Evaluating Login Challenges as a Defense Against Account Takeover

Periwinkle Doerfler[†] Maija Marincenko[◇] Juri Ranieri[◇] Yu Jiang[◇]
Angelika Moscicki[◇] Damon McCoy[†] Kurt Thomas[◇]

New York University[†] Google[◇]

ABSTRACT

In this paper, we study the efficacy of login challenges at preventing account takeover, as well as evaluate the amount of friction these challenges create for normal users. These secondary authentication factors—presently deployed at Google, Microsoft, and other major identity providers as part of risk-aware authentication—trigger in response to a suspicious login or account recovery attempt. Using Google as a case study, we evaluate the effectiveness of fourteen device-based, delegation-based, knowledge-based, and resource-based challenges at preventing over 350,000 real-world hijacking attempts stemming from automated bots, phishers, and targeted attackers. We show that knowledge-based challenges prevent as few as 10% of hijacking attempts rooted in phishing and 73% of automated hijacking attempts. Device-based challenges provide the best protection, blocking over 94% of hijacking attempts rooted in phishing and 100% of automated hijacking attempts. We evaluate the usability limitations of each challenge based on a sample of 1.2M legitimate users. Our results illustrate that login challenges act as an important barrier to hijacking, but that friction in the process leads to 52% of legitimate users failing to sign-in—though 97% of users eventually access their account in a short period.

KEYWORDS

account takeover; account recovery; two-factor authentication

ACM Reference Format:
Periwinkle Doerfler, Maija Marincenko, Juri Ranieri, Yu Jiang, Angelika Moscicki, Damon McCoy, Kurt Thomas. 2019. Evaluating Login Challenges as a Defense Against Account Takeover. In *Proceedings of the 2019 World Wide Web Conference (WWW '19), May 13–17, 2019, San Francisco, CA, USA.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3308558.3313481

1 INTRODUCTION

In response to the threat of wide-scale password theft and account takeover [22], identity providers have re-framed password-only authentication from a binary task of validating password correctness into *risk-aware authentication* [19]. This approach incorporates multiple passive authentication signals with layered proofs of identity. For example, Freeman et al. developed a statistical model of a user's geolocation, browser configuration, and access patterns to detect aberrant login attempts [10]. Google and Microsoft actively deploy similar systems that judge IP address reputation, device reputation, and automation patterns [18, 23]. Of course, with any risk analysis scheme there exists a chance for false positives as users travel and acquire new devices, and networks churn due to DHCP lease expiration. As such, outright blocking any suspicious sign-in would lock a significant volume of legitimate users out of their accounts.

In order to differentiate hijackers from legitimate account holders in edge cases, risk-aware authentication triggers a *challenge* to act as a second proof of identity. Challenges can take many forms: proof of access to a registered device, proof of access to a backup account, knowledge of a shared secret (e.g., a security question), or merely proof of access to a scarce resource (e.g., a CAPTCHA). This layered authentication approach is distinct from two-factor authentication as challenges appear dynamically in response to a perceived threat, thus reducing overall friction on every user sign-in. Additionally, challenges scale to every user by adapting to the security posture of each account, taking advantage of recovery accounts or trusted devices when available. While ideal from a deployment standpoint, it remains unclear how well these varied challenges protect against hijacking or what friction they create for legitimate account holders.

In this paper, we evaluate the security and usability trade-offs of fourteen challenges based on their real-world performance. We begin by building a ground truth corpus of over 350,000 hijacking attempts from increasingly sophisticated actors including automated bots, phishers, and targeted attackers. Then, using login traces from Google, we examine whether risk-aware authentication triggered a challenge for these hijacking attempts, what challenge the algorithm selected, and ultimately whether the hijacker was able to access the account. Additionally, we isolate 1.2 million challenges solved by legitimate users to measure how often challenges temporarily locked out account holders. Critically, our evaluation approach is independent of Google's risk analysis techniques and generalizes to any identity provider.

Our analysis reveals that even weak, knowledge-based challenges can offer hijacking protections against automation to billions of users without requiring any enrollment. That said, the security posture of users plays an important role in protecting against more sophisticated attacks. Users that established a trusted device or a delegated recovery account received up to 10 times better protections against phishing, and to a lesser extent, targeted attacks. We argue that risk-aware authentication and challenges can help extend the viability of passwords in the interim as the community

WWW '19, May 13–17, 2019, San Francisco, CA, USA

ACM ISBN 978-1-4503-6674-8/19/05.
https://doi.org/10.1145/3308558.3313481

considers alternative forms of authentication. For at-risk users looking to mitigate the risk of remote hijacking entirely, we emphasize that security keys offer the best immediate solution.

We summarize our key findings as follows:

- Knowledge-based challenges, such as recalling a secondary email address, prevented over 73% of automated hijacking attempts but only 10% of attacks rooted in phishing.
- On-device prompts, like those shown during two-factor authentication, provide the strongest protection. They blocked 99% of attacks rooted in phishing and 90% of targeted attacks.
- SMS-based challenges provided weaker protections, preventing only 96% of attacks rooted in phishing and 76% of targeted attacks.
- We show that risk-aware authentication in aggregate prevented over 99.99% of automated hijacking attempts and over 92% of attacks rooted in phishing.
- Despite low per-challenge pass rates by legitimate users (13–64% depending on the challenge), over 97% of users gained access to their account within a short period of seeing a challenge. This figure is comparable to password-only authentication.

2 LOGIN & RECOVERY CHALLENGES

Before diving into our study, we outline the fourteen challenges currently deployed at Google that can be triggered at login or during account recovery. We categorize these challenges into four classes as shown in Table 1. While our study focuses on empirical measures of performance, we also frame each challenge's trade-offs between memorability, ease of use, and susceptibility to attacks according to the criteria set down by Bonneau et al. for evaluating web authentication schemes [3]. We extend this taxonomy to include five new criteria. *Frictionless setup*—the challenge does not require a pre-established secret with an identity provider. *No additional network access*—there is no need for a cellular or other network connection. *Resilient to third-party leaks*—the challenge does not involve data that might be revealed by a breach. *Not publicly searchable*—the challenge cannot be gleaned from public information. And lastly *Affinity to user*—the challenge establishes some relationship to a user, rather than just a hard to acquire resource. We discuss each class of challenge according to this criteria and how Google opts to elect one challenge over another during risk analysis.

