The Mobile Paradox: How Interface Friction Can Improve 2FA Security

Mohammed Jubur
College of Engineering and Computer Science
Jazan University, Jazan, Kingdom of Saudi Arabia
Email: mjabour@jazanu.edu.sa

***Abstract*—Push-based multi-factor authentication (MFA) is vulnerable to habituation and "push-fatigue": when notifications arrive, users approve reflexively. We ask how the *mobile* context—small screens and app switching—affects different user segments in compare-and-confirm (number matching) under a first-factor-compromise threat model. Using a controlled within-subjects study (N=65; 24 trials/participant; benign vs. attack; PC vs. phone), we analyze attack-trial correctness and decision time. The most surprising result is not a uniform mobile penalty but a *mobile improvement* for medium-skill participants: on attack trials they are almost 9 percentage points more likely to respond correctly on phones than on PCs, while high-skill users show little device difference and the small low-skill group exhibits ceiling effects. We hypothesize a Forced Focus mechanism: on phones, transient, full-screen prompts and tighter interaction loops reduce competing stimuli and force attention onto the compare-and-confirm task, offsetting switch costs for some users. Time-to-decision signatures and qualitative comments about app switching and small-screen comparison are consistent with this account. We translate these findings into deployable mitigations that treat interface friction as a design parameter: unify the comparison surface (deep-link/overlay), surface lightweight origin/time context, and apply device- and segment-aware escalation where risk signals or segments warrant it—hardening mobile MFA without broadly slowing everyone down.**

***Index Terms*—multi-factor authentication, push notifications, number matching, mobile security, usable security, habituation, human factors, context-aware authentication**

I. Introduction

Push-based multi-factor authentication (MFA) is popular because it is fast and familiar, yet it is vulnerable to habituation and "push-fatigue": when prompts arrive, users often approve reflexively. Compare-and-confirm ("number matching") has emerged as the prevailing mitigation, recommended in public guidance and widely enabled in practice, while longer-term policy favors phishing-resistant authenticators. In parallel, standards emphasize that authentication choices must balance assurance with usability and customer experience [1]–[3]. We adopt a standard first-factor-compromise threat model: an adversary who already knows (or has phished) the password can trigger plausibly legitimate push requests but cannot compromise the user's possession device. Within this setting, compare-and-confirm intentionally introduces a small amount of workflow friction to interrupt reflexive approvals without making authentication unusable.

What these policies do not address explicitly is *how the mobile context affects different users*. On smartphones, people frequently switch between the requesting app and the authenticator on a small screen, suggesting extra cognitive burden for some segments. Classic results in cognition and usable security predict that such app-switching introduces *switch costs*—slower responses and more errors—with larger effects for older or lower-skill users. A straightforward expectation follows: mobile should uniformly hurt correctness on challenging trials.

Analyzing data from a controlled within-subjects study (N=65; 24 trials/participant; benign vs. attack; PC vs. phone; 4 vs. 6 digits), and building on the dataset reported in [4], we find that this prediction does not fully hold. We uncover a *Mobile Paradox*: *medium-skill* participants are more likely to correctly detect attacks on a *phone* than on a *PC*, with an observed improvement of about nine percentage points on attack trials, while high-skill participants show little difference across devices and the small low-skill group exhibits near-ceiling behavior. Aggregate behavior aligns with prior reports—high benign true-positive rates and near-chance attack detection with no reliable main effects of device or code length—so the segment-specific mobile gain stands out [4].

We propose a *Forced Focus* explanation for this paradox. