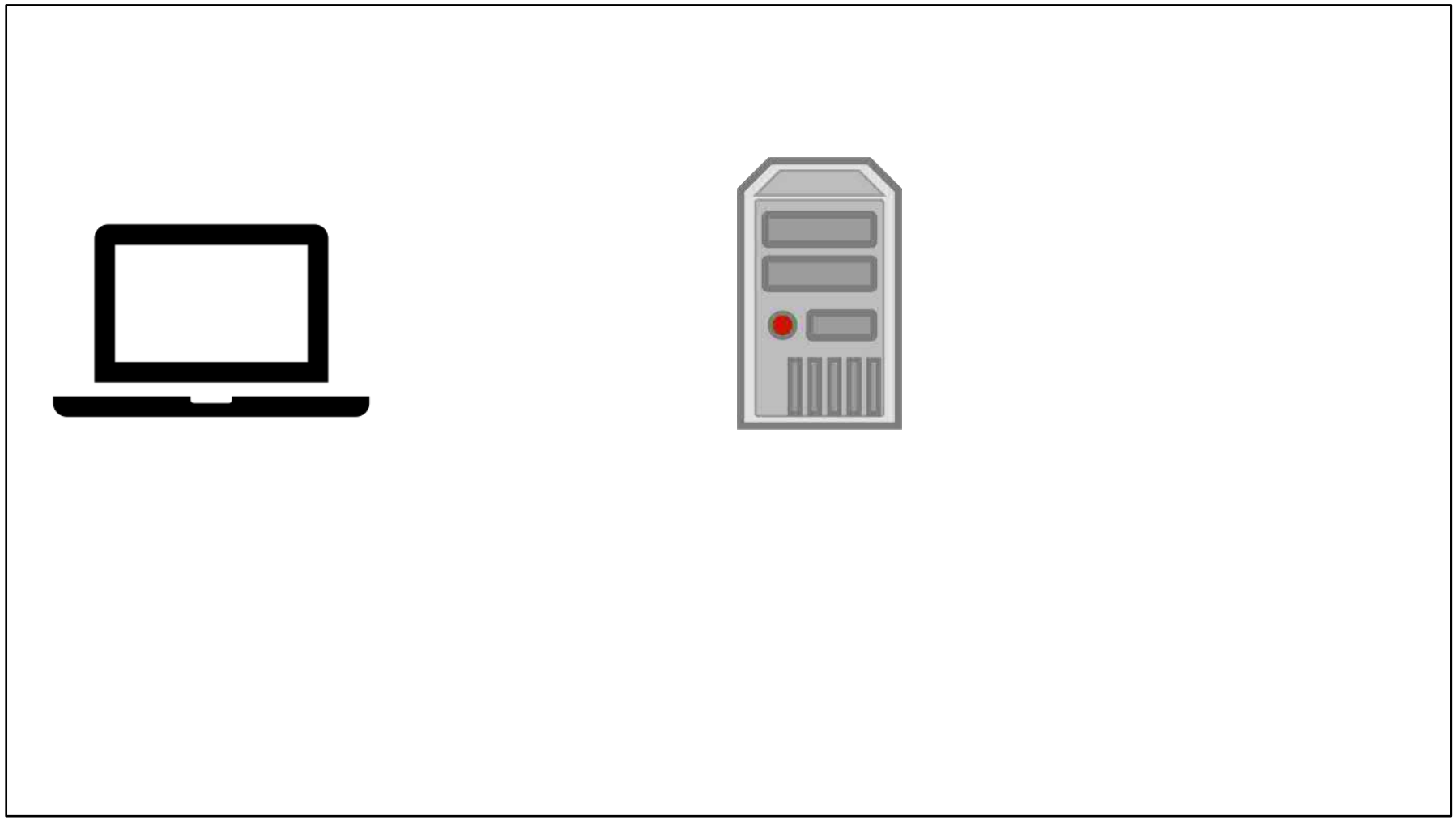
- High reliability
- Only good RBA performance in combination with single feature