2.1 Types of challenges

Device-based Challenges. Device-based challenges leverage the ubiquity of mobile devices to establish a second authentication factor for users or, less frequently, access to a hardware security token. For users with mobile devices, identity providers can leverage one-time pads (e.g., a 6-8 digit code that users re-type) generated by an app like Google Authenticator [16] or Duo [7] or via the phone's operating system [11]. Both can operate offline by using a synchronized key between a server and device. Additional delivery mechanisms include on-device prompts where a user clicks "Yes" to confirm a sign-in attempt [7, 13] or SMS where a user re-types a numeric code texted to their device. Both of these require additional network access.

As detailed in Table 1, the primary usability concern of device-based challenges is both access to a mobile device or phone number and the willingness of users to proactively associate these with their account. Sharing a phone number incurs at least some friction in terms of setup, while registering a device or authenticator app requires even more steps. Additionally, while the time-sensitive nature of device challenges obviates some classes of attacks, they are nevertheless vulnerable to man-in-the-middle attacks such as inline phishing. This stems from the inability of apps that generate a code to observe the domain a victim is visiting, something U2F security keys solve. Theft of mobile devices is another risk as it both locks the victim out of their account and gives attackers access to OTP codes. Requiring a PIN or password to access the device partially mitigates this threat.

Delegation-based Challenges. Delegation-based challenges rely on a secondary identity provider to serve as an authority for establishing trust with a user. At Google, this involves sending a one-time code to a secondary email address. Other alternatives include Facebook's delegated recovery [9] where Facebook provides a break-glass, pre-registered secret token that users can employ in lieu of a password at the other identity provider. These schemes require memorizing a username and password for the delegated identity provider. As a single identity (and thus password) can span multiple other providers, we consider backup emails to be partially memorywise effortless. Compared to device-based challenges, these schemes are susceptible to stepping-stone attacks, phishing, or brute force guessing of the secondary account's password, where security is only as strong as the weakest link in the identity chain.

Knowledge-based Challenges. This class of challenges relies on establishing a shared secret with users either at registration time or as the result of actions on a service. Examples include answering security questions, or providing a recovery email or phone number (where no challenge code is sent). Alternatively, identity providers may probe information that a user implicitly provided, such as where the user commonly logs in from or the date they first created an account. Finally, while not presently deployed at Google, challenges may cover a history of the user's interactions, such as selecting the names or faces of contacts the user recently communicated with among decoy contacts [8].

Some knowledge-based schemes provide a frictionless account creation process, but then force users to recall potentially non-obvious interactions in the future. We consider re-usable secrets, such as an email or phone number, to be partially frictionless to establish. Knowledge-based challenges are also vulnerable to any form of observation, including third-party leaks or search indexing, and even fail to protect against throttled guessing when the solution space is biased towards common answers [2]. Nevertheless, these challenges are the only fallback mechanism for establishing some proof of identity when users refuse requests to establish device-based or delegation-based backups.

Resource-based Challenges. Resource challenges do not attempt to establish a proof of identity with users. Instead, they offer a hurdle to prevent automation. These challenges include confirming access to any phone number or (secondary) email address. As such, the primary goal is to restrict throttled guessing. However, attackers

Category	Challenge	Context	Memorywise Effortless	Nothing to Carry	No Additional Network Access	Frictionless Setup	Affinity to User	Resilient to Physical Observation	Resilient to Phishing	Resilient to Third-party Leaks	Not Publicly Searchable	Resilient to Physical Theft	Resilient to Throttled Guessing	Resilient to Unthrottled Guessing	Single Trusted Identity Provider
Device-based	App-based OTP[7, 16]	S	●		●		●	●	◐	●	●	◐	●		●
	Device Settings OTP[11]	L, R	●		●		●	●	◐	●	●	◐	●		●
	Device Prompt[7, 13]	S, L, R	●				●	●	◐	●	●	◐	●	●	●
	Mobile SMS OTP	S, L, R	●			◐	●	●	◐	●	●	◐	●		
	Security key	S	●		●		●	●	●	●	●		●	●	●
Delegation-based	OTP to backup email	L, R	◐	●	●	◐	●			●	●	●			
Knowledge-based	Known pre-registered email	L, R	◐	●	●	◐	●					●	●		●
	Security question	L, R		●	●		●				●	●			●
	Known pre-registered phone number	L, R	◐	●	●	◐	●					●	●		●
	Last login location	L		●	●	●	●			◐	◐	●	●		●
	Account creation date	R		●	●	●	●			●	●	●	●		●
	Printed backup code	S, R	●		●		●			●	●		●		●
Resource-based	OTP to any phone	L	●			●		●	◐	●	●	◐	●		
	OTP to any email address	R	●	●	●	●				●	●	●	●		

Table 1: Challenges presently deployed at Google, broken down by their usability and security concerns. Challenges may appear in multiple contexts: at login due to a suspicious sign-in attempt (L), during recovery when a user forgets their passwords (R), or as a second-factor for accounts with 2FA enabled (S). We indicate fulfillment of a property with a filled circle (●) and partial fulfillment with a half circle (◐). We refer readers to Bonneau et al. for a detailed description of each criteria [3].

can farm these challenges out to unsuspecting users or rely on compromised resources including phones or email accounts.

2.2 Challenge Selection in Practice

Challenge selection at Google is a function of both security posture and context. In the event of a suspicious sign-in attempt, Google's risk analysis engine selects the strongest challenge that an account's legitimate owner should ideally be able to solve. For accounts with an associated device or phone number, the risk engine exclusively allows device-based challenges[1]. Absent a device, the engine falls back to delegation-based challenges, then knowledge-based challenges, and ultimately additional resource challenges. As such, even though a hijacker may know a victim's phone number or backup email, the risk engine will only present such a challenge if no stronger option exists. We present a sample challenge served by Google in Figure 1. Failing a challenge does not allow an actor to subsequently select a weaker class of challenge; for legitimate users, this may mean account lockout until the user can regain access to a trusted network or device. This lock out friction mitigates the risk of downgrade attacks influencing the challenge selection process towards weaker challenges.

While we primarily focus on challenges shown during login and account recovery in this study, we also consider the usability and security of two-factor authentication. We provide a detailed breakdown in Table 1 of the contexts where a challenge may appear. Two-factor authentication always involves a phone or security key to solve a device-based challenge. During recovery, Google, in some cases, might allow users to solve weaker challenges such as providing an account's creation date or confirming access to any email account[2]. The recovery flow mitigates the hijacking risk of such knowledge-based challenges by requiring users to solve multiple, successive challenges and by blocking recovery attempts from devices or networks deemed as high-risk. As with login challenges, Google's recovery process elects the strongest challenge available based on the pre-negotiated secrets and recovery options established by an account holder.

