The 2FA Illusion: Uncovering Weak Links of Web Account Access in the Wild

Ke Coby Wang
Visa Research

Sunpreet S. Arora
Visa Research

Michael K. Reiter
Duke University

***Abstract*—Single-factor authentication (1FA) and two-factor authentication (2FA) for secure and reliable website account access have become everyday tasks for most users. However, the complexity of integrating 1FA, 2FA, and password reset mechanisms makes real-world deployments challenging to navigate, leaving key questions about their implications for account security and accessibility unanswered. In this paper, we present a comprehensive investigation into the deployment of 1FA, 2FA, and password reset mechanisms across 50 major websites in six industries. By formally modeling account access and password reset patterns and applying Karnaugh maps for logical optimization, we uncover surprising consequences of current integrations of authentication mechanisms. We present key findings on the implications of modern authentication integrations for account security and accessibility, highlighting both the overestimated strengths and overlooked weaknesses of current deployments. Our research aims to provide a valuable and practical understanding of real-world authentication deployments for advancing web authentication practices.**

1. Introduction

As online services have integrated into nearly every aspect of daily life, the need for secure web authentication has become increasingly critical. For decades, passwords have remained the most prevalent method due to their implementation simplicity and user familiarity. However, passwords carry inherent weaknesses: they are often easy to guess [11], [12], [65], [86], [95], vulnerable to phishing attacks [26], [48], [63], and frequently exposed in large-scale data breaches [79]. The widespread habit of reusing passwords across multiple services further exacerbates these risks [19], [88]. Efforts to encourage the creation of strong and unique passwords [5], [28], [72], [87] have shown limited effectiveness due to the substantial cognitive burden placed on users [77]. Password managers [4], [7], [36], [50], [59] aim to ease this burden by helping users manage secure, distinct passwords across their accounts, yet concerns about their usability [17] and potential security flaws [3], [54], [74] have affected their widespread adoption. Additionally, while email-based password resets enable a convenient way of recovering account access when users forget their passwords [29], they further expand the attack surface of user accounts due to inherent vulnerabilities in email-based authentication [32], [42].

In response to password insecurity, FIDO2 passwordless authentication, commonly referred to as passkeys, has emerged as a promising solution that offers a stronger defense against phishing and credential theft [2]. The initial proposal of device-bound passkeys, i.e., a passkey bound to a specific device, enhances security but raises significant concerns regarding key management and account recovery, particularly in scenarios involving device loss [8], [56], [93]. To mitigate these challenges, synced passkeys were introduced and users can now back up and synchronize their passkeys across devices through cloud services provided by major platforms like Apple and Google. However, this improved usability comes at the cost of requiring complete trust in these providers for secure and reliable passkey storage and synchronization. Meanwhile, two-factor authentication has been widely endorsed by security standards [37], [43], industry guidelines [18], [64], and government regulations [1], [82], as well as by security experts [14], [58], to enhance protection by supplementing passwords with a second layer of verification. Common implementations include possession-based factors such as SMS or voice-based one-time codes [62] and authenticator apps that use HMAC- or time-based one-time passwords [60], [61]. However, even these second factors can be compromised, as attackers may gain access to them without directly obtaining the user's device [33], [51].

Given the complexity of web authentication, which must balance security, usability, and deployability [14], it is essential to examine how websites implement single-factor authentication (1FA), two-factor authentication (2FA), and password reset mechanisms to identify strengths and weaknesses in existing deployments. Prior studies [15], [53] that manually examined password reset practices found that 80%–89% of the studied websites support email-based password reset. In terms of 2FA, previous studies [31], [33] on real-world 2FA deployments indicate that only 42% of studied websites support 2FA, with authenticator apps and SMS-based one-time codes being the most popular second factors. Amft et al. [6] found that most websites with 2FA/MFA enabled provide poor or no guidance for account recovery on their help pages, and that half of account lockouts can be resolved through email alone. As passkeys have been viewed as a promising alternative to passwords due to better

2576-9103/25/$31.00 ©2025 IEEE
DOI 10.1109/ACSAC67867.2025.00060

security, a recent study [10] observes that approximately 30% of the websites examined currently support passkeys for user authentication.

While previous research investigated the deployments of 1FA, 2FA, and password reset mechanisms *separately* and provided insights into their potential benefits and limitations, there has been limited systematic research on how these components are integrated in the wild. Important questions regarding the security of web authentication and account accessibility remain unanswered, such as: *How do password reset mechanisms, when integrated with 1FA or 2FA, impact the account security and accessibility? When passkeys are implemented, how are they integrated with existing 1FA and 2FA methods? What roles do different factors, e.g., passwords, email services, SMS, and mobile phones, play individually or together in current web authentication deployments? Is a particular authentication factor or a set of factors sufficient or necessary for account access or password reset?* and so forth. This gap highlights the need for a comprehensive study to systematically and rigorously examine and evaluate current web authentication practices, with particular focus on how multiple authentication components are integrated by a website and how such integration affects account security and account accessibility.

In this paper, we fill this gap by conducting a thorough investigation into the current deployments of 1FA, 2FA, and password reset mechanisms across 50 major websites in six industries from August through October 2024. We systematize the current state of web authentication deployments and integrations, aiming to provide insights into the diverse patterns of account access and password reset employed by these major websites. We adopt a rigorous approach to analyzing these authentication deployments by formalizing account access and password reset patterns as logical expressions. Using Karnaugh maps [46], a tool often used for logical optimization, we systematically examine these patterns and identify weak links in the authentication setups of the studied websites.

To the best of our knowledge, this is the first work in the literature to apply such a rigorous approach to analyzing the implications for account security and accessibility resulting from integrations of 1FA, 2FA, and password reset mechanisms in the wild. This approach allows us to identify key insights into the security and usability of current authentication deployments. For example, we found that passwords have become increasingly irrelevant from both security and usability perspectives. Additionally, there is a clear divergence among websites regarding the role of FIDO passkeys when integrated into 1FA. Critically, we also found cases where deployed 2FA offers only the same or a similar level of security as 1FA, casting doubt on the true effectiveness of 2FA: does it deliver expected account security benefits or merely create an "illusion" of improved account security to give both users and web service providers peace of mind? Through these findings, our paper provides a more nuanced understanding of the weak links in real-world authentication deployments and reveals how different authentication methods can be integrated to either mitigate or inadvertently introduce these vulnerabilities.

To summarize, our contributions are as follows.

- We examine the deployments and integrations of 1FA, 2FA, and password reset mechanisms across 50 major websites in six industries and provide a comprehensive overview of how authentication methods are deployed and integrated today.
- We adopt a formal approach to modeling account access, password reset patterns, and their integration as logical expressions. By applying Karnaugh maps, we identify specific weak links in the current authentication deployments by major websites.
- Based on our investigation and analysis, we present 7 key findings that highlight important observations, overestimated strengths, and overlooked weaknesses in current authentication deployments, offering insights into their practical implications for account security and accessibility.

2. Background and Related Work

In this section, we provide background on common web authentication methods (Sec. 2.1) and discuss prior work that investigates real-world web authentication deployments (Sec. 2.2).

2.1. Common Web Authentication Methods

Passwords: Passwords have dominated web authentication due to deployment simplicity and ease of use [14]. However, the inherent vulnerabilities of passwords have been extensively studied since their inception [15], [25], [26], [39]: they are vulnerable to guessing attacks [12], [65], [86] and are frequently reused across web services [5], [19], [25], [28], [72], [92], making them vulnerable to credential stuffing and tweaking attacks [88]. Passwords are also subject to phishing [26], [48], [63] and, due to rampant data breaches, can be easily exposed to attackers on a large scale [80], [89]. Furthermore, even when users are notified about password vulnerabilities such as password reuse, they are often reluctant to change their credentials [34], [41].

Significant efforts have been made to improve password security, such as encouraging users to choose strong [9], [27] and unique passwords across different websites [5], [28], [72], [87], enabling detection of password reuse or credential abuse, adopting password hardening services [21], [66], [80], [88] and detection of password breaches [45], [89]. However, these measures fall short of making passwords substantially more secure in practice, due to the inherent weaknesses of passwords as a knowledge-based authentication factor: strong passwords are difficult to remember and recall [77] yet easily stolen due to the transferable nature of knowledge. So even when users follow the recommendations in choosing passwords, those strong passwords are still vulnerable to sophisticated threats like phishing attacks [26], [48] and large-scale data breaches [58], [79], [89]. From the usability perspective, users often forget their passwords [29],

[77]. In response, password managers are recommended to ease the burden of memorizing and managing passwords, but concerns about their usability [17] and security [3], [54], [74] have limited their widespread adoption. For this reason, password reset mechanisms have been a critical part of the entire web authentication life cycle [15]. Despite potential security vulnerabilities in current implementations and deployments [42], email-based authentication methods remain widely used for password resets [53].

Email-based one-time codes/links: Another widely adopted web authentication method is email-based one-time codes or URL links [53]. When a user requests authentication, the website generates a one-time short-lived code or URL link and sends it to the user's registered email address. The user with access to the code or link is viewed by the website as the legitimate user. Leveraging the fact that users frequently access their email services, this method allows users to access or recover their accounts without the need to register and manage additional credentials [30] thereby making authentication more deployable and convenient. However, aside from potential security flaws in its current implementations and deployments by websites [32], [42], part or even the entirety of the responsibility for defending against unauthorized account access is unilaterally shifted to the third-party email service provider. For example, if the user's email account is protected by weak authentication, such as an easily guessable password, user account security at the websites ultimately boils down to the password security at the email service provider. It would be even more concerning if the user's email account were used for authentication across a relatively large number of web accounts. Moreover, this method heavily relies on the availability guarantees offered by the email service. Technical issues with email servers [53], improperly configured spam filters, delivery delays, or the unavailability of email account credentials can prevent users from receiving the codes or links sent to their email accounts.