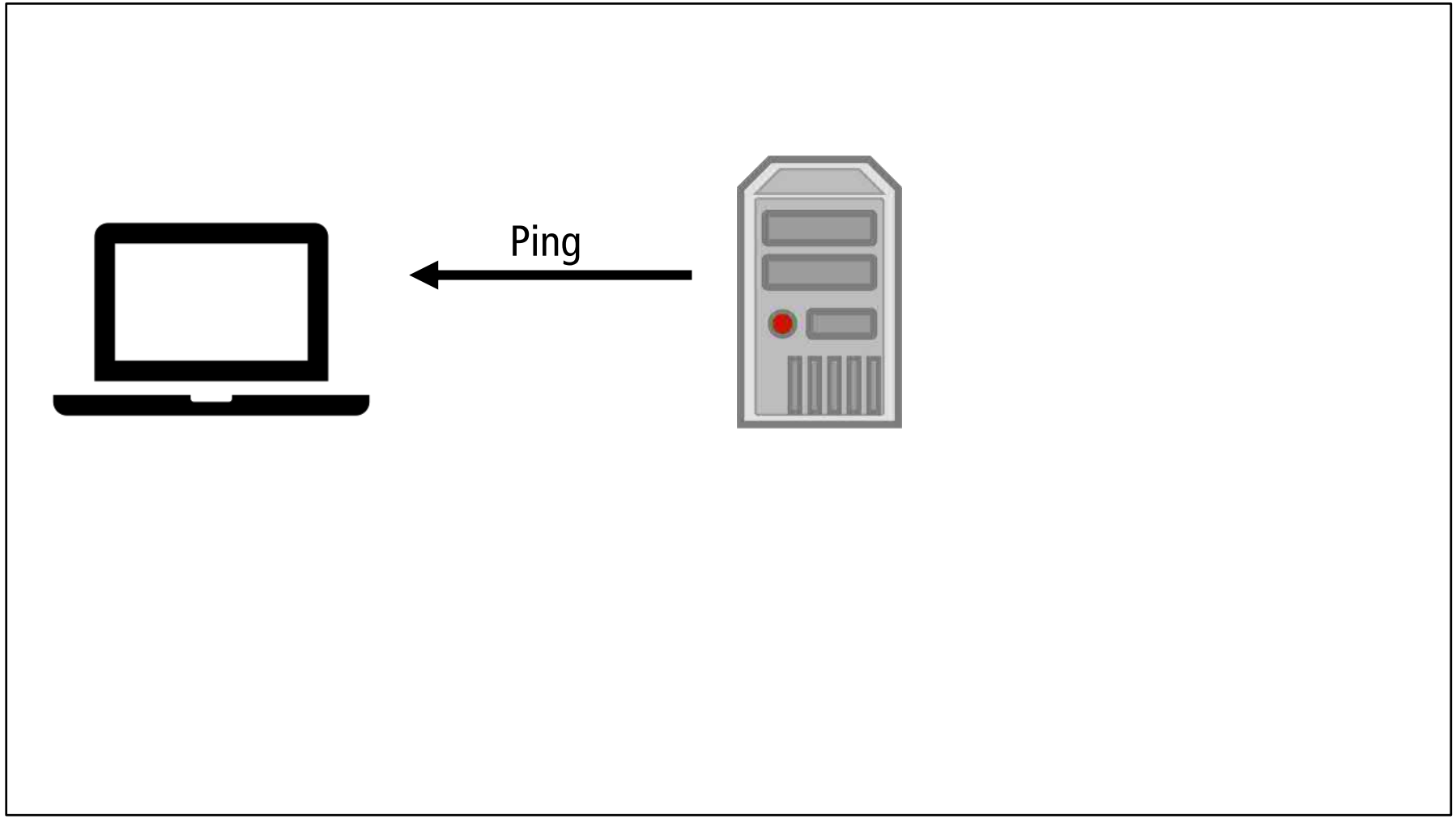
Feature	JavaScript not required	Risk Score Ratio	Unique values	Median logins until re-authentication
RTT-10MS	○	1.75	●●○○○	1.50
RTT-5MS	○	1.37	●●○○○	1.71
ASN (IP)	●	0.91	●●○○○	3.00
RTT-MS	○	0.56	●●●●○	2.00
Hour	●	0.23	●○○○○	4.00
Region (IP)	●	0.15	●○○○○	1.71
Weekday and hour	●	0.15	●●●○○	4.00

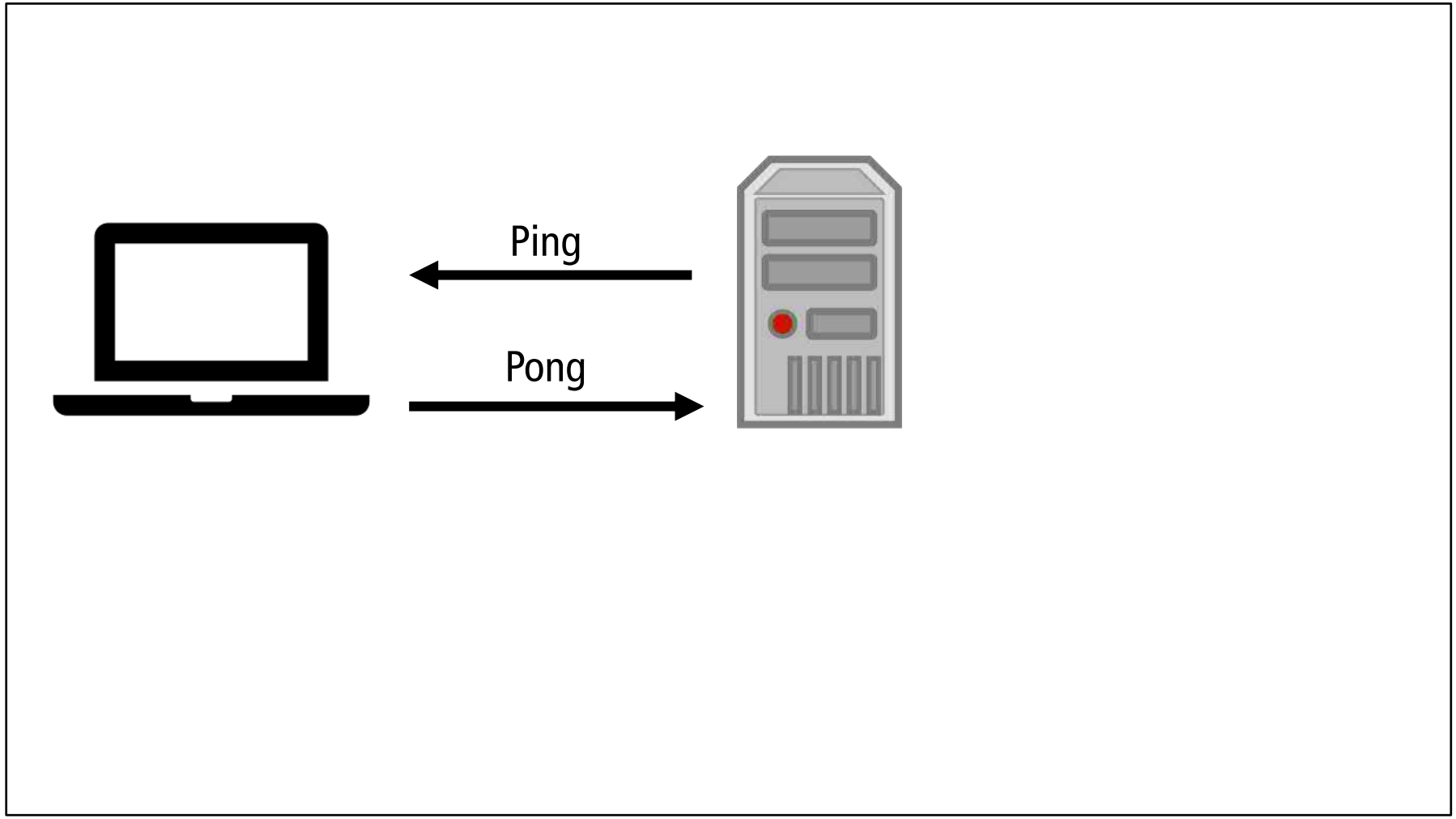
Feature	JavaScript not required	Risk Score Ratio	Unique values	Median logins until re-authentication
RTT-10MS	○	1.75	●●○○○	1.50
RTT-5MS	○	1.37	●●○○○	1.71
ASN (IP)	●	0.91	●●○○○	3.00
RTT-MS	○	0.56	●●●●○	2.00
Hour	●	0.23	●○○○○	4.00
Region (IP)	●	0.15	●○○○○	1.71
Weekday and hour	●	0.15	●●●○○	4.00