3 DATASETS

In order to evaluate the security and usability of challenges in practice, we had to overcome two hurdles: (1) obtaining a dataset of challenges that hijackers attempted without our findings being biased by Google's risk detection; (2) obtaining a dataset of challenges served to legitimate users where memorability, misplaced devices, and an incentive to access to the account were all in play. We describe our approach to constructing each dataset below and the

[1]The risk engine also makes a judgment to the "liveness" of the device or phone number. This is why phone numbers can be part of an SMS challenge as well as a knowledge challenge. The process for determining liveness is beyond the scope of this paper.

[2]Only for accounts with no two-factor authentication enabled.

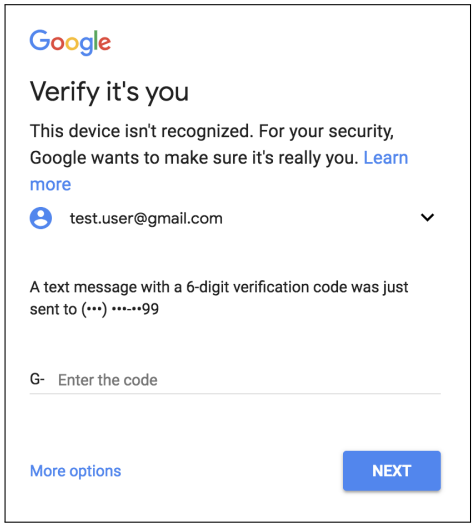

Figure 1: Sample challenge deployed as part of Google's risk engine for login and account recovery. The example shown requests that an account holder confirm an OTP sent via SMS.

Category	Description	Period	Samples
Usability	Challenge to login from familiar device or network.	Jun 19, 2017–Aug 15, 2017	300,000
	Challenge to login from unfamiliar device and network.	Feb 23, 2018–May 31, 2018	300,000
	Recovery challenge from familiar device or network.	Feb 23, 2018–Mar 22, 2018	300,000
	Two-factor challenge from familiar device or network.	Feb 23, 2018–Mar 22, 2018	300,000
Security	Challenge served to suspected automated bot.	Feb 23, 2018–May 31, 2018	300,000
	Challenge served to suspected phishing attack.	Feb 23, 2018–May 31, 2018	50,867
	Challenge served to suspected targeted attack.	Feb 23, 2018–May 31, 2018	453

Table 2: Datasets for studying usability and security performance for challenges.

ethical considerations involved. We provide a high-level overview of the resulting datasets in Table 2.

3.1 Security Datasets

For our study, we leverage three existing threat intelligence feeds that contain credentials stolen by automated bots, phishers, and targeted attackers. We retroactively joined these with records of logins to Google from February 23, 2018–May 31, 2018. For each login, we isolated sign-in attempts from hijackers (described below) and ultimately output whether Google's risk engine served a challenge in response to the hijacking attempt. Our resulting dataset consists of the type of challenge served and one of four outcomes: *passed* indicating the attacker supplied the correct challenge response, *failed* indicating an incorrect response, *skipped* indicating the attacker opted to try and solve an alternative challenge, or *abandoned* indicating the attacker exited the login flow and never supplied a solution.

3.1.1 Challenges to Automated Bots. Automated bots rely on third-party data breaches and brute force password guessing in an attempt to access millions of accounts every day. To detect this activity, Google scores every login using its passive reCAPTCHA system to identify bot activity [12]. We sampled a total of 300,000 login attempts flagged as bot activity, where the target account displayed previous organic sign-in activity (e.g., not a fake account). We manually evaluated metadata for 100 of these sessions and found all were indeed hijacking attempts. We then examined whether Google's risk engine triggered an accompanying challenge for each login. This occurred in every login attempt, resulting in a final dataset of 300,000 challenges. We caution that some bots may only be engaged in checking password validity and not attempt to solve a challenge. As such, we likely underestimate the capabilities of bots to automatically solve challenges. Furthermore, as bot detection plays a role in risk analysis, any estimate of overall hijacking protection provided against bots may be biased towards the types of bots that reCAPTCHA can identify.

3.1.2 Challenges to Phishers. Phishers harvest usernames and passwords, but also additional personal information including IP addresses and phone numbers in an attempt to bypass challenges [5]. Using data from Thomas et al. [22], we collected a sample of 400,000 Google accounts that had their valid password stolen between February 23–May 21, 2018. We then retroactively analyzed all Google sign-in attempts to these accounts during the same period. Only 20.4% of these accounts had any new sign-in activity, indicating phishers either waited on the stolen credentials or that another service may have been the target of interest. For accounts with some sign-in activity, we clustered each login in an effort to identify a single attacker accessing multiple phished accounts. The features we use for clustering are sensitive, but the overall technique identifies hijackers that re-use automation tools or resources when logging in. We note this clustering is independent of risk analysis and thus should not introduce any bias. In total, clustering identified 81,450 sessions as potential hijacking attempts. We manually evaluated metadata for 100 of these sessions and found all were indeed hijacking.

For each login attempt, we then examined whether Google's risk analysis engine triggered a challenge. In total, we identified 50,867 challenges served to phishers. In each case, the risk analysis engine triggered on a suspicious sign-in from a new country previously unassociated with the victim's account. The absence of a challenge in some cases stems from incorrect passwords (the login is rejected and no challenge shown) and because risk analysis will not trigger on repeated, successful accesses (if a phisher successfully accesses the account once, they will not be challenged in the future). We reiterate this analysis was retroactive; there were no additional protections we could provide to the victims at the time of the hijacker's sign-in attempt. In the course of our investigation, if we identified a successfully hijacked account, we forced the account into recovery thus blocking any further access by hijackers.

3.1.3 Challenges to Targeted Attackers. Our last security dataset consists of 214 Google accounts previously identified as victims of targeted attacks between February 23–May 21, 2018 [20]. We lack context for the attack techniques involved in every case, but many relate to spear phishing. We retroactively analyzed each account to isolate the hijacker's attempted login. From this, we identified 453 challenges that Google's risk engine served to attackers. As with phishing, this retroactive analysis precludes the possibility of providing any additional defense to victims at the time of hijacking. In the course of our investigation, if we identified a successfully hijacked account, we forced the account into recovery thus blocking any further access by hijackers.

3.2 Usability Datasets

In order to capture a spectrum of users and contexts, we developed four datasets of challenges to legitimate account holders: challenges to sign-ins from a familiar device or network (via an active experiment), organic challenges shown to sign-ins from an unfamiliar device or network, challenges shown during two-factor authentication, and challenges served during account recovery. As with our security dataset, we ultimately obtain only the type of challenge served and the resulting outcome.