SMS/voice-based one-time codes: One-time codes via short message service (SMS) or voice calls are also commonly used for web authentication. In this case, a valid one-time code sent via SMS/voice calls to the user's registered phone number can assist the website in verifying the user's identity. This method is popular due to its ease of implementation by websites and the convenience it provides to users due to the wide use of mobile phones [96]. However, multiple security vulnerabilities associated with phone services have been effectively exploited by attackers [62]. Targeted social engineering attacks [83], mobile malware [20], [52], or SIM swapping attacks [51] allow attackers to access a user's SMS or phone calls without detection. Despite their short lifespan, one-time codes can still be phished through spoofed websites or applications [52]. Additionally, these one-time codes rely on trusted and reliable telecommunication services as well as the availability of the communication channel [81].

HMAC-/time-based one-time passwords: *HMAC-based one-time password (HOTP)* [60] and *time-based one-time password (TOTP)* [61] have been used as part of 2FA by many websites to enhance account security, and in some cases, to secure account recovery [35]. During the OTP setup phase, both the website and the user's OTP authenticator application agree on a randomly selected long-term shared secret (referred to as a long-term *seed* in standards [37]). During subsequent authentication, the website requires the user to provide a shared ephemeral secret (or *a nonce* [37]) derived from the long-term seed registered earlier, without needing to expose the long-term seed during authentication. A HOTP/TOTP authenticator application usually resides on a user's mobile device today [33]. Although OTPs are vulnerable to observation or guessing attacks due to their low entropy, attackers must act within a short window—typically within seconds or a minute—between nonce generation and expiration. This time constraint limits the potential harm of such attacks. However, a shared long-term secret may be leaked during synchronization [76] or in a data breach [78]. As Gilsenan et al. [33] point out, the security and usability of OTP authenticator applications are largely compromised by insecure and cumbersome backup mechanisms, which are intended to enable account recovery when the original OTP authenticator becomes unavailable.

Passkeys: FIDO2 passwordless authentication, referred to as *passkeys* [24], is a public key cryptography-based authentication method that relies on a set of protocols including WebAuthn [85] and CTAP [23]. To enhance usability and address account recovery concerns [49], [56], FIDO synchronized passkeys [22], i.e., passkeys that can be synchronized across different devices, have been proposed and pushed by major platform providers and websites today. More specifically, passkey providers allow a user to back up her passkeys to passkey-provider cloud storage and synchronize them across the user's devices, enabling seamless account access across multiple user devices and convenient passkey backup and recovery. However, the trade-off for enhanced usability and recoverability is a degradation in security [10], [44]: the security of all a user's synced passkeys for their web accounts depends entirely on the provider's implementation of passkey synchronization and storage, as well as the security of the user's passkey provider account.

2.2. Related Work

Several previous studies have examined the authentication deployments of websites in real-world settings. In 2010, Bonneau and Preibusch [15] conducted the first large-scale study of password implementations, collecting data from 150 websites on their password deployment practices. Their findings indicate that 80% of the studied websites allowed users to reset their passwords via email-based one-time codes or one-time links (and, for 1/3 of the websites, by sending the forgotten passwords back in the clear directly!) [15, Table 4]. The remaining websites either supported password resets through a combination of email and security questions [13], [68], [71] or security questions alone. In 2018, Quermann et al. [67] examined the

authentication mechanisms adopted by 48 different services, including websites, IoT (Internet of Things) or smart home devices, and mobile devices, showing that passwords are still the most prevalent authentication method and are supported by almost all services they investigated. In 2018, Li et al. [53] examined 239 high-traffic websites and observed that 89.1% and 16.7% of these websites allowed users to reset passwords using email or phone alone, respectively. Regarding 2FA or MFA, a study by Gavazzi et al. [31] of 208 widely used websites offering account creation reported that only 42.3% of accounts supported MFA. Gilsenan et al. [33] investigated 22 commonly used HOTP/TOTP authenticator applications and their backup and recovery methods. They found that, to back up the necessary secrets used by OTP authenticator applications in case of device loss, the security of backups often falls back to other factors with larger attack surfaces than OTPs, e.g., passwords or email-based authentication. Amft et al. [6] mined the help pages of 1,303 2FA/MFA-capable websites, simulated account lockouts on 71 live accounts, and attempted to recover access to these accounts. They found that 52.1% of the accounts could be recovered using only access to the registered email address. They also observed that help pages regarding MFA and recovery were unreliable. Regarding FIDO passkeys, which are considered a promising alternative with better security over passwords, Blessing et al. [10] examined 94 websites and found that 29.8% of them support passkeys for user authentication. Other prior work [55], [69] has also investigated real-world login policies, e.g., regarding HTTPS requirements, login rate limiting, login failures.

To the best of our knowledge, however, no prior work has examined the implications for account security and accessibility arising from the real-world deployments and factor composition, i.e., the integrations of 1FA, 2FA, and password resets as a whole, which we address in this paper.

3. Methodology

3.1. Scope

We focus on authentication methods deployed by web service providers to secure account access, excluding any client-side credential management *not* implemented by them. For example, we examine authentication methods like passwords or email/SMS-based one-time codes, as well as password reset mechanisms. We do not, however, consider steps users may take prior to account access, e.g., retrieving from a password manager or a notebook, as these steps are independent of the websites. We also do not consider authentication or verification processes that may be part of account access paths but are *not* implemented by the websites, such as authentication or verification for unlocking a phone, an authenticator, or a password manager.

For this reason, while there are conflicting opinions [49] about whether FIDO passkeys alone suffice to provide 2FA capability (and our findings in Sec. 4 confirm resulting divergence in real-world deployment), in our analysis, we have chosen to treat FIDO passkeys as a single authentication factor to align with our scope, i.e., *authentication factors verifiable by websites*, and to ensure a conservative perspective on security. Additionally, while our focus is not on single sign-on (SSO) where user authentication is managed by third-party identity providers, our tests included several major identity providers (which are themselves major email or social media service providers) and revealed how popular websites support SSO (see App. A).

Websites may seek to reduce user friction by leveraging risk-based authentication (RBA) or "session-based" authentication. RBA allows websites to skip invoking 2FA/MFA when a login attempt appears consistent with the user's typical login patterns (e.g., characterized by the user's IP address, user-agent string, device, and geolocation). However, because it is often (and intentionally) opaque whether a website implements RBA or how it is configured [31], we did not consider it in our study. We also do not consider "session-based" authentication, which allows users to bypass deployed 1FA or 2FA by leveraging the fact that they are already logged into their accounts, e.g., through a cookie.

3.2. Study design

To provide a comprehensive view of how leading online web service providers implement 1FA, 2FA, and password reset processes in real-world settings, we manually investigated 50 major websites in six industries over a three-month period, August through October 2024: online shopping, social media, entertainment, financial services, travel, and email services. For online shopping, social media, and entertainment, we identified the ten most popular websites based on their global traffic rankings [75]. For travel websites, we examined the five most popular airline and hotel websites in the U.S. based on their market share [16], [40]. We investigated the five most popular email service providers ranked by the number of global users [73]. For financial-services websites, because account creation is tied to users' true identities—and, in some cases, requires creating a bank account—we investigated five financial-service websites 1) with which at least one of the authors has an account, and 2) that rank among the top ten financial-service websites globally by traffic [75]. By doing so, our investigation aims to capture a diverse set of user authentication and security practices, as these industries vary significantly in terms of user expectations, security requirements, and usability needs.

The experimental procedure was designed as follows:

(1) **Initial 1FA Setup**: For each website where we did not already have accounts, we created a new account and registered the 1FA by default, which typically involves setting a new password in most cases. This step also involved setting up any other options provided during account creation, such as email address or phone number verification via one-time codes or links.
(2) **1FA Method**: After registering the account, we logged in using the newly set credentials to verify that the provided 1FA method(s) functioned as expected.

Symbol	Description
A	authenticator apps (and backup codes for recovery)
E	email-based one-time codes/links
P	passwords
K	FIDO passkeys
S	SMS/voice-based one-time codes
Q	security questions
X	account access
$\vee$	"or": commonly used for alternative methods
$\wedge$	"and": commonly used for 2FA/MFA
$\Leftarrow$	implied by (indicating sufficiency)
$\Rightarrow$	imply (indicating necessity)

TABLE 1: Symbols used in this paper.

(3) **Simulating Forgotten Password Scenario**: To examine the password reset process, we logged out and simulated a forgotten password scenario by initiating the "Forgot Password" process in a new tab in the private browsing mode. We followed the website's specific password reset procedure and set a new password.
(4) **2FA Setup (If Available)**: After successfully resetting the password, we logged back in and enabled 2FA if the website offered this option. If a specific website didn't support 2FA, we skipped the rest of the steps for this website.
(5) **2FA Method**: To validate the 2FA setup, we logged out and then logged in again via 2FA in a new tab in the private browsing mode. This step confirmed that the 2FA was working for each account as expected.
(6) **Simulating Forgotten Password with 2FA Enabled**: In this final step, we logged out again and simulated a second forgotten password scenario while 2FA was enabled. This allowed us to observe the password reset process under the added complexity of 2FA.