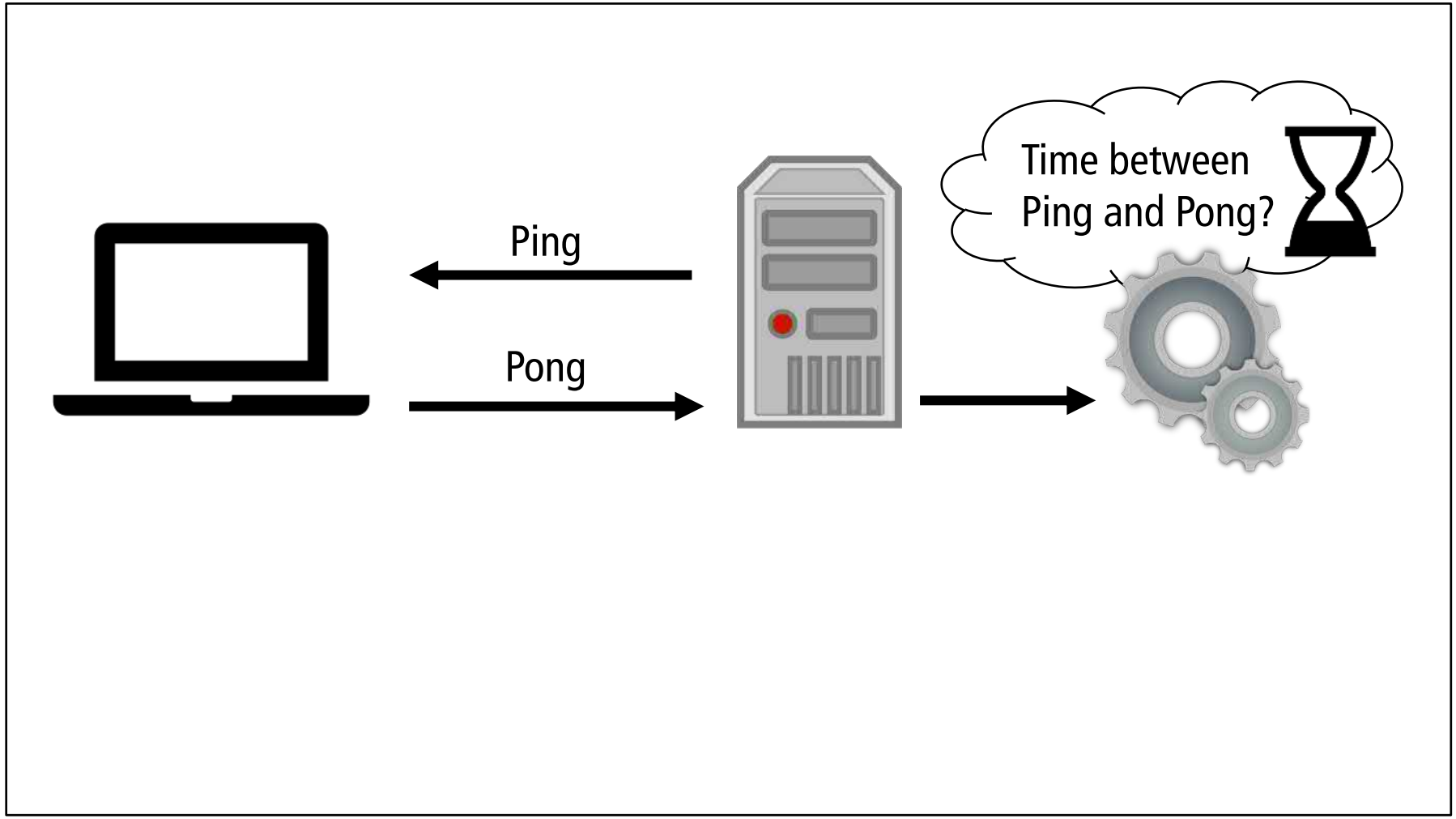
Round Trip Time

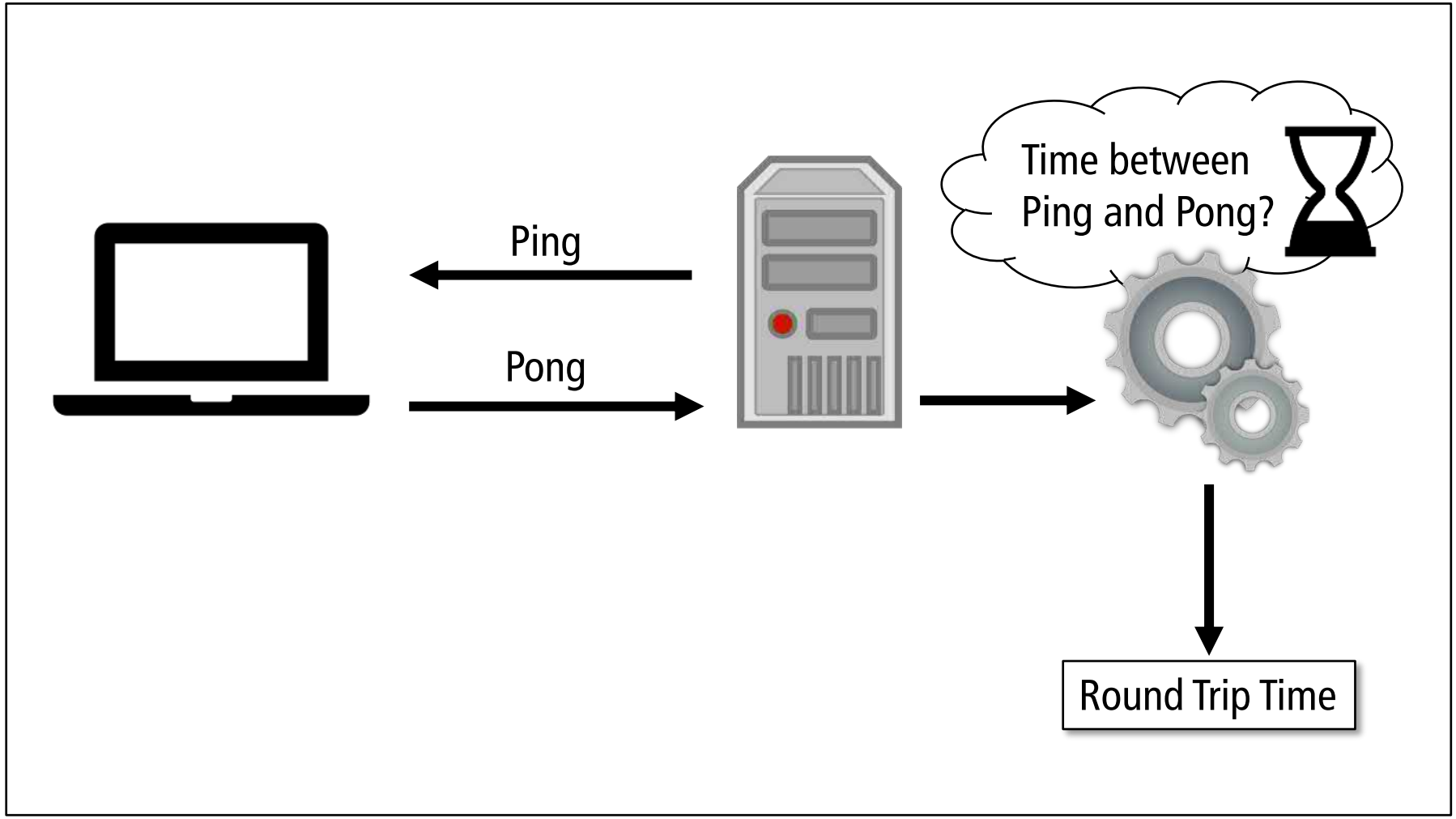