3.2.1 Challenges to Familiar Devices & Networks. In order to gather a sample of challenges solved exclusively by legitimate users, we ran an experiment where we augmented Google's sign-in flow to challenge a random sample of user logins from familiar devices or networks. In the normal course of risk-based authentication, Google would not challenge these logins. We ran our experiment from June 19, 2017–August 15, 2017 and collected a total of 300,000 challenges and responses. We selected the challenge displayed randomly from a set of 9 available challenges.[3]

We ameliorated any risk of lock out by introducing a "Skip" button after a delay of 5 seconds which allowed users to bypass the experiment and continue to their account. Similarly, we allowed users to access their account regardless of the correctness of their response to the challenge. While this may skew results, ethically it presents the least friction to users. We caution that pass rates are thus likely lower than in practice.

3.2.2 Challenges to Low-risk New Devices & Networks. Given the bias of our previous experiment, we also consider a second set of organic challenges and responses served to a random sample of 300,000 user logins from February 23, 2018–May 31, 2018. Google's risk-analysis triggered these challenges due to users signing in from unfamiliar devices and networks, but considered them low risk as there were no other signs of automation or known hijacking access patterns. While the majority of such logins belong to legitimate users—risk analysis is biased towards false positives that lead to friction rather than false negatives that lead to hijacking—it may be that the dataset contains some hijacking attempts. We cannot empirically evaluate how many hijackings appear in this dataset due to a lack of supplementary signals.

3.2.3 Account Recovery Challenges to Familiar Devices & Networks. Account recovery captures users in moments where they have forgotten their password and potentially any other pre-shared secrets. To measure this effect on challenge pass rates, we randomly sampled 300,000 challenges shown organically as part of Google's account recovery flow from February 23, 2018–March 22, 2018. We limit our selection to flows initiated by users from familiar devices and networks to minimize any risk of including hijacker activity.

3.2.4 Two-factor Authentication Challenges to Familiar Devices & Networks. On the other end of the spectrum, users with two-factor authentication are primed to expect a challenge and to have a device on-hand. We randomly sampled 300,000 device-based challenges from February 23, 2018–March 22, 2018 for users with two-factor authentication enabled. We restrict our selection to sign-ins from familiar devices and networks to minimize any risk of including hijacker activity.

3.3 Ethics

Our in situ measurement of challenges raises a number of ethical concerns. While Google does not require IRB approval, we still took care to design our data collection to never capture sensitive private or identifying information related to the legitimate account holders under analysis. Our controlled experiment—the only instance where we introduce new challenges—also minimized any risk of account lock out. When analyzing hijacking attempts, our signals either derive from existing user protections or from auxiliary annotations that preclude real-time usage as a protection. As such, while we may learn of a successful hijacking, it is always after-the-fact. In those cases, we force the affected account into recovery in order to prevent any further damage by hijackers.

3.4 Limitations

Our security dataset is biased towards the class of bots, phishers, and targeted attackers for which we have threat intelligence and thus not a true random sample. Additionally, as attacks and capabilities may evolve over time, our measurements capture only a current snapshot of the effectiveness of challenges at preventing hijacking. Furthermore, as our experiment is in situ, we do not control the challenges these attackers receive. This may result in smaller samples for one type of challenge as compared to another. For example, if a victim has a device associated with their account it precludes showing any other class of challenges. We provide error margins on all measurements to account for small sample sizes. Finally, while we randomly sample Google's entire global user base to assess usability, in practice challenge solution rates likely differ by region. As such, any overall overall passrate statistics will be biased towards the underlying demographic and device accessibility distribution of Google's users.

4 THE SECURITY OF CHALLENGES

We begin our analysis by assessing the performance of individual challenges and ultimately the hijacking risk posed by bots, phishing, and targeted attacks. Our results illustrate that device-based and delegation-based challenges—despite their known limitations—protect against 100% of automated bots and over 92% of attacks rooted in phishing. Furthermore, we show that certain device-based and knowledge-based challenges outperform their category peers. Identity providers should prioritize these challenges when available.

[3]The experiment excludes challenges shown only during recovery or that require two factor authentication via an app or security key.

Category	Challenge	Prevention rate			Skip rate		
		Bot	*Phishing*	*Targeted*	*Bot*	*Phishing*	*Targeted*
Device-based	Authenticator OTP	100% ± 3%	94% ± 7%	–	0% ± 3%	38% ± 7%	–
	Device Prompt	100% ± 1%	99% ± 1%	90% ± 9%	0% ± 1%	45% ± 1%	43% ± 9%
	Device SMS OTP	100% ± 1%	96% ± 1%	76% ± 6%	0% ± 1%	38% ± 1%	12% ± 6%
	Device Settings OTP	100% ± 7%	100% ± 5%	–	0% ± 7%	49% ± 5%	–
	Security Key	100% ± 25%	100% ± 28%	–	0% ± 25%	25% ± 28%	–
Delegation-based	OTP to backup email	100% ± 1%	92% ± 3%	–	0% ± 1%	38% ± 3%	–
Knowledge-based	Account creation date	–	91% ± 5%	100% ± 22%	–	48% ± 5%	5% ± 22%
	Known pre-registered email	73% ± 2%	68% ± 1%	79% ± 26%	0% ± 2%	19% ± 1%	21% ± 26%
	Known pre-registered phone number	100% ± 16%	26% ± 1%	50% ± 20%	0% ± 16%	6% ± 1%	17% ± 20%
	Last login location	100% ± 28%	10% ± 2%	–	0% ± 28%	1% ± 2%	–
	Printed backup code	–	–	–	–	–	–
	Security question	–	93% ± 4%	–	–	38% ± 4%	–
Additional resource	OTP to any email address	–	61% ± 5%	–	–	33% ± 5%	–
	OTP to any phone	100% ± 1%	45% ± 2%	–	0% ± 1%	16% ± 2%	–

Table 3: Hijacking attempts from bots, phishers, and targeted attackers prevented by login or recovery challenges along with error margins for 95% confidence intervals. Device-based challenges prevent 94% of attacks rooted in phishing and 76% of targeted attacks.

4.1 Prevention rate per challenge

The primary security metric for any challenge is whether the information requested cannot be produced by a hijacker. Rogue sources of challenge solutions may come from third-party breaches, phishing, or other forms of observation such as malware on a victim's device. We measure a challenge's prevention rate as:

$$1 - \frac{passed\ challenges}{total\ challenges}$$

We provide a breakdown of prevention rates per challenge and class of attack in Table 3. We omit statistics in scenarios where we have fewer than 10 samples. We include error margins for 95% confidence intervals.