For each website, we recorded the following details: 1) supported authentication factors for 1FA; 2) the steps required to reset passwords with 1FA enabled; 3) all supported authentication factors for 2FA; 4) the steps required to reset passwords with 2FA enabled, including whether the 2FA setup added any additional security layers. Two researchers conducted the study: one performed all site workflows for our study and logged per-site entries; a second independently replicated the workflows and cross-checked the logs. Disagreements were resolved by jointly replaying steps using the logs.

Ethics: Our tests involved no personal data or human subjects. All manual inspections were conducted carefully to avoid placing undue load on websites, and no interactions with customer-support representatives were involved. While App. B lists all sites we tested, we withhold the identities of those where potential vulnerabilities were identified, in accordance with responsible disclosure practices; any vulnerability deemed (plausibly) critical will be reported to the relevant parties before we release our findings publicly.

3.3. Formalization

In this paper, to allow for a precise representation of account access and password reset patterns, the notations introduced in Table 1 formalize commonly used authentication factors within a symbolic logic structure. Each letter corresponds to access to a specific authentication factor or account. For example, P indicates access to a valid password and E access to email-based one-time codes or links. Similarly, S for SMS or voice-based one-time codes, A for HOTP/TOTP authenticator applications (and their backup codes), K for FIDO passkeys, Q for security questions, and X for account access. These symbols, together with implication symbols, i.e., $\Leftarrow$ and $\Rightarrow$, enable the construction of logical sequences that model access patterns in web authentication. For example, the pattern $X \Leftarrow P$ indicates that access (X) to a user's web account is implied by a valid password (P). $\vee$ and $\wedge$ represent "or" and "and" compositions of multiple authentication factors, respectively. So $X \Leftarrow P \vee E$ indicates that account access is implied by access to either a valid password *or* a valid one-time code or link sent to the registered email address. In contrast, $X \Leftarrow P \wedge E$ indicates that account access (protected by 2FA) can be granted given access to both a valid password *and* an email-based one-time code or link. P on the left side of $\Leftarrow$ means a password can be reset when the factors on the right side of $\Leftarrow$ are presented by the user (or an attacker). For example, $P \Leftarrow E$ represents the case where a password can be reset if the user (or an attacker) can access a valid email-based one-time code or URL link sent by the website.

Sufficiency and necessity: We model account access in two directions: *sufficiency* (or "is implied by", denoted by $\Leftarrow$) and *necessity* (or "implies", denoted by $\Rightarrow$). Sufficiency shows which authentication factors are sufficient to grant account access to a user or an attacker. For example, $X \Leftarrow P \vee E$ means that either a valid password or a valid email-based one-time code is sufficient for account access. Necessity, on the other hand, indicates which authentication factors are necessary to obtain account access. For example, $X \Rightarrow P$ represents that a valid password (possibly together with additional factors) must be presented for account access.

We distinguish between these two aspects because they offer different perspectives on *account security* and *account accessibility*. In terms of account security, *sufficiency* allows us to consider how many accounts could be compromised if an attacker obtains one or more credentials, while *necessity* helps us assess how many accounts would remain secure if one or more factors are reliably secure. From the account accessibility perspective, *sufficiency* enables us to see how many accounts a user can access when they have one or more valid credentials available. On the other hand, *necessity* is essential for understanding account accessibility when one or more factors are not available to the user, e.g., due to the password being forgotten, email/SMS service unavailability, or loss of device-based credentials.

To more clearly see why we analyze both sufficiency and necessity, consider a site that requires 2FA for access with

(a) 2FA: X⇐P∧S (b) Password reset: P⇐S (c) Integrating 2FA with password reset: X⇐P∧S and P⇐S (d) Simplified access pattern by Karnaugh maps: X⇐S

Figure 1: An illustrative example of how an integration of a 2FA access pattern and a password reset mechanism can be simplified using Karnaugh maps. Solid lines and dashed lines represent implications for account access and password reset, respectively.

the pattern X⇐P∧S and supports password reset via email, P⇐E. From the access rule alone, one might conclude that P is *necessary* (it appears in all observed sufficient conjunctions). However, when we compose reset with access, any factor that implies P can substitute for it, yielding X⇐E∧S. Thus, P is *not necessary* for account access in this example: an attacker (or user) with E and S can obtain account access without P. In comparison, S is *necessary* but not *sufficient*. In short, sufficiency does not necessarily reveal necessity in account access pattern analysis; necessity must be assessed from the *full set* of observed access and reset patterns.

Disjunctive normal form: In this paper, to provide a clear presentation of all possible account access or password reset options available at a website we tested, we write complex access patterns in disjunctive normal form, i.e., where the expression is represented as a disjunction (∨) of multiple conjunctions (∧) of authentication factors in our context. For example, X⇐(P∧E)∨(P∧S) indicates that account access is implied by access to P as the first factor *and* either E or S as the second for 2FA (or equivalently, X⇐P∧(E∨S) if written in conjunctive normal form).

Karnaugh maps: Karnaugh maps (K-maps) [46] are a tool used in Boolean algebra and digital logic design to simplify complex logical expressions and identify patterns in a systematic but intuitive manner. In this paper, we use K-maps to simplify complex access patterns, hoping to provide a clear view of potential vulnerabilities in authentication deployments where complex compositions of authentication components are involved.

An illustrative example of how Karnaugh maps can be used to simplify access patterns is shown in Fig. 1. In this example, a 2FA account access pattern, X⇐P∧S, is integrated (insecurely) with a password reset pattern, P⇐S, by reusing S for both 2FA and password reset. The resulting account access pattern, if simplified by Karnaugh maps, becomes X⇐S. Such simplification allows us to easily identify the weak link here: S alone is sufficient for account access, even though the account appears to be protected by 2FA. We will show later in our paper that such weak links are neither artificial nor hypothetical; they do exist in the current authentication deployments of major websites.

Threat model: In our paper, when we state that an attacker has access to a factor (or a credential for a factor), we assume the attacker can present a valid credential for that authentication factor for account access or password reset. We also assume that the attacker has access to user information that may be necessary for authentication but is not considered secret, e.g., usernames or user IDs. If a user's account username is her email or phone number, we assume the attacker can access this information as well. Additionally, the attacker may obtain personal data that is difficult to change yet may be required for account access on certain websites. This includes, but is not limited to, a user's name, date of birth, driver's license number, and social security number, potentially obtained from publicly accessible records, data breaches, or even (compromised) user emails or devices. Since such information is not intended to serve as an authentication secret and it is challenging to determine what the attacker might already know, we do not treat it as an additional authentication factor or security layer for account access. Finally, we do not consider cases where the attacker might bypass authentication through other means, e.g., using stolen cookies or sessions, or compromising the authentication server at the websites.

4. Authentication Deployment in the Wild

Based on our investigation, among the 50 websites we tested, 98.0% (49 of them) permit authentication with only a single factor and 66.0% (33 of them) support 2FA. Among them, only one enforces 2FA upon account creation, while the remaining 32 allow users to enable it at their discretion.

In this section, we present the results of our investigation as follows: in Sec. 4.1, we outline the account access patterns and distribution for the 49 websites protected by 1FA. Next, in Sec. 4.2, we examine the authentication methods chosen by these 1FA-protected websites to secure password reset. We then show the account access patterns and distribution for the 33 websites protected by 2FA (when 2FA is enabled) in Sec. 4.3. Finally, as these 33 2FA-protected websites also support passwords as their common first factor for 2FA, we discuss how they implement password reset when 2FA is enabled in Sec. 4.4.

Factor Sufficiency	Percentage (Count)
X⇐P	55.1% (27)
X⇐K∨P	20.4% (10)
X⇐E∨K∨P	4.1% (2)
X⇐E∨K∨P∨S	4.1% (2)
X⇐E∨P	4.1% (2)
X⇐E∨P∨S	4.1% (2)
X⇐P∨S	4.1% (2)
X⇐K∨P∨S	2.0% (1)
X⇐S	2.0% (1)
Total	100.0% (49)
Factor Necessity	**Percentage (Count)**
X⇒E∨K∨P∨S	100.0% (49)
X⇒E∨K∨P	83.7% (41)
X⇒K∨P∨S	83.7% (41)
X⇒K∨P	75.5% (37)
X⇒E∨P∨S	69.4% (34)
X⇒P∨S	61.2% (30)
X⇒E∨P	59.2% (29)
X⇒P	55.1% (27)
X⇒E∨K∨S	2.0% (1)
X⇒E∨S	2.0% (1)
X⇒K∨S	2.0% (1)
X⇒S	2.0% (1)

TABLE 2: Single-factor authentication access patterns of 49 (out of 50) websites that support 1FA.

4.1. Account access at 1FA-protected websites

Table 2 summarizes the 1FA account access patterns across 49 websites that support 1FA.

Factor sufficiency for account access (1FA): As shown in Table 2, only 1 website (X⇐S) does not support passwords for authentication. So the remaining 98.0% of the 1FA-protected websites tested grant account access when a valid password is presented. This observation highlights web service providers' continued preference and trust in passwords, despite the known vulnerabilities and security risks associated with passwords as a single authentication factor (see Sec. 2.1). Also, 42.9% of the tested websites support additional non-password factors, e.g., K, E, and S, as alternative 1FA methods. Among them, 30.6% support FIDO passkeys as an alternative to passwords. Finally, X⇐S accounts for only 2.0% and no tested websites use email-based authentication as the sole factor.