- Based on WebSockets

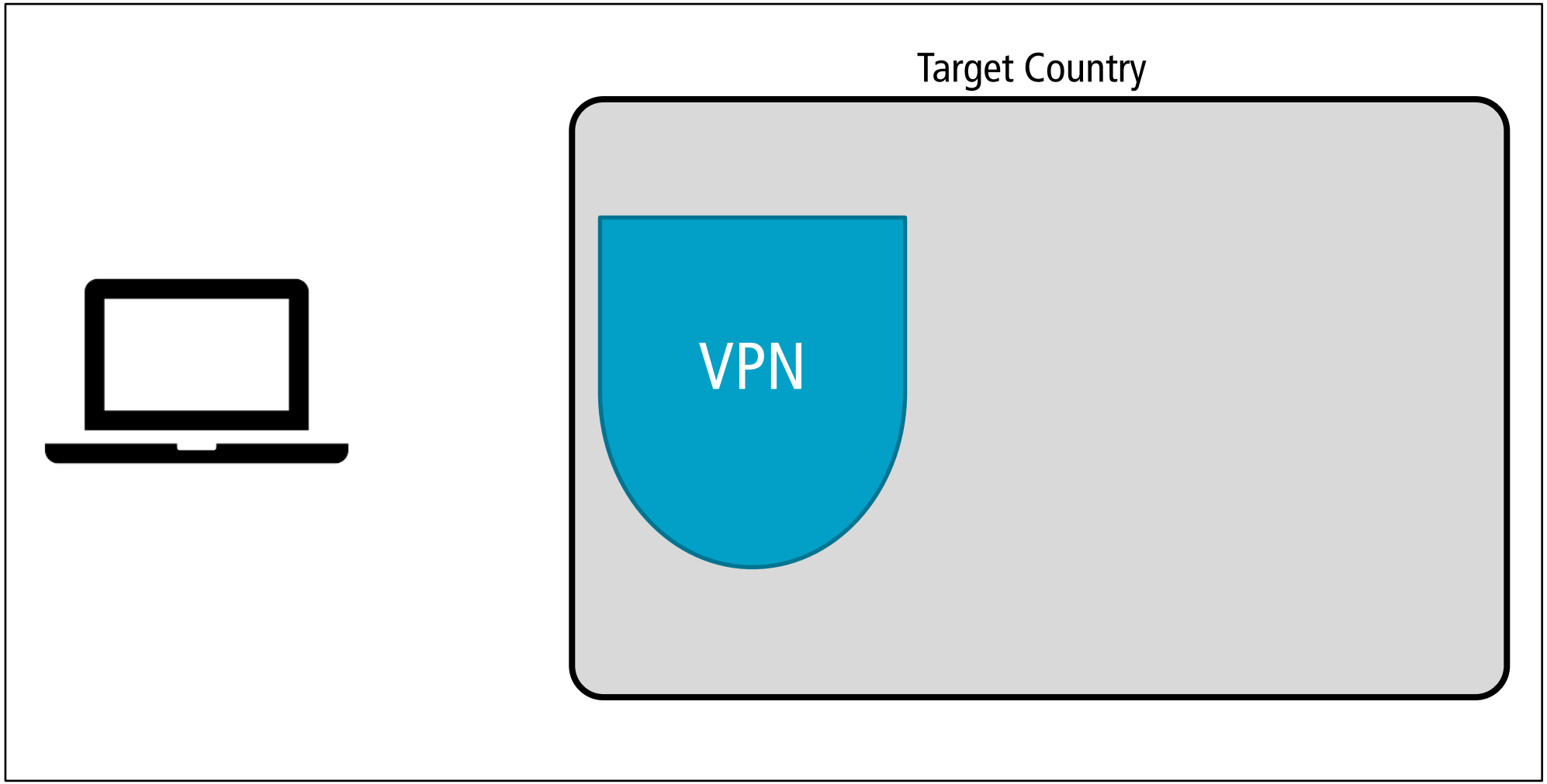