Automated Bots. We find that bots failed to solve 100% of device-based and delegation-based challenges in our sample. This indicates that automation tools lack both real-time access to OTPs and stepping-stone access to the target's secondary accounts. Similarly, bots appear to lack access to expensive-to-scale resources such as phone numbers for receiving an SMS challenge, failing 100% of such challenges. Knowledge-based challenges posed the only hijacking risk, with bots solving 27% of challenges asking for a pre-registered recovery emails. This likely stems from third-party breaches revealing billions of email records and potential recovery details [22]. In the same vein however, these bots do not appear to have access to prior login locations or phone numbers. We note that we cannot calculate the prevention rate of three challenges as they only appear in a recovery context where bots are outright blocked.[4] Our results illustrate that while third-party breach data may be readily available in underground communities, we do not observe its widespread use in hijacking attempts beyond access to valid passwords and secondary identities.

[4]Bots are not exclusively blocked from login in the unlikely event of a false positive, which might otherwise lock a legitimate user out.

Phishing. We find that no single category of challenges is immune to hijacking attempts rooted in phishing. Device-based challenges provided the strongest protection, preventing 94–100% of attacks in our sample. The presence of any successful hijackings indicates that a fraction of attackers can relay OTP codes (or have users click a prompt) within the challenge's freshness window. Given the remote nature of attackers, we believe this likely results from attackers coercing users through a man-in-the-middle attack. Zooming in, we find that phishers correctly solved 4% of SMS challenges compared to 1% of on-device prompts. We believe this statistically significant difference ($p < 0.0001$) results from the context that account holders have about the source of a sign-in. If a user enters their credentials into a man-in-the-middle phishing page, an SMS challenge relays only an OTP code with no additional information. In comparison, an on-device prompt reveals the device type (e.g., phone, desktop), operating system, and geolocation of the phisher who initiated the sign-in attempt. Account holders can use this information to then block the hijacking attempt. Alternatively, it is possible that attackers deceive cellular providers into handing over control of the victim's phone number. There is no equivalent remote attack on device prompts. Further, because there is no overhead for offline codes, the freshness window can be much shorter than for SMS codes. Offline codes from apps are refreshed every 30-60 seconds, whereas SMS based codes may be valid for hours or even days. These longer windows may be another reason that SMS challenges are more susceptible to phishing. In order to avoid all phishing risks, security keys provide the best protection, blocking 100% of attacks in our dataset.

Our phishing dataset also quantifies the risk of relying on secondary identity providers. Here, we find that phishers supplied proof of access to a backup email 8% of the time. Two possible explanations exist: hijackers either phished the OTP code from victims via a man-in-the-middle attack just as they did with device-based challenges, or the attackers had direct access to the backup account. If OTP phishing were the sole technique, we might expect similar prevention rates as OTP codes delivered via SMS (or

Challenge	Incorrect solution rate		
	Bot	*Phishing*	*Targeted*
Account creation date	–	70% ± 10%	–
Known pre-registered email	69% ± 3%	30% ± 3%	50% ± 40%
Known pre-registered phone	–	8% ± 3%	8% ± 27%
Last login location	–	8% ± 3%	–
Printed backup code	–	–	–
Security question	–	78% ± 7%	–

Table 4: Frequency that attackers supply incorrect solutions to challenges, exclusively for knowledge-based challenges.

even higher rates, due to users forgetting their backup account's password). Instead, the lower prevention rates for backup emails suggests that some hijackers have stepping-stone access to backup accounts, reducing the security of the overall scheme to that of the weakest trusted identity provider. Given the propensity of bots to also know a victim's backup email address, we believe it likely this access derives from third-party breaches.

Finally, we find that challenges asking for a user's phone number and last login location prevented only 10–26% of hijacking attempts. This information is likely stolen directly from users via a phishing page, something observed by Thomas et al. in their analysis of phishing kits [22]. We find that challenges asking for an account's creation date, backup email, and security questions prevented 68–93% of hijacking attempts. Security here stems from the inability of victims to recall the correct challenge solution and supply it to the phishing prompt. To illustrate this, we calculate the frequency that hijackers provided an incorrect solution for knowledge-based challenge:

$$1 - \frac{passed\ challenges}{total\ challenges - abandoned - skipped}$$

This metric differs from a challenge's prevention rate as we exclude samples where an attacker fails to provide any solution (e.g., they skip or abandon the challenge). As detailed in Table 4, we find that attackers provided an incorrect solution for a victim's account creation date in 70% of cases and an incorrect solution for a victim's security question in 78% of cases. In contrast, attackers supplied an incorrect phone number and login location for only 8% of such challenges. We confirm issues with memorability in non-phishing contexts later in Section 5.

Targeted Attacks. Our sample of targeted attacks emphasizes the weakness of mobile device-based challenges as a protection against concerted adversaries. We find that SMS challenges blocked only 76% of attacks and that device prompts blocked 90% of attacks. The security of device prompts over SMS is statistically significant ($p < 0.002$). Compared to large-scale phishing, targeted attackers are more adept at stealing SMS OTP codes, likely due to more sophisticated tools and more time available to expend per victim, perhaps employing social engineering as descried by Siadati et al. [21]. Our sample size for knowledge-based challenges is too limited to draw conclusions, but we argue such challenges are unlikely to perform any better against a targeted attacker than in the phishing context where memorability is also a factor. While not shown in Table 3, our sample also includes two targeted attacks on accounts with security keys. The second-factor prevented both attacks. This reiterates the importance of domain validation as part of two-factor authentication rather than just proof of access to a physical resource.

4.2 Overall hijacking prevention

While our focus thus far has been on per-challenge prevention rates, in practice, the protection provided by risk analysis is a holistic function of each user's security posture and the frequency of repeated hijacking attempts. We capture this by measuring the fraction of users in our dataset with any successful sign-in from a hijacker. This includes accounts that never received a challenge, or where multiple challenges may have been involved. As our bot dataset includes only a sample of 300K sign-in attempts, we rely on a separate snapshot of all bot sign-in attempts over a month long window. We find that Google's risk analysis engine blocked over 99.99% of automated hijacking attempts by bots and 92% of attacks rooted in phishing. For targeted attacks, we believe protection is better measured by the number of hijacking attempts by attackers—effectively the window in which an identity provider can trigger more rigorous protections. Successful attacks involved 1–9 challenges, while failed attacks involved 1–12 challenges before the attackers abandoned their efforts. We provide only ranges due to the qualitative nature of the sample (N=214) involved.

5 USABILITY OF CHALLENGES

Layered authentication schemes are counterproductive if legitimate users are unable to access their account from new devices and locations. We measure the friction introduced by each challenge in terms of whether users skip, abandon, or provide incorrect solutions to challenges. We also consider whether, despite failing a challenge, users ultimately retry and gain access to their account. Our results indicate that over 97.3% of users can navigate challenges successfully. For comparison, 95.8% of users can navigate password-only authentication according to the same metric.