Factor necessity for account access (1FA): To gain account access, 55.1% of websites require solely a valid password. However, no website relies solely on K or E, and only one website uses S as the sole factor for access.

Key Finding 1

While 98.0% of the 1FA-protected websites support passwords, valid passwords are *necessary* only at **around half** of them for account access (when password reset is not considered).

4.2. Password reset at 1FA-protected websites

Factor sufficiency for password reset (1FA): Table 3 indicates a clear preference for using E or S for password reset, with nearly half of the websites (45.8%) implementing P⇐E∨S. The second most common pattern is E-only recovery, which is supported by 37.5% of websites. A smaller percentage of websites (6.3%) use S alone for recovery. Another 6.3% of websites implement more flexible recovery options like P⇐A∨E∨S, which allows a user to reset her password via A, E, or S. The least common patterns are P⇐A∨S and P⇐Q∧E which are available at only 2.1% of websites each. Overall, Table 3 shows that E remains the most commonly used authentication method for password resets, which accounts for 91.7% of the password-enabled websites. The second most commonly used factor for password resets is S—60.4% of the websites allow the use of S alone for password reset. In contrast, only 2.1% implements both Q and E for password reset, which shows that security questions are rarely used for password resets.

Factor Sufficiency	Percentage (Count)
P⇐E∨S	45.8% (22)
P⇐E	37.5% (18)
P⇐A∨E∨S	6.3% (3)
P⇐S	6.3% (3)
P⇐A∨S	2.1% (1)
P⇐E∧Q	2.1% (1)
Total	100.0% (48)
Factor Necessity	**Percentage (Count)**
P⇒A∨E∨S	100.0% (48)
P⇒E∨S	91.7% (44)
P⇒E	39.6% (19)
P⇒A∨Q∨S	10.4% (5)
P⇒A∨S	8.3% (4)
P⇒Q∨S	8.3% (4)
P⇒S	6.3% (3)
P⇒Q	2.1% (1)

TABLE 3: Password reset/recovery patterns of 48 (out of 50) websites that support passwords for 1FA.

Factor necessity for password reset (1FA): Table 3 shows that 39.6% of password-protected websites we tested rely on E for password reset. In contrast, websites rarely depend on S, Q, or A, possibly due to the smaller deployment costs provided by E [53]. Surprisingly, however, 91.7% of the websites require users to provide either E or S for password reset. These results highlight the importance of keeping at least one of E or S available for password reset.

4.3. Account access at 2FA-protected websites

Factor sufficiency for account access (2FA): Table 4 shows that the most common access pattern observed is X⇐(P∧A)∨(P∧S), accounting for 15.2% of the tested 2FA-protected websites . The second most common patterns

are X⇐P∧A and X⇐(P∧A)∨(P∧E)∨(P∧S), with each accounting for 12.1%. For the 2FA-protected websites tested, 36.4% allow account access through P∧E while 78.8% allow account access through P∧A and 81.8% through P∧S. Notably, for the 14 2FA-protected websites that support passkeys, 28.6% (4 out of 14) use passkeys as a single factor alternative to passwords, meaning that the user still needs to complete an additional authentication after passkey authentication. However, 71.4% (10 out of 14) websites use passkeys alone as a 2FA deployment—a successful passkey authentication represents a full two-factor authentication including a local user verification (e.g., PINs or biometrics) by a passkey authenticator and the passkey itself [38], [70]. This is a clear signal that, while web service providers are increasingly deploying passkeys, they have not yet reached a consensus on the role of passkeys in 2FA (see Sec. 6.3).

Factor Sufficiency	Percentage (Count)
X⇐(A∧P)∨(P∧S)	15.2% (5)
X⇐A∧P	12.1% (4)
X⇐(A∧P)∨(E∧P)∨(P∧S)	12.1% (4)
X⇐(A∧P)∨K∨(P∧S)	12.1% (4)
X⇐(A∧K)∨(A∧P)∨(K∧S)∨(P∧S)	9.1% (3)
X⇐(A∧P)∨(E∧P)∨K∨(P∧S)	9.1% (3)
X⇐P∧S	9.1% (3)
X⇐(E∧P)∨(P∧S)	6.1% (2)
X⇐(A∧E)∨(A∧K)∨(A∧P)∨(E∧S)∨(K∧S)∨(P∧S)	3.0% (1)
X⇐(A∧P)∨(A∧S)∨(E∧P)∨(E∧S)∨K∨(P∧S)	3.0% (1)
X⇐(A∧P)∨K	3.0% (1)
X⇐E∧P	3.0% (1)
X⇐(E∧P)∨K∨(P∧S)	3.0% (1)
Total	100.0% (33)
Factor Necessity	**Percentage (Count)**
X⇒A∨E∨K∨S	100.0% (33)
X⇒K∨P	93.9% (31)
X⇒A∨K∨S	63.6% (21)
X⇒P	57.6% (19)
X⇒A∨S	48.5% (16)
X⇒A∨E∨K	18.2% (6)
X⇒E∨S	18.2% (6)
X⇒A∨E	15.2% (5)
X⇒A	12.1% (4)
X⇒K∨S	9.1% (3)
X⇒S	9.1% (3)
X⇒E	3.0% (1)

TABLE 4: Two-factor authentication access patterns of 33 (out of 50) websites that support 2FA.

Key Finding 2

Among the **14** (out of 50) tested websites that support 2FA and FIDO passkeys, **28.6%** use passkeys as **a single factor**, while the remaining **71.4%** treat **passkeys alone** as providing **2FA** capability.

Factor necessity for account access (2FA): While only 57.6% of 2FA-protected websites require a valid password as one of the two authentication factors for account access, 93.9% require users to present either a valid password or passkey. Passkeys alone are not necessary for account access if a valid password is provided, but obviously, websites have reduced their reliance on passwords by supporting passkeys as password alternatives. Additionally, 63.6% of the websites require users to have their mobile phones available for account access to provide a valid K, A, or S. Furthermore, X⇒K∨A∨E∨S is observed at 100.0% of the websites, meaning that all websites require users to have access to at least one of the K, A, E, or S, possibly along with other factors like P, to complete 2FA.

Key Finding 3

No website we tested, whether 1FA or 2FA, solely relies on FIDO passkeys.

4.4. Password reset at 2FA-protected websites

Factor Sufficiency	Percentage (Count)
P⇐E	30.3% (10)
P⇐E∨S	21.2% (7)
P⇐(A∧E)∨(A∧S)	12.1% (4)
P⇐(A∧E)∨(E∧S)	12.1% (4)
P⇐(E∧S)	6.1% (2)
P⇐S	6.1% (2)
P⇐A∧E	3.0% (1)
P⇐(A∧E)∨(A∧S)∨(E∧S)	3.0% (1)
P⇐A∨E∨S	3.0% (1)
P⇐A∨S	3.0% (1)
Total	100.0% (33)
Factor Necessity	**Percentage (Count)**
P⇒A∨E∨S	100.0% (33)
P⇒E∨S	93.9% (31)
P⇒A∨E	66.7% (22)
P⇒E	51.5% (17)
P⇒A∨S	45.5% (15)
P⇒A	15.2% (5)
P⇒S	12.1% (4)

TABLE 5: Password reset/recovery patterns of 33 (out of 50) websites that support 2FA with 2FA enabled and passwords being the first factor.

Factor sufficiency for password reset (2FA): According to Table 5, the most frequently adopted password reset pattern is P⇐E, which accounts for 30.3% of the studied websites. Following closely, P⇐E∨S, which includes S as an alternative second factor for E for password reset, is adopted by 21.2% of the studied websites. Password reset patterns that integrate more complex combinations, e.g., E paired with either A or S (i.e., P⇐(A∧E)∨(E∧S)) and A paired with E or S (i.e., P⇐(A∧E)∨(A∧S)), are less frequent but still notable, each taking 12.1% of the cases. Less common patterns, including P⇐E∧S and P⇐S with each observed in only 6.1% of cases, indicate that, while they use S for password reset, whether requiring a second factor for password reset is not consistent across these 2FA-protected websites. Overall, the distribution reveals a preference for

using E for password reset in approximately two-thirds of 2FA-protected websites.

Table 5 also reveals two interesting findings regarding password reset practices among 2FA-protected websites. Only 36.4% of these websites require users to complete two-factor authentication when resetting their passwords, which means that 2FA for password reset is not required consistently across accounts. Indeed, the majority (63.6%) allow password reset through a single authentication factor. Also, over half (54.5%) of these websites allow password resets using E alone. This suggests that many sites prioritize convenience over security in password reset, potentially exposing accounts to greater vulnerability due to E's larger attack surface [42].

Factor necessity for password reset (2FA): Interestingly, when compared to Table 3, the reliance on E or S (i.e., E∨S) for password resets remains consistent between 1FA and 2FA settings, at 91.7% (1FA) vs. 93.9% (2FA), despite a significant difference in denominator—48 (1FA) vs. 33 (2FA)—as 15 of the 48 websites that support passwords as 1FA do not support 2FA. Also, all password-enabled websites rely on A∨E∨S for password reset, suggesting that they are the three most important factors for password reset today.

5. Analysis

5.1. Simplified account access at 1FA-protected websites

We combine 1FA account access and password reset patterns and simplify them via Karnaugh maps [90]. The simplified access patterns and their distributions are shown in Table 6. Derived from Table 6, Table 7 provides a different perspective by presenting the percentage of websites where a user or attacker can access a 1FA-protected account using the specified factors, either directly or through password resets.