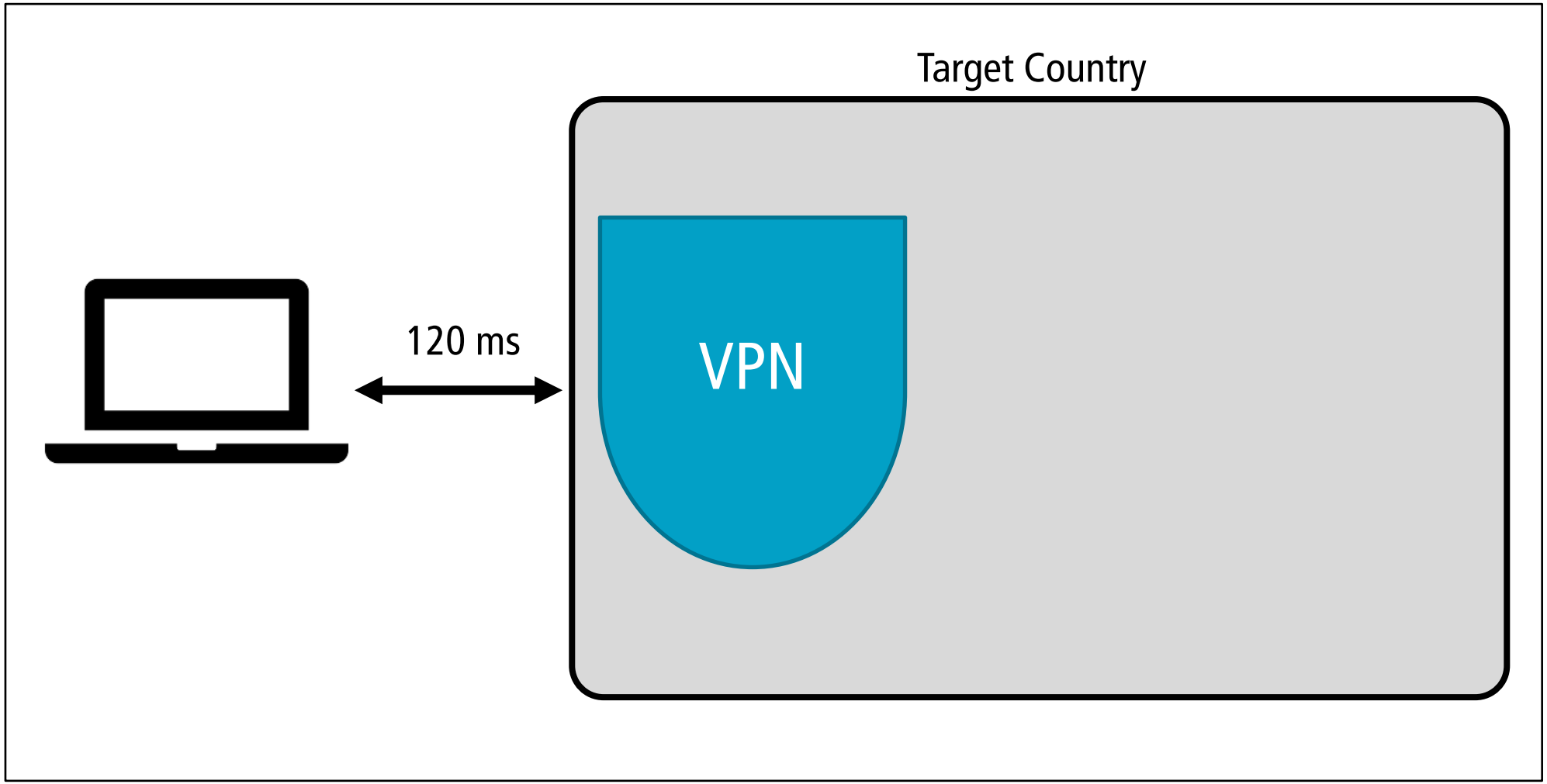

Ping

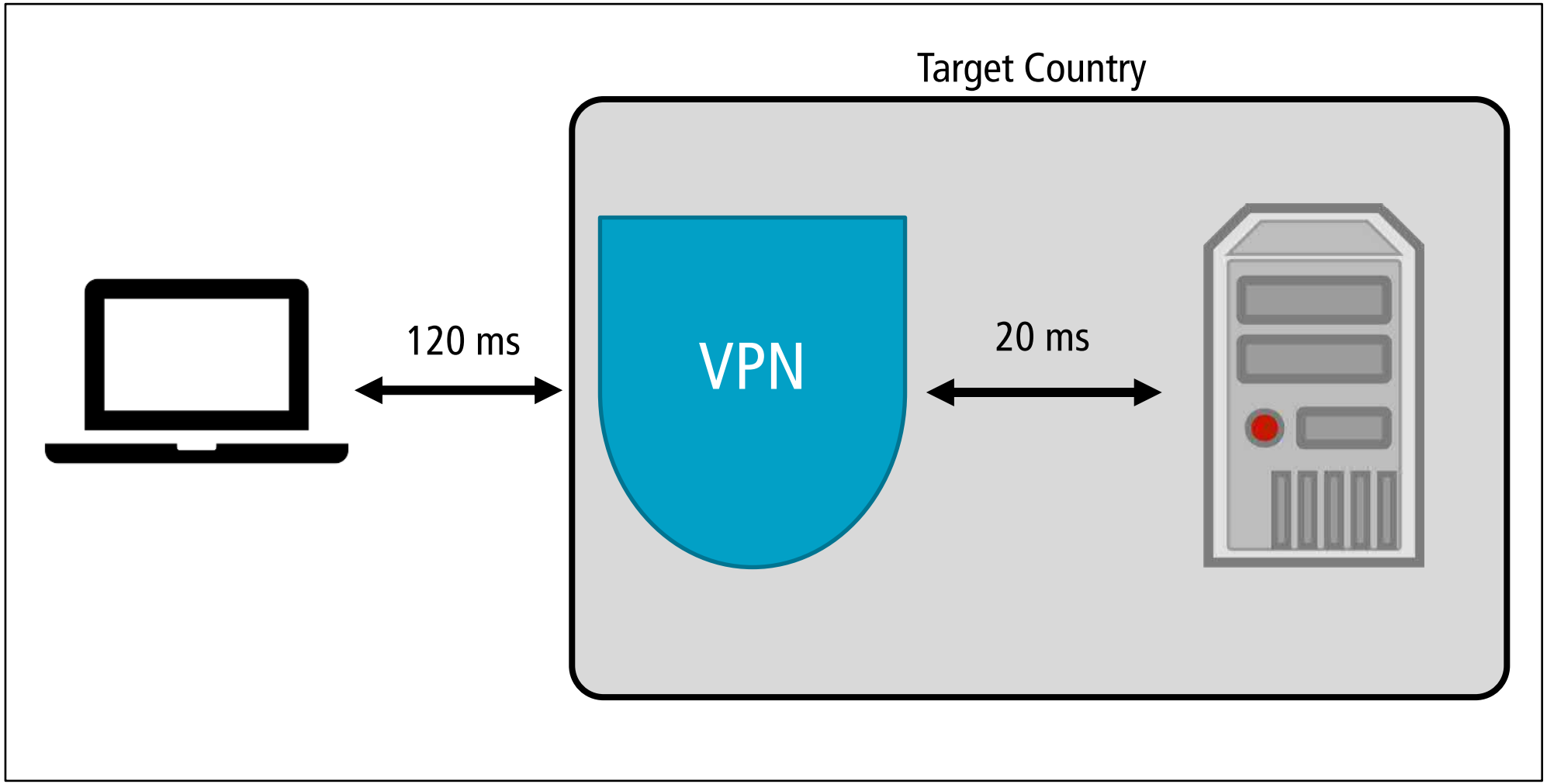

Ping