5.1 Pass rate per login challenge

In the inverse of the security context, our primary metric for usability is whether legitimate account holders can correctly solve a login challenge, thus avoiding account lockout. We consider two sets of users: those logging in from a familiar device or network and those logging in from a new device and network (which as discussed in Section 3, may include hijacking attempts). For each dataset we measure the overall pass rate per challenge as:

$$\frac{passed\ challenges}{total\ challenges}$$

This metric includes multiple failure modes: users providing an incorrect solution to a challenge, users skipping a challenge in favor of an alternative, and users abandoning a sign-in attempt entirely. To isolate issues related to memorability or mistyping, we calculate the adjusted pass rate per challenge as:

$$\frac{passed\ challenges}{total\ challenges - abandoned - skipped}$$

Category	Challenge	Pass rate		Skip rate		Abandon rate		Adjusted pass rate	
		Familiar	*Unfamiliar*	*Familiar*	*Unfamiliar*	*Familiar*	*Unfamiliar*	*Familiar*	*Unfamiliar*
Device-based	Device Prompt	35%	38%	16%	30%	44%	23%	93%	87%
	Device SMS OTP	45%	62%	28%	12%	25%	20%	98%	93%
	Device Settings OTP	13%	9%	16%	34%	64%	48%	72% ± 1%	56% ± 2%
Delegation-based	OTP to backup email	22%	5%	18%	1%	56%	92%	90% ± 1%	94% ± 3%
Knowledge-based	Known pre-reg. email	48%	66%	6%	8%	35%	14%	83%	86%
	Known pre-reg. phone	64%	65%	5%	10%	27%	13%	96%	86%
	Last login location	62%	77% ± 7%	5%	5% ± 7%	23%	7% ± 7%	88% ± 1%	89% ± 7%
	Security question	37%	48% ± 1%	18%	23% ± 1%	33%	13% ± 1%	78%	77% ± 1%
Additional resource	OTP to any phone	43%	42%	9%	17%	45%	34%	96% ± 1%	92%

Table 5: Usability of individual challenges as assessed by their overall pass rates and detailed failure modes including users skipping, abandoning, or incorrectly answering a challenge.

We isolate the other failure modes by calculating the skip rate and abandon rate per challenge. We present our results in Table 5. Unless otherwise noted, all error margins are within 1% at a 95% confidence interval. We note that challenges related to Google's Authenticator app, security keys, account creation time, and proof of access to any email address are not present as they appear exclusively during two-factor authentication and recovery.

Familiar Device & Network. We find that ready access to a trusted mobile device proved challenging for most users. Only 35% of users successfully responded to an on-device prompt and another 45% successfully provided an SMS OTP. Asking users to access their device settings and supply an OTP code proved even more challenging, with only 13% of users passing the challenge. When confronted with a device-based challenge, 25–64% of users outright abandoned the sign-in attempt while another 16–28% opted to solve an alternative challenge. These failure modes highlight the friction introduced by requiring "something you have" for authentication, despite the ubiquity of mobile devices. All of these device-based methods also rely on action flows that users may never have previously encountered. If we isolate challenges solely to those where users attempted to provide a solution, we find that 93% correctly clicked "Yes" to the on-device prompt and 98% supplied the correct SMS code. In aggregate, our findings illustrate that while device-based security challenges provide the best security, there is a non-negligible risk that users may not have immediate access to their device.

OTP codes sent to a backup email had the second lowest pass rate at 22%. Users abandoned a majority of these–56%–either due to an inability to remember their alternative email and password or the overhead involved. We expect that these rates would be higher for service providers that, unlike Google, are not major email providers. Many users have one primary email address that serves as the backup and recovery mechanism for other web services, and many users use Gmail accounts for this, so we might expect that they have difficulty remembering the credentials to an account they do not necessarily use every day. It's possible that a significant fraction of the accounts in our sample are such accounts. Adjusted pass rates of 90% indicate that users had a harder time copying the supplied code, possibly due to having to memorize the code while switching between browser tabs. In comparison, SMS OTPs had higher adjusted pass rates at 98% likely due to having a second screen.

Knowledge-based challenges exhibited the lowest degree of user friction, with pass rates ranging from 37–64%. If we isolate our analysis solely to memorability, we find that users correctly recalled their phone number 96% of the time and their last login location 88% of the time. Secret questions proved the most difficult for users, with users opting to skip them three times more than other knowledge-based challenges and providing a correct solution in only 78% of cases. As previously noted by Bonneau et al., users provide fake responses to security questions which decreases their recall at a later time [2]. Taken as a whole, knowledge-based challenges provide the best overall usability—though at the expense of failing to prevent over 74% of phishing attacks. These trade-offs highlight the decision identity providers must make when determining whether to minimize hijacking or account lockout.

We find that challenges which require only a single, re-usable secret with nothing to carry such as a phone number or email address provide the best memorability and usability. Users struggle with more complex pre-shared secrets such as passwords at secondary verifiers or security questions. These basic knowledge challenges, though simple for users, are also the easiest for attackers to bypass. In contrast, device-based challenges exhibit slightly higher user friction while providing better security protections.

Unfamiliar Device & Network. Compared to familiar devices and networks, users logging in from unfamiliar devices and networks exhibited 0.3–28% higher overall pass rates with the exception of only three challenges as shown in Table 5. We believe this stems from users having both a higher incentive to gain access from their new device and more context for why a challenge occurred. For example, we find users were 20–209% less likely to abandon a challenge. Skip rates were 4–42% higher, indicating users attempted to navigate towards a challenge they could solve. If we isolate our analysis to issues of memorability, we find that users exhibited lower adjusted pass rates for all device-based challenges and resource-based challenges and similar adjusted pass rates for all knowledge-based challenges (with the exception of their phone number). Interpreting these results, we believe that the lower adjusted pass rates for device-based challenges stems from the same place as the lower abandonment rate—users who would otherwise