As shown in Table 6, when password reset mechanisms are considered, most 1FA-protected websites can be accessed with E alone. Specifically, according to Table 7, user accounts at 89.8% (44 of 49) of the tested 1FA-protected websites can be accessed through E alone when password reset is leveraged.

Key Finding 4

Access to the **email account** associated with a website account is *sufficient* for account access at **89.8%** of the 1FA-protected websites tested. In particular, **neither the password nor access to a mobile device is necessary**.

According to Table 7, the most critical factor here is P, as P alone grants access to 98.0% of the websites (48 out of 49). In other words, an attacker with users' passwords could potentially compromise 98.0% of user accounts. On the other hand, if users manage their passwords effectively and reliably, they will retain access to 98.0% of their accounts.

Factor Sufficiency	Percentage (Count)
X⇐E	28.6% (14)
X⇐E∨S	26.5% (13)
X⇐E∨K∨S	22.4% (11)
X⇐E∨K	6.1% (3)
X⇐S	6.1% (3)
X⇐A∨E∨S	4.1% (2)
X⇐A∨E∨K∨S	2.0% (1)
X⇐A∨S	2.0% (1)
X⇐E∧Q	2.0% (1)
Total	100.0% (49)

TABLE 6: Simplified 1FA access patterns of 49 (out of 50) websites. Each access pattern is simplified via Karnaugh maps by combining 1FA access and password reset patterns for each studied website.

Factors	w/o P reset	w/ P reset
{P, S}	100.0% (49)	-
{P, A}	98.0% (48)	-
{P, E}	98.0% (48)	-
{P}	98.0% (48)	-
{A, E, K, S} (=M^+)	44.9% (22)	98.0% (48)
{E, K}	38.8% (19)	89.8% (44)
{E}	16.3% (8)	89.8% (44)
{A, K, S} (=M)	40.8% (20)	69.4% (34)
{K, S}	40.8% (20)	69.4% (34)
{S}	16.3% (8)	63.3% (31)
{A, K}	30.6% (15)	36.7% (18)
{K}	30.6% (15)	30.6% (15)
{A}	0.0%	8.2% (4)

TABLE 7: The percentage of websites at which a user or an attacker can access a 1FA-protected account using the indicated factors, without or with leveraging password resets. Here M^+ and M represent mobile phone access with or without access to emails on the phone, respectively.

E is the second most important factor for both users and attackers. While only 16.3% of the 1FA-protected websites can be accessed using E alone, an additional 73.5 percentage points can be accessed via password reset mechanisms where E alone is sufficient (see Table 3), resulting in a total of 89.8% of 1FA-protected web accounts accessible through this method. In other words, even without P, as long as users have access to their registered email accounts, or if an attacker has access to these email accounts, 89.8% of user accounts remain accessible to the user or the attacker, respectively. Similarly, without P, an attacker with access to S can directly access 16.3% of 1FA-protected web accounts. However, leveraging password reset mechanisms that use S increases this access to 63.3%. While K is sufficient for accessing 30.6% of the 1FA-protected websites, password reset does not provide access to additional websites in this case, because, according to our observations, passkeys are used exclusively for account access, not for password reset.

When an attacker compromises a user's mobile device (M), where credentials for K, S, and A are assumed available, 40.8% of the 1FA-protected websites we tested (20 out of 49) become vulnerable. However, if the attacker with

M access also uses password reset options to take over accounts, this vulnerability increases to 69.4% (34 out of 49) of the 1FA-protected websites. Going further, if we assume the attacker with M access also has access to the user's email services on the device (denoted by M^+), the percentage of potential account takeovers via password reset rises from 44.9% to 98.0%.

Key Finding 5

For the 98% of the 1FA-protected websites tested that support passwords for account access, **passwords** would, in fact, become **entirely irrelevant** for account security on approximately **70.0%** of these websites if one has access to the user's mobile device, on **89.8%** if one can access the user's email, and on **98.0%** if combined.

5.2. Simplified account access at 2FA-protected websites

Table 8 presents access patterns across 33 2FA-protected websites that are simplified using Karnaugh maps. The most commonly adopted access pattern, accounting for 12.1% of the cases, is X⇐(A∧E)∨(E∧S)∨K. However, account access granted solely through a single factor, e.g., X⇐E, X⇐S, X⇐A∨S, and X⇐(A∧E)∨S, account for 36.4% and 60.6% of the 33 tested websites in total, with K excluded and included, respectively. Specifically, as shown in Table 9, this includes 2FA-protected websites that in fact allow use of A (6.1%), E (18.2%), or S (27.3%) as single factors for account access.

Factor Sufficiency	Percentage (Count)
X⇐(A∧E)∨(E∧S)∨K	12.1% (4)
X⇐A∧E	9.1% (3)
X⇐(A∧E)∨(A∧S)	9.1% (3)
X⇐E∧S	9.1% (3)
X⇐E	9.1% (3)
X⇐(A∧E)∨(A∧K)∨(E∧S)∨(K∧S)	6.1% (2)
X⇐(A∧E)∨(A∧S)∨(E∧S)∨K	6.1% (2)
X⇐(A∧E)∨(A∧S)∨K	6.1% (2)
X⇐(A∧E)∨S	6.1% (2)
X⇐(A∧E)∨(A∧K)∨S	3.0% (1)
X⇐(A∧E)∨(E∧S)	3.0% (1)
X⇐(A∧K)∨(E∧S)∨(K∧S)	3.0% (1)
X⇐A∨E∨S	3.0% (1)
X⇐A∨S	3.0% (1)
X⇐E∨K∨S	3.0% (1)
X⇐E∨S	3.0% (1)
X⇐K∨S	3.0% (1)
X⇐S	3.0% (1)
Total	100.0% (33)

TABLE 8: Simplified 2FA access patterns of 33 (out of 50) websites. Each access pattern is simplified via Karnaugh maps by combining 2FA access and password reset patterns for each studied website.

According to Table 8, when P and E are available, user accounts at only 36.4% of the 2FA-protected websites (12 out of 33) are accessible without password reset. When P and A are available, accounts at 78.8% of the websites (26 out of 33) are accessible. The same percentage, 78.8% of the websites (26 out of 33) are accessible if one has access to both P and S. However, the percentage rises to 97.0% (32 out of 33) and 100% when the user or an attacker has access to {P, A, S} and {P, A, E, S} respectively.

Factors	w/o P reset	w/ P reset
{P, A, E, S}	100.0% (33)	-
{P, A, S}	97.0% (32)	-
{P, E, S}	84.8% (28)	-
{P, S}	81.8% (27)	-
{P, A}	78.8% (26)	-
{P, E}	36.4% (12)	-
{A, E, K, S} (=M^+)	42.4% (14)	100.0% (33)
{A, E, K}	42.4% (14)	87.9% (29)
{E, K, S}	42.4% (14)	81.8% (27)
{A, K, S} (=M)	42.4% (14)	69.7% (23)
{A, K}	42.4% (14)	48.5% (16)
{E, K}	30.3% (10)	45.5% (15)
{K, S}	42.4% (14)	60.6% (20)
K	30.3% (10)	30.3% (10)
S	0.0%	27.3% (9)
E	0.0%	18.2% (6)
A	0.0%	6.1% (2)

TABLE 9: The percentage of websites at which a user or an attacker can access a 2FA-protected account using the indicated factors, without or with leveraging password resets. Here M^+ and M represent mobile phone access with or without access to emails on the phone, respectively.

5.2.1. 2FA security reduced to 1FA security. As Table 9 shows, 2FA security falls back to 1FA security at 36.4% of tested websites that support 2FA. This occurs because these sites reuse the same non-password authentication factor, e.g., A, E, or S, for both *account access authentication as the 2nd factor* and *password resets* at the same time. This configuration effectively reduces the security of 2FA to that of 1FA, as compromising this single non-password factor (which is the second factor) allows an attacker to reset the password (obtaining the first factor) and bypass both factors of the 2FA, resulting in an account takeover.

While both websites and users may believe that 2FA provides an additional layer of protection for user accounts, users actually bear an extra authentication burden without gaining improved account security as expected. In other words, this costly layer of additional protection provided by 2FA in these cases is largely an "illusion" and fails to deliver the intended security benefits.

Key Finding 6

Due to insecure composition of 2FA and password resets, **only one** compromised factor can lead to account takeovers at **36.4%** of the **2FA-protected websites**.

5.2.2. 2FA security reduced to phone security. Table 9 shows a security concern in the implementation of 2FA on most 2FA-protected websites, where both authentication

factors rely on access to a user's mobile phone. Typically, these websites use a password as the first factor and a phone-based method, such as A or S, as the second. Additionally, the password can be reset using another phone-based method (different from the second factor).

While this setup is intended to provide an extra layer of protection compared to the insecure configuration discussed in Sec. 5.2.1, it introduces a potential weakness: an attacker with access to the user's mobile phone could access the user's 2FA-protected account if the account access is implied by phone access. In other words, an attacker who has access to FIDO passkeys, SMS/Voice-based one-time codes, and OTP authenticator applications on a *breached* phone can circumvent both factors required for 2FA without the additional attack effort or complexity typically expected in a secure 2FA setup.

Key Finding 7

Access to a user's mobile device is sufficient for account access at **69.7%** of the 2FA-protected sites. This number increases to **100%** with access to the user's **email account** and mobile device.