Pong

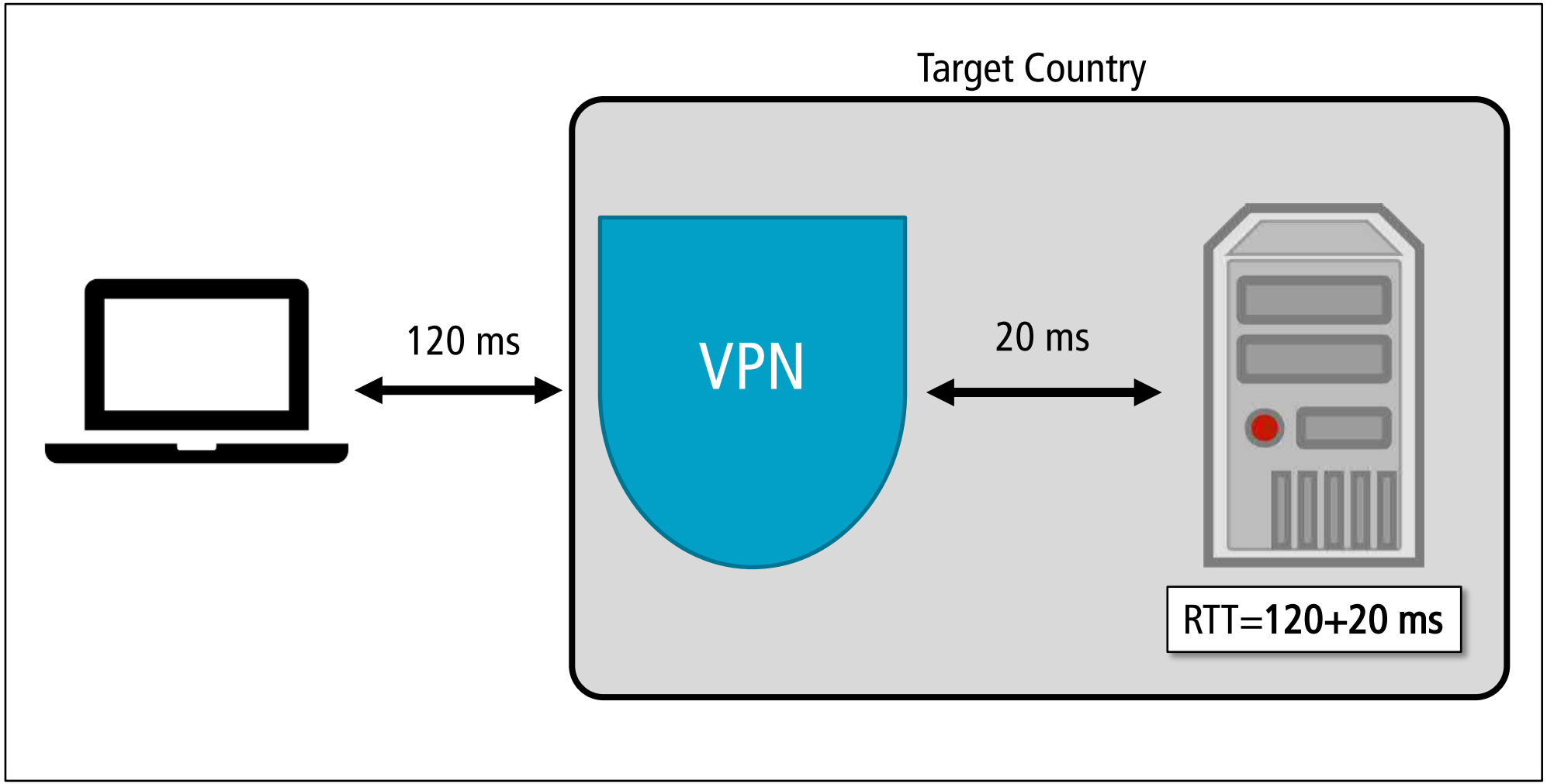

Ping
Pong
Time between
Ping and Pong?