Category	Challenge	Pass rate				Adjusted pass rate			
		2FA	*Familiar*	*Unfamiliar*	*Recovery*	*2FA*	*Familiar*	*Unfamiliar*	*Recovery*
Device-based	Authenticator OTP	75%	–	–	46% ± 9%	98%	–	–	94% ± 13%
	Device Prompt	76%	35%	38%	55%	97%	93%	87%	89%
	Device SMS OTP	82%	45%	62%	42%	98%	98%	93%	92%
	Device Settings OTP	–	13%	9%	4% ± 19%	–	72% ± 1%	56% ± 2%	25% ± 49%
	Security Key	67% ± 1%	–	–	–	100% ± 1%	–	–	–
Delegation-based	OTP to backup email	–	22%	5%	22%	–	90% ± 1%	94% ± 3%	87% ± 1%
Knowledge-based	Account creation date	–	–	–	13%	–	–	–	34%
	Known pre-reg. email	–	48%	66%	–	–	83%	86%	–
	Known pre-reg. phone	–	64%	65%	61% ± 1%	–	96%	86%	83% ± 1%
	Last login location	–	62%	77% ± 7%	–	–	88% ± 1%	89% ± 7%	–
	Printed backup code	48% ± 1%	–	–	7% ± 6%	83% ± 2%	–	–	64% ± 18%
	Security question	–	37%	48% ± 1%	22% ± 1%	–	78%	77% ± 1%	38% ± 1%
Additional resource	OTP to any email address	–	–	–	19%	–	–	–	70%
	OTP to any phone	–	43%	42%	–	–	96% ± 1%	92%	–

Table 6: Variations in usability across contexts for two-factor authentication, login challenges, and account recovery. Two-factor authentication users have a higher likelihood of carrying a device, while users in recovery struggle more with knowledge-based challenges.

have abandoned the challenge in our familiar device and network experiment are instead attempting to solve it. Additionally, this dataset likely includes some hijacking activity, pulling down aggregate adjusted pass rates. As such, when interpreting usability, we believe pass rates for both our familiar and unfamiliar context datasets represent a lower bound.

5.2 Spectrum of preparedness

Beyond risk analysis at login, challenges also appear in two-factor authentication and account recovery. Combined, these three contexts form a spectrum of preparedness from users who are primed to expect a challenge to users in a forgetful frame of mind. We examine how these contexts influence overall pass rates and memorability in Table 6.

Users enrolled in two-factor authentication had access to their security key for 67% of sign-in attempts, while another 75–82% had access to their mobile device. These pass rates, while roughly twice as high as the login challenge context, illustrate the limitation of expecting users to have a device on hand. At the other end of the preparedness spectrum, we find that device-based challenges displayed during account recovery exhibited pass rates roughly in line with login challenges. Taken as a whole, while improving user familiarity with SMS OTPs or on-device prompts may help increase their pass rate, device accessibility will cause at least a quarter of users to fail such challenges.

Unfortunately, users who struggled to remember their password were equally likely to struggle with accessing their backup account. We find users passed only 22% of delegation-based challenges during recovery, in line with the pass rates of users in a login challenge context.

Finally, knowledge-based challenges proved even more challenging for users in a recovery context. Here, only 13% of users passed a challenge for their account creation date and 22% their security question. Even when we restrict failures to memory alone, we find adjusted pass rates for both challenges fall below 38%. Users were more apt to produce their phone number–succeeding 61% of the time–at rates in line with users in a login challenge context. Printed backup codes also proved difficult to locate, with only 7% of users passing the challenge during recovery. In contrast, two-factor authentication users who selected to submit a code ultimately succeeded in 48% of cases. Our results highlight how recovery introduces even more usability challenges than a login challenge context. Given the issues of memorization at play, we argue that associating a phone number with an account presents the best path towards helping users who forget their password—though this fails to mitigate the risk of phishing without also requiring an SMS OTP code.

5.3 Overall success rate

While exploring challenges in isolation lets us reason about factors that influence usability, the most critical metric for any layered authentication scheme is whether users can solve at least one available challenge and thus access their account. Here, we examine *all* sign-in attempts to Google over a one month period to determine (1) whether users succeeded in logging in during a single challenged session, and (2) whether any user who failed a challenge ultimately succeeded in accessing their account within one week.[5] For this computation, we segment users into three groups: those with two-factor authentication enabled and accessing their account from a familiar device or network; those attempting to recover access to their account from a familiar device or network; and users signing in from an unfamiliar device and network. We note that we lack an equivalent experiment for login challenges to users signing in from a familiar device or network as this scenario never occurs in situ. For comparison, we also calculate the success rate of password-only authentication for users signing in from familiar devices or networks.

[5]We are unable to provide more granular results due to the computation and anonymization involved.

Source of initial failure	Success rate	
	Same session	*Within one week*
Two-factor authentication	88.2%	98.4%
Login challenge,	47.8%	97.3%
Login challenge, device-based	59.3%	97.9%
Account recovery	49.7%	93.8%
Account recovery, device-based	46.6%	94.6%
Password-only	89.5%	95.8%

Table 7: Success rates of users who fail a challenge to gain access to their account within one week. Login challenges and two-factor authentication exhibit less user friction than passwords.

We caution that overall success rate does not equate to lockout as some users may infrequently access their account. Likewise, for login challenges, hijacking activity will depress per-session success rates.

We present our results in Table 7. We find that two-factor authentication exhibits similar user friction compared to password-only authentication. This holds both per-session and overall. Conversely, just 47.8% of challenged sign-ins result in success, with similar rates of 49.7% for account recovery. However, over the course of a week, users have an opportunity to regain access to a familiar device or network and thus eventually access their account. If we isolate our analysis to device-based challenges only, we find that success rates increased for login challenges, but decrease slightly for users attempting to recover their accounts.

These results illustrate that inherent trade-off of login challenges. Identity providers can significantly reduce the risk of hijacking, but the resulting friction increases the risk of account lockout. However, by allowing users to re-try or to find a trusted device or network, users can eventually access their account at rates similar to password-only authentication. The significantly higher success rate of two-factor authentication indicates that education and habituation play a major role in reducing lock out risks.

6 RELATED WORK

Numerous studies have explored knowledge-based challenges in isolation from a usability perspective. Bonneau et al. extensively examined issues with secret questions as secondary authentication measures [2]. They found that users do not answer questions truthfully—with a mind to increase their own security. Consequently, the authors found users were unable to recall their answers during recovery. Similarly, Jakobsson et al. showed that security questions are more effective when responses are restricted to boolean values rather than free-form responses [15]. While improvements to knowledge-based questions in general help to reduce the risk of lockout, our results emphasize that this class of challenges provides only limited protection against phishing and targeted attacks. However, they do provide value against automated attacks and thus there is merit to exploring usability improvements.

Given the push towards two-factor authentication, recent studies have also explored usability challenges with security keys. Das et al. examined barriers to security key adoption and found users were most concerned with losing their second factor and did not feel they needed the extra layer of security [6]. Lang et al. explored security keys and device OTPs from a practical deployment standpoint and found keys provided both better security and reduced the time that users spent authenticating [17]. We believe that on-device prompts can similarly provide a more seamless authentication experience compared to copying an SMS code, though at present SMS is a more familiar experience for most users. In a similar vein, Siadati et al. examined the phishability of SMS codes and found they could socially engineer 50% of participants into handing over an OTP code [21]. With improvements to warning language included in the text of the SMS, only 8% of participants handed over their OTP code.