5.2.3. 2FA security reduced to password management security. Based on our study, for all websites that support TOTP/HOTP authenticator apps, users are given the option to save backup codes when they first register for TOTP/HOTP authentication. These codes allow for account recovery if users lose access to their authenticator apps (e.g., in the event of device loss). While backup codes (denoted by C) act as a recovery mechanism for authenticator apps (i.e., A⇐C for websites that support them), they essentially function as a knowledge-based authentication factor, similar to a system-generated strong password. Although most websites instruct users to store these backup codes securely, the responsibility for storing these codes falls entirely on the users. This transfer of responsibility introduces potential risks to account security, as websites cannot control or predict how users will handle these backup codes. This risk is particularly concerning when users store backup codes together with their (strong but hard-to-remember) passwords in the same password storage, e.g., a password manager, a cloud-based backup storage, or a physical password notebook. In such cases, according to Table 9, a subset of the 78.8% of 2FA-protected websites that grant account access with $\{P, A\}$ would become vulnerable: an attacker with access to the user's password storage, e.g., via a password manager breach [3], could access both P and A (via A⇐C).

6. Discussion

6.1. 1FA or 2FA? It's really passwords or phones?!

Almost all websites that support 2FA do not really enforce 2FA. Instead, they allow users to choose between 1FA and 2FA, which seems to be a choice between basic account security (with better account accessibility) and enhanced account security (with reduced account accessibility). However, a closer examination based on our study reveals that this choice is in fact less about 1FA vs. 2FA and more fundamentally about choosing between password/knowledge-based authentication vs. phone/possession-based authentication.

98.0% of the 1FA-protected websites we tested support passwords. In other words, a user can use her password as the sole factor for almost all websites if 2FA is either not supported or not enabled. However, when the user enables 2FA, the account security immediately shifts toward possession-based authentication: According to Key Finding 7, with their registered mobile phones (and credentials on them like A, S, and K) available, users can access their accounts at 69.7% of the 2FA-protected websites. Furthermore, if users can access their emails on their phone, as most users do today [84], a mobile phone is sufficient for account access at 100% of the 2FA-protected websites (and, according to Key Finding 5, 98% of the 1FA-protected websites!). For this reason, rather than between 1FA and 2FA, users are, in fact, choosing between using passwords alone and keeping a phone around for account access, aside from security considerations.

6.2. Another reason why "your password doesn't matter"

In 2019, researchers from Microsoft [58] argued that strong and complex passwords are "overkill" against password spray attacks. Meanwhile, even complex passwords alone still fall short in protecting user accounts from other common threats, including brute-force guessing after credential breaches, phishing, and keylogging, which led them to conclude that "your password doesn't matter." While their insight arises from inherent weaknesses of passwords, our study supports and extends this understanding from a whole new perspective. According to Key Finding 5 and Key Finding 7, as long as a user has her mobile device with email access, she can access any account without needing to remember or recall a single password, regardless of whether those accounts are protected by 1FA or 2FA. This finding indicates that device-based authentication is quickly supplanting reliance on traditional passwords, effectively making passwords optional in modern web authentication. In other words, our finding provides the missing half of the insight: passwords don't matter—not only because they are *not sufficient for account security* due to their inherent weaknesses, but also because they are *not necessary for account access* due to significant reliance on device-based authentication today. We believe this extended insight provides a valuable inspiration for the community to reconsider the role of passwords in today's authentication landscape.

6.3. Ambiguous role of FIDO passkeys in 2FA

Approximately one-third of the 2FA-protected websites tested use passkeys as a single-factor alternative to passwords, while the remaining two-thirds view passkeys as

a complete 2FA process already (i.e., local authenticator verification paired with passkey authentication) [38], [70]. In other words, enabling 2FA for an account has *no effect* on security when passkeys have been set up for account access in the latter case. This intriguing divergence highlights the ambiguous role of FIDO passkeys in current 2FA deployments.

On the one hand, websites that use passkeys as a single factor demonstrate a preference for improving account security by pairing passkeys with additional authentication factors. This configuration aims to improve account security by leveraging the security advantages of passkeys over passwords, especially against threats like server-side breaches, guessing and phishing. However, it also raises usability concerns, as the additional authentication steps may lead to user frustration or reduced adoption of 2FA due to the increased complexity of the authentication process.

On the other hand, passkeys alone are sometimes treated as satisfying 2FA requirements because users are expected to complete additional local verification (e.g., via PINs or biometrics) on the passkey authenticator before using a passkey. For this reason, some websites do not require an additional factor even when 2FA is explicitly enabled. Usability-wise, FIDO passkeys provide a user-friendly authentication experience by eliminating the need to memorize/recall and input credentials manually (e.g., as required by passwords or one-time passwords/codes). However, websites that are expected to be protected by 2FA but rely solely on passkeys may be affected by their underlying weaknesses [47], [94].

The ambiguous role of FIDO passkeys in 2FA highlights potential confusion in balancing security and usability when websites implement them. We urge the security community to address this issue effectively to better leverage the security and usability benefits that passkeys provide.

6.4. (In)Effectiveness of login/reset notifications

According to Key Finding 7, if an attacker gains access to both a user's mobile device and email account, they can compromise all of the user's accounts, whether they are 1FA or 2FA protected. To mitigate the impact of account takeovers, many websites today implement login or password reset notifications as an additional layer of security to alert users to potential unauthorized access or password changes. However, this defense can be easily nullified if the attacker controls the device or service through which these notifications are delivered, e.g., a mobile device or email service. In such cases, the attacker can intercept and delete these notifications or even disable them permanently for those accounts. Unfortunately, this can happen even before the user notices the unauthorized access or password reset.

To mitigate this risk, websites should avoid sending notifications or alerts through the same channels used for other authentication factors. Reusing them for multiple purposes without thorough scrutiny can compromise the security of the entire authentication configuration. This is similar to the risk posed by reusing a factor for both password resets and two-factor authentication, as highlighted in Key Finding 6.

6.5. Limitations

An inherent limitation of our study is the dynamic and evolving nature of authentication deployment by websites, which is often driven by technological advances, changes in security requirements, and user feedback. Therefore, the account access and password reset patterns observed in our study reflect the practices in place at the time of data collection and these patterns may have changed after our study was carried out. For this reason, we only provide a snapshot of authentication as implemented by web service providers at a moment in time. Nevertheless, we believe that our study exposes important findings that reflect challenges of a more persistent nature for the field to address.

Another limitation of our study is its relatively small sample size compared to some prior large-scale automated measurements. Accordingly, our findings should be read as descriptive of practices among prominent sites rather than exhaustive coverage of the web. We encourage larger-scale investigations to validate and extend these observations.

7. Conclusion

As web authentication continues to evolve in response to increasingly sophisticated threats while struggling to meet usability needs, understanding the practical implications of real-world deployments becomes critical. In this paper, we provide a comprehensive examination of modern web authentication practices, focusing on the deployment and integration of 1FA, 2FA, and password reset mechanisms across 50 major websites in six industries. By formalizing account access and password reset patterns as logical expressions and using Karnaugh maps for analysis, we identified weaknesses and vulnerabilities in current authentication deployments. Our paper offers 7 key findings that expose overestimated strengths and overlooked weaknesses in current authentication practices, revealing that despite increasing adoption of 2FA and FIDO passkeys, significant gaps in security and usability remain. We hope our research provides a valuable and practical understanding of real-world authentication deployments for advancing web authentication practices.

References

[1] "Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 (General Data Protection Regulation)," https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A32016R0679, 2016.

[2] "CNBC: Why passkeys from Apple, Google, Microsoft may soon replace your passwords," https://fidoalliance.org/cnbc-why-passkeys-from-apple-google-microsoft-may-soon-replace-your-passwords/, 2024.

[3] "LastPass' latest data breach exposed some customer information," https://www.theverge.com/2022/11/30/23486902/lastpass-hackers-customer-information-breach, Available on: 2024-03-01.

[4] 1Password, "1Password," https://1password.com/.

[5] J. Abbott, D. Calarco, and L. J. Camp, "Factors influencing password reuse: A case study," 2018.

[6] S. Amft, S. Höltervennhoff, N. Huaman, A. Krause, L. Simko, Y. Acar, and S. Fahl, ""We've disabled MFA for you": An evaluation of the security and usability of multi-factor authentication recovery deployments," in *30th ACM Conference on Computer and Communications Security*, 2023, pp. 3138–3152.

[7] Apple, "Keychain Services," https://developer.apple.com/documentation/security/keychain_services.

[8] S. S. Arora, S. Badrinarayanan, S. Raghuraman, M. Shirvanian, K. Wagner, and G. Watson, "Avoiding lock outs: Proactive FIDO account recovery using managerless Group Signatures," *Cryptology ePrint Archive*, 2022.

[9] A. Azimi and A. Azimi, "Encouraging users to improve password security and memorability," *International Journal of Information Security*, vol. 18, no. 6, 2019.

[10] J. Blessing, D. Hugenroth, R. J. Anderson, and A. R. Beresford, "SoK: Web authentication in the age of end-to-end encryption," in *25th Proceedings on Privacy Enhancing Technologies*, 2025.

[11] J. Blocki, B. Harsha, and S. Zhou, "On the economics of offline password cracking," in *39th IEEE Symposium on Security and Privacy*, 2018.

[12] J. Bonneau, "The science of guessing: analyzing an anonymized corpus of 70 million passwords," in *33th IEEE Symposium on Security and Privacy*, 2012.