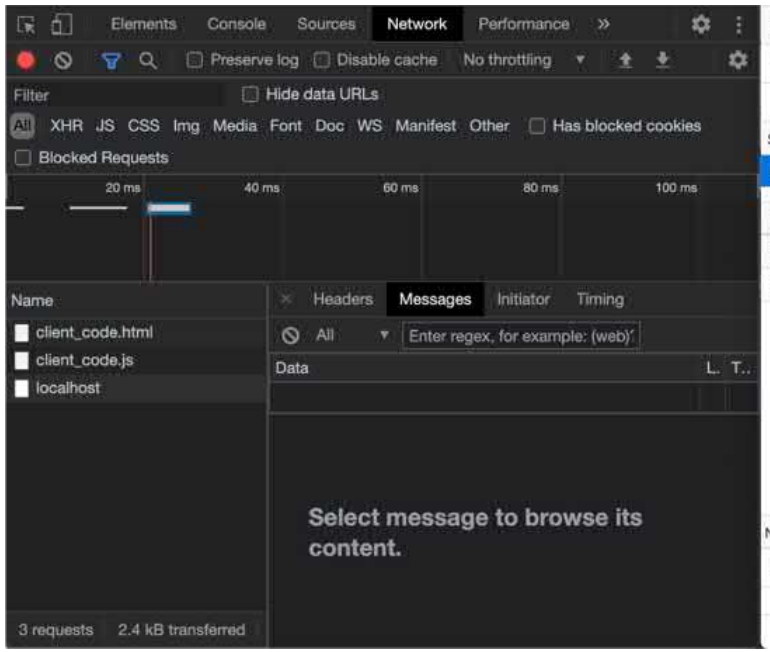

Ping
Pong
Time between
Ping and Pong?
Round Trip Time


Target Country
VPN

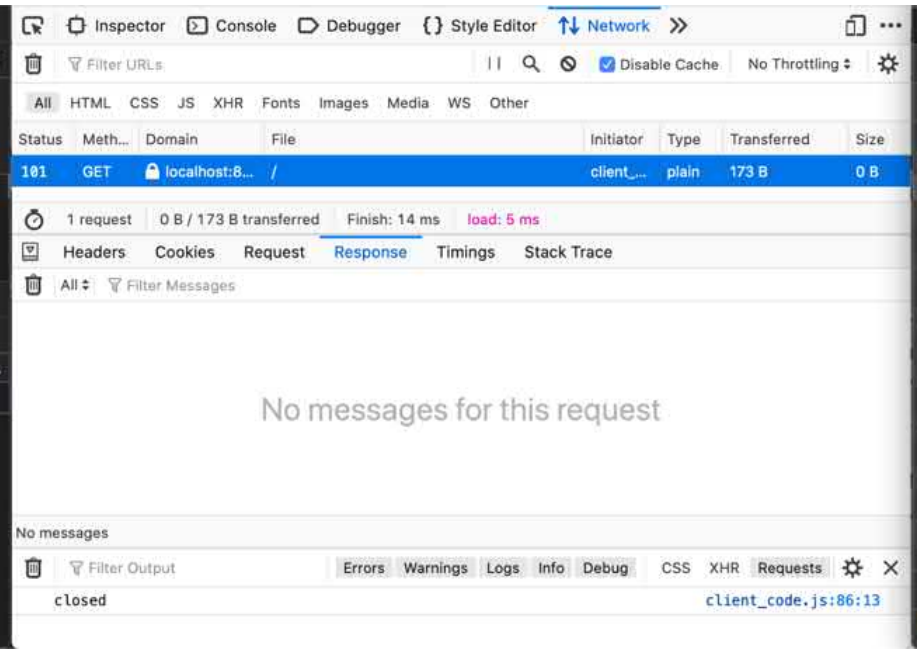

Target Country
120 ms
VPN


Target Country
120 ms
VPN
20 ms

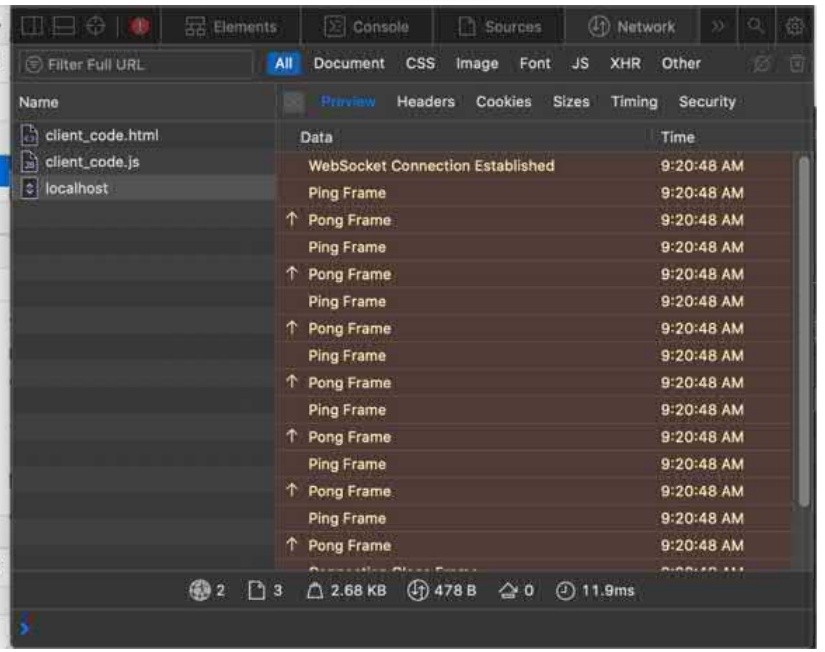

Target Country
120 ms
VPN
20 ms
RTT=120+20 ms



Hidden



Hidden



Visible

Conclusion

- Only few features are useful for RBA

- EXTEND model can achieve low re-authentication rates when blocking >99.45% targeted attackers

- Needs to be optimized for use-case scenario

Thank you

riskbasedauthentication.org
das.h-brs.de

luigi.lo_iacono@h-brs.de