Finally, in terms of potential new forms of challenges, research has recently started to explore social challenges. Alomar et al. provide an overview of proposals in this space and potential trade-offs [1]. Brainard et al. proposed vouching by a single additional party [4], where authentication becomes a transitive property. Alternatively, Jain et al. proposed asking users to identify recent activity from friends or to identify a list of friends—more akin to knowledge-based challenges [14]. We are unaware of any large-scale deployments of such schemes, but our taxonomy is comprehensive enough to capture the challenges in this space.

7 CONCLUSION

While the quest to replace passwords continues, authentication challenges offer an effective barrier against remote hijacking threats rooted in password theft. From a practical standpoint, we found that challenges, in conjunction with risk-aware authentication, blocked over 99.99% of automated hijacking attempts and 92% of attacks rooted in phishing at Google. These protections come at a cost of increased failed sign-in attempts from legitimate users, but with eventual success rates at levels similar to password-only authentication. We caution our metrics represent a current snapshot of an arms race. But unlike many cybercrime threats, users can take simple proactive steps to dramatically increase their security. Users who associate a device with their account can reduce their phishing risk by up to 99%. This approach provides similar levels of protection to two-factor authentication while removing the requirement of always having a device on-hand. However, as our data showed, risk-aware authentication cannot reliably protect against repeated, targeted hijacking attempts that involve social engineering. Here, security keys provide the only 100% guarantee of protection against remote password theft. By providing risk-aware authentication as a default protection and security keys as opt-in, users can choose the security and usability trade-off that best works for them.

8 ACKNOWLEDGEMENTS

We thank Elie Bursztein and Tadek Pietraszek for their feedback on developing our study. This work was supported in part by NSF award CNS-1717062 and gifts from Comcast, Google, Linkedin, and YouGov.

REFERENCES

[1] Noura Alomar, Mansour Alsaleh, and Abdulrahman Alarifi. Social authentication applications, attacks, defense strategies and future research directions: a systematic review. *IEEE Communications Surveys & Tutorials*, 2017.

[2] Joseph Bonneau, Elie Bursztein, Ilan Caron, Rob Jackson, and Mike Williamson. Secrets, lies, and account recovery: Lessons from the use of personal knowledge questions at google. In *Proceedings of the International Conference on World Wide Web*, 2015.

[3] Joseph Bonneau, Cormac Herley, Paul C Van Oorschot, and Frank Stajano. The quest to replace passwords: A framework for comparative evaluation of web authentication schemes. In *Proceedings of the Symposium on Security and Privacy*, 2012.

[4] John Brainard, Ari Juels, Ronald L Rivest, Michael Szydlo, and Moti Yung. Fourth-factor authentication: somebody you know. In *Proceedings of the Conference on Computer and Communications Security*, 2006.

[5] Marco Cova, Christopher Kruegel, and Giovanni Vigna. There is no free phish: an analysis of "free" and live phishing kits. In *Proceedings of the USENIX Workshop on Offensive Technologies*, 2008.

[6] Sanchari Das, Andrew Dingman, and L Jean Camp. Why johnny doesn't use two factor a two-phase usability study of the fido u2f security key. In *Proceedings of the International Conference on Financial Cryptography and Data Security*, 2018.

[7] Duo Security. Guide to two-factor authentication. https://guide.duo.com/, 2018.

[8] Facebook. Facebook has users identify friends in photos to verify accounts, prevent unauthorized access. https://www.adweek.com/digital/facebook-photos-verify/, 2010.

[9] Facebook. Improving account security with delegated recovery. https://www.facebook.com/notes/protect-the-graph/improving-account-security-with-delegated-recovery/1833022090271267/, 2017.

[10] David Freeman, Sakshi Jain, Markus Dürmuth, Battista Biggio, and Giorgio Giacinto. Who are you? a statistical approach to measuring user authenticity. In *Proceedings of the Network and Distributed System Security Symposium*, 2016.

[11] Google. Confirm your identity using your android device. https://support.google.com/accounts/answer/6046815?hl=en, 2018.

[12] Google. reCAPTCHA v3. https://developers.google.com/recaptcha/docs/v3, 2018.

[13] Google. Sign in faster with 2-step verification phone prompts. https://support.google.com/accounts/answer/7026266?&hl=en, 2018.

[14] Sakshi Jain, Juan Lang, Neil Zhenqiang Gong, Dawn Song, Sreya Basuroy, and Prateek Mittal. New directions in social authentication. In *Proceedings of the Workshop on Usable Security*, 2015.

[15] Markus Jakobsson, Liu Yang, and Susanne Wetzel. Quantifying the security of preference-based authentication. In *Proceedings of the Workshop on Digital Identity Management*, 2008.

[16] Mark Kaufman. Google Authenticator will add a formidable layer of protection to your e-mail account. https://mashable.com/2017/10/29/how-to-set-up-google-authenticator, 2017.

[17] Juan Lang, Alexei Czeskis, Dirk Balfanz, Marius Schilder, and Sampath Srinivas. Security keys: Practical cryptographic second factors for the modern web. In *Proceedings of the International Conference on Financial Cryptography and Data Security*, 2016.

[18] Microsoft. Azure active directory identity protection. https://docs.microsoft.com/en-us/azure/active-directory/active-directory-identityprotection, 2018.

[19] Grzegorz Milka. Anatomy of account takeover. In *Enigma*, 2018.

[20] Ariana Mirian, Joe DeBlasio, Stefan Savage, Geoffrey M. Voelker, , and Kurt Thomas. Hack for hire: Exploring the emerging market for account hijacking. In *Proceedings of The Web Conf*, 2019.

[21] Hossein Siadati, Toan Nguyen, Payas Gupta, Markus Jakobsson, and Nasir Memon. Mind your smses: Mitigating social engineering in second factor authentication. *Computers & Security*, 2017.

[22] Kurt Thomas, Frank Li, Ali Zand, Jacob Barrett, Juri Ranieri, Luca Invernizzi, Yarik Markov, Oxana Comanescu, Vijay Eranti, Angelika Moscicki, et al. Data breaches, phishing, or malware?: Understanding the risks of stolen credentials. In *Proceedings of the Conference on Computer and Communications Security*, 2017.

[23] Kurt Thomas and Angelika Moscicki. New research: Understanding the root cause of account takeover. https://security.googleblog.com/2017/11/new-research-understanding-root-cause.html, 2017.