[13] J. Bonneau, E. Bursztein, I. Caron, R. Jackson, and M. Williamson, "Secrets, lies, and account recovery: Lessons from the use of personal knowledge questions at Google," in *24th International World Wide Web Conference*, 2015.

[14] J. Bonneau, C. Herley, P. C. V. Oorschot, and F. Stajano, "The quest to replace passwords: A framework for comparative evaluation of web authentication schemes," in *33th IEEE Symposium on Security and Privacy*, 2012.

[15] J. Bonneau and S. Preibusch, "The Password Thicket: Technical and Market Failures in Human Authentication on the Web," in *Workshop on the Economics of Information Security*, 2010.

[16] Bureau of Transportation Statistics, "TranStats: Intermodal Transportation Database," https://www.transtats.bts.gov/, 2024.

[17] S. Chiasson, P. van Oorschot, and R. Biddle, "A usability study and critique of two password managers." in *15th USENIX Security Symposium*, vol. 15, 2006.

[18] Cybersecurity and Infrastructure Security Agency (CISA), "Two-Factor Authentication (2FA) Guidance," https://www.cisa.gov/two-factor-authentication, 2020.

[19] A. Das, J. Bonneau, M. Caesar, N. Borisov, and X. Wang, "The tangled web of password reuse," in *21st ISOC Network and Distributed System Security Symposium*, 2014.

[20] A. Dmitrienko, C. Liebchen, C. Rossow, and A. R. Sadeghi, "On the (in) security of mobile two-factor authentication," in *18th International Conference on Financial Cryptography and Data Security*, 2014.

[21] A. Everspaugh, R. Chatterjee, S. Scott, A. Juels, and T. Ristenpart, "The pythia PRF service," in *24th USENIX Security Symposium*, 2015.

[22] FIDO Alliance, "Passkeys: Accelerating the availability of simpler, stronger passwordless sign-ins," https://fidoalliance.org/passkeys/.

[23] FIDO Alliance, "Client to Authenticator Protocol (CTAP)," https://fidoalliance.org/specs/fido-v2.2-rd-20230321/fido-client-to-authenticator-protocol-v2.2-rd-20230321.html, 2023.

[24] FIDO Alliance. (March 2022) How FIDO Addresses a Full Range of Use Cases. https://fidoalliance.org/wp-content/uploads/2022/03/How-FIDO-Addresses-a-Full-Range-of-Use-Cases-March24.pdf.

[25] D. Florêncio and C. Herley, "A large-scale study of web password habits," in *16th International World Wide Web Conference*, 2007.

[26] D. Florêncio, C. Herley, and B. Coskun, "Do strong web passwords accomplish anything?" in *2nd USENIX HotSec*, 2007.

[27] D. Florêncio, C. Herley, and P. C. V. Oorschot, "An administrator's guide to internet password research," in *28th Large Installation System Administration Conference*, 2014.

[28] D. Florêncio, C. Herley, and P. C. V. Oorschot, "Password portfolios and the finite-effort user: Sustainably managing large numbers of accounts," in *23rd USENIX Security Symposium*, 2014.

[29] X. Gao, Y. Yang, C. Liu, C. Mitropoulos, J. Lindqvist, and A. Oulasvirta, "Forgetting of passwords: Ecological theory and data," in *27th USENIX Security Symposium*, 2018.

[30] S. L. Garfinkel, "Email-based identification and authentication: An alternative to PKI?" *IEEE Security and Privacy*, vol. 1, no. 6, 2003.

[31] A. Gavazzi, R. Williams, E. Kirda, L. Lu, A. King, A. Davis, and T. Leek, "A study of multi-factor and risk-based authentication availability," in *32nd USENIX Security Symposium*, 2023.

[32] N. Gelernter, S. Kalma, B. Magnezi, and H. Porcilan, "The password reset MitM attack," in *38th IEEE Symposium on Security and Privacy*, 2017.

[33] C. Gilsenan, F. Shakir, N. Alomar, and S. Egelman, "Security and privacy failures in popular 2FA apps," in *32nd USENIX Security Symposium*, 2023.

[34] M. Golla, M. Wei, J. Hainline, L. Filipe, M. Dürmuth, E. Redmiles, and B. Ur, ""what was that site doing with my Facebook password?" designing password-reuse notifications," in *25th ACM Conference on Computer and Communications Security*, 2018.

[35] Google, "Google Authenticator now supports Google account synchronization," https://security.googleblog.com/2023/04/google-authenticator-now-supports.html.

[36] Google, "Google Password Manager," https://passwords.google.

[37] P. A. Grassi *et al.*, "Digital identity guidelines: Authentication and lifecycle management," https://doi.org/10.6028/NIST.SP.800-63b, 2017, NIST Special Publication 800-63B.

[38] J. Gu, "Are passkeys 2FA?" https://www.beyondidentity.com/resource/are-passkeys-2fa, 2023.

[39] C. Herley, P. C. Van Oorschot, and A. S. Patrick, "Passwords: If we're so smart, why are we still using them?" in *13th International Conference on Financial Cryptography and Data Security*, 2009.

[40] T. Hinton, "The Role of Branded Chains in the U.S. Hotel Sector," https://www.statista.com/chart/32643/market-share-of-branded-hotel-chains-in-the-us/, 2024.

[41] J. H. Huh, H. Kim, S. Rayala, R. B. Bobba, and K. Beznosov, "I'm too busy to reset my linkedin password: On the effectiveness of password reset emails," in *35th ACM Conference on Human Factors in Computing Systems*, 2017.

[42] T. Innocenti, S. A. Mirheidari, A. Kharraz, B. Crispo, and E. Kirda, "You've got (a reset) mail: A security analysis of email-based password reset procedures," in *18th International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment*, 2021.

[43] International Organization for Standardization, *ISO/IEC 27001:2022 Information security, cybersecurity and privacy protection — Information security management systems — Requirements*. ISO, 2013.

[44] M. Islam, S. S. Arora, R. Chatterjee, and K. C. Wang, "Detecting compromise of passkey storage on the cloud," in *34th USENIX Security Symposium*. USENIX Association, Aug 2025.

[45] A. Juels and R. L. Rivest, "Honeywords: Making password-cracking detectable," in *20th ACM Conference on Computer and Communications Security*, 2013.

[46] M. Karnaugh, "The map method for synthesis of combinational logic circuits," *Transactions of the American Institute of Electrical Engineers, Part I: Communication and Electronics*, vol. 72, no. 5, 1953.

[47] D. Kuchhal, M. Saad, A. Oest, and F. Li, "Evaluating the security posture of real-world FIDO2 deployments," in *30th ACM Conference on Computer and Communications Security*, 2023.

[48] D. Lain, K. Kostiainen, and S. Čapkun, "Phishing in organizations: Findings from a large-scale and long-term study," in *43rd IEEE Symposium on Security and Privacy*, 2022.

[49] L. Lassak, E. Pan, B. Ur, and M. Golla, "Why aren't we using passkeys? Obstacles companies face deploying FIDO2 passwordless authentication," in *33rd USENIX Security Symposium*, 2024.

[50] LastPass, "LastPass," https://www.lastpass.com.

[51] K. Lee, B. Kaiser, J. Mayer, and A. Narayanan, "An empirical study of wireless carrier authentication for SIM swaps," in *16th Symposium on Usable Privacy and Security*, 2020.

[52] Z. Lei, Y. Nan, Y. Fratantonio, and A. Bianchi, "On the insecurity of SMS one-time password messages against local attackers in modern mobile devices," in *28th ISOC Network and Distributed System Security Symposium*, 2021.

[53] Y. Li, H. Wang, and K. Sun, "Email as a master key: Analyzing account recovery in the wild," in *37th IEEE Conference on Computer Communications*, 2018.

[54] Z. Li, W. He, D. Akhawe, and D. Song, "The emperor's new password manager: Security analysis of web-based password managers," in *23rd USENIX Security Symposium*, 2014.

[55] B. Lu, X. Zhang, Z. Ling, Y. Zhang, and Z. Lin, "A measurement study of authentication rate-limiting mechanisms of modern websites," in *34thAnnual Computer Security Applications Conference*, 2018.

[56] S. G. Lyastani, M. Schilling, M. Neumayr, M. Backes, and S. Bugiel, "Is FIDO2 the kingslayer of user authentication? a comparative usability study of FIDO2 passwordless authentication," in *41st IEEE Symposium on Security and Privacy*, 2020.

[57] C. Mainka, V. Mladenov, J. Schwenk, and T. Wich, "SoK: Single Sign-On security—an evaluation of OpenID connect," in *2ndIEEE European Symposium on Security and Privacy*, 2017.

[58] Microsoft, "Your Password Doesn't Matter," https://techcommunity.microsoft.com/blog/identity/your-password-doesnt-matter/731984, 2019.

[59] Mozilla, "Password manager achievement unlocked," https://www.mozilla.org/en-US/firefox/features/password-manager/.

[60] D. M'Raihi, M. Bellare, F. Hoornaert, D. Naccache, and O. Ranen, "HOTP: An hmac-based one-time password algorithm," *IETF RFC 4226*, 2005.

[61] D. M'Raihi, S. Machani, M. Pei, and J. Rydell, "TOTP: Time-based one-time password algorithm," *IETF RFC 6238*, 2011.

[62] C. Mulliner, R. Borgaonkar, P. Stewin, and J. P. Seifert, "Sms-based one-time passwords: Attacks and defense: (short paper)," in *10thInternational Conference on Detection of Intrusions and Malware, and Vulnerability Assessment*, 2013.

[63] A. Oest, P. Zhang, B. Wardman, E. Nunes, J. Burgis, A. Zand, K. Thomas, A. Doupé, and G.-J. Ahn, "Sunrise to sunset: Analyzing the end-to-end life cycle and effectiveness of phishing attacks at scale," in *29thUSENIX Security Symposium*, 2020.

[64] Open Web Application Security Project (OWASP), "OWASP Cheat Sheet Series — Authentication Cheat Sheet," https://cheatsheetseries.owasp.org/cheatsheets/Authentication_Cheat_Sheet.html, 2023.

[65] B. Pal, T. Daniel, R. Chatterjee, and T. Ristenpart, "Beyond credential stuffing: Password similarity models using neural networks," in *40th IEEE Symposium on Security and Privacy*, 2019.

[66] B. Pal, M. Islam, M. S. Bohuk, N. Sullivan, L. Valenta, T. Whalen, C. Wood, T. Ristenpart, and R. Chatterjee, "Might i get pwned: A second generation compromised credential checking service," in *31st USENIX Security Symposium*, 2022.

[67] N. Quermann, M. Harbach, and M. Dürmuth, "The state of user authentication in the wild," in *14thSymposium on Usable Privacy and Security*, 2018.

[68] A. Rabkin, "Personal knowledge questions for fallback authentication: Security questions in the era of Facebook," in *4thSymposium on Usable Privacy and Security*, 2008.

[69] S. A. Roomi and F. Li, "A Large-Scale measurement of website login policies," in *32ndUSENIX Security Symposium*, 2023.

[70] M. Sanders, "Passkeys, 2FA, TOTP: What's the difference?" https://blog.1password.com/passkeys-2fa-totp-differences/, 2023.

[71] S. Schechter, A. B. Brush, and S. Egelman, "It's no secret. Measuring the security and reliability of authentication via "secret" questions," in *30th IEEE Symposium on Security and Privacy*, 2009.

[72] T. Seitz, M. Hartmann, J. Pfab, and S. Souque, "Do differences in password policies prevent password reuse?" in *35th ACM Conference on Human Factors in Computing Systems*, 2017.

[73] SellCell, "Most Popular Email Providers by Number of Users," https://www.sellcell.com/blog/most-popular-email-provider-by-number-of-users/, 2023.

[74] D. Silver, S. Jana, D. Boneh, E. Chen, and C. Jackson, "Password managers: Attacks and defenses," in *23rdUSENIX Security Symposium*, 2014.

[75] Similarweb, "Top Websites Ranking — Most Visited Websites in June 2024," https://www.similarweb.com/top-websites/, 2024.

[76] Sophos News. (2023) Google leaking 2FA secrets? Researchers advise against new account sync feature — for now. https://news.sophos.com/en-us/2023/04/26/google-leaking-2fa-secrets-researchers-advise-against-new-account-sync-feature-for-now/.

[77] E. Stobert and R. Biddle, "The password life cycle," vol. 21, no. 3, 2018.

[78] The New York Times, "SecurID company suffers a breach of data security," https://www.nytimes.com/2011/03/18/technology/18secure.html, 2011.

[79] K. Thomas, F. Li, A. Zand, J. Barrett, J. Ranieri, L. Invernizzi, Y. Markov, O. Comanescu, V. Eranti, A. Moscicki, D. Margolis, V. Paxson, and E. Bursztein, "Data breaches, phishing, or malware? Understanding the risks of stolen credentials," in *24th ACM Conference on Computer and Communications Security*, 2017.

[80] K. Thomas, J. Pullman, K. Yeo, A. Raghunathan, P. Kelley, L. Invernizzi, B. Benko, T. Pietraszek, S. Patel, D. Boneh, and E. Bursztein, "Protecting accounts from credential stuffing with password breach alerting." in *28thUSENIX Security Symposium*, 2019.

[81] P. Traynor, W. Enck, P. McDaniel, and T. L. Porta, "Mitigating attacks on open functionality in SMS-capable cellular networks," in *12thACM International Conference on Mobile Computing and Networking*, 2006.

[82] UK National Cyber Security Centre (NCSC), "Multi-factor Authentication (MFA) — Why You Need it and How to Implement it," https://www.ncsc.gov.uk/guidance/multi-factor-authentication-online-services, 2022.

[83] E. Ulqinaku, H. Assal, A. AbdelRahman, S. Chiasson, and S. Capkun, "Is real-time phishing eliminated with FIDO? social engineering downgrade attacks against FIDO protocols," in *30th USENIX Security Symposium*, 2021.

[84] J. van Rijn, "The Ultimate Mobile Email Statistics Overview," https://www.emailmonday.com/mobile-email-usage-statistics/, 2024.

[85] W3C, "Web Authentication: An API for accessing Public Key Credentials Level 2," https://www.w3.org/TR/2021/REC-webauthn-2-20210408/, 2021.

[86] D. Wang, Z. Zhang, P. Wang, J. Yan, and X. Huang, "Targeted online password guessing: An underestimated threat," in *23rd ACM Conference on Computer and Communications Security*, 2016.

[87] K. C. Wang and M. K. Reiter, "How to end password reuse on the web," in *26th ISOC Network and Distributed System Security Symposium*, 2019.

[88] K. C. Wang and M. K. Reiter, "Detecting stuffing of a user's credentials at her own accounts," in *29th USENIX Security Symposium*, 2020.

[89] K. C. Wang and M. K. Reiter, "Using Amnesia to detect credential database breaches," in *30th USENIX Security Symposium*, 2021.

[90] K. C. Wang, S. S. Arora, and M. K. Reiter, "Artifact: The 2FA Illusion: Uncovering Weak Links of Web Account Access in the Wild," https://github.com/k3coby/kmap4auth, 2025.

[91] R. Wang, S. Chen, and X. Wang, "Signing me onto your accounts through Facebook and Google: A traffic-guided security study of commercially deployed Single Sign-On web services," in *33rd IEEE Symposium on Security and Privacy*, 2012.

[92] R. Wash, E. Rader, R. Berman, and Z. Wellmer, "Understanding password choices: How frequently entered passwords are re-used across websites," in *12th Symposium on Usable Privacy and Security*, 2016.

[93] L. Würsching, F. Putz, S. Haesler, and M. Hollick, "FIDO2 the rescue? platform vs. roaming authentication on smartphones," in *41th ACM Conference on Human Factors in Computing Systems*, 2023.

[94] T. K. Yadav and K. Seamons, "A security and usability analysis of local attacks against FIDO2," in *30th ISOC Network and Distributed System Security Symposium*, 2023.

[95] Y. Zhang, F. Monrose, and M. K. Reiter, "The security of modern password expiration: An algorithmic framework and empirical analysis," in *17th ACM Conference on Computer and Communications Security*, 2010.

[96] Zippia, "20 Vital Smartphone Usage Statistics [2023]: Facts, Data, and Trends On Mobile Use In The U.S." https://www.zippia.com/advice/smartphone-usage-statistics/.

Appendix A. The Case of Single Sign-On

Single Sign-On (SSO) enables users to access multiple web services or applications using a single identity provided by a trusted identity provider, e.g., Google, Facebook, or Apple. While SSO has been criticized for its potential security and privacy concerns [57], [91], it simplifies the login experience and reduces the need for users to manage multiple credentials across different accounts. Although this paper does not focus on delegated authentication, such as SSO, we provide observations on current SSO support across the websites we tested.

According to our investigation, 30.0% (15 out of 50) of the websites we examined support SSO. Google is the most popular SSO identity provider, accounting for 93.3% (14 out of 15) of SSO-enabled sites, followed by Apple and Facebook at 80.0% (12 out of 15) and 66.7% (10 out of 15), respectively. 4 of these 15 sites also support SSO from other identity providers, e.g., Amazon and X (formerly Twitter). Interestingly, SSO adoption also varies widely by category: 4 out of 10 shopping and entertainment websites, 6 out of 10 social media websites, and 1 email provider (using SSO provided by another email provider we tested). Notably, *none* of the travel or financial-service websites we tested support SSO, likely due to the need for stricter identity verification and concerns about relying on third parties for account security and privacy.

Appendix B. Tested Websites

TABLE 10: Tested websites by category

Shopping	**Social Media**
1. amazon.com	1. facebook.com
2. ebay.com	2. whatsapp.com
3. walmart.com	3. instagram.com
4. target.com	4. tiktok.com
5. costco.com	5. x.com
6. bestbuy.com	6. linkedin.com
7. etsy.com	7. snapchat.com
8. homedepot.com	8. pinterest.com
9. wish.com	9. discord.com
10. macys.com	10. reddit.com
Entertainment	**Travel**
1. netflix.com	1. delta.com
2. twitch.tv	2. united.com
3. spotify.com	3. aa.com
4. disneyplus.com	4. southwest.com
5. imdb.com	5. alaskaair.com
6. hulu.com	6. hilton.com
7. fandom.com	7. hyatt.com
8. bilibili.com	8. ihg.com
9. pixiv.net	9. marriott.com
10. youtube.com	10. wyndhamhotels.com
Email	**Financial**
1. gmail.com	1. paypal.com
2. icloud.com	2. chase.com
3. outlook.com	3. wellsfargo.com
4. yahoo.com	4. americanexpress.com
5. proton.me	5. citi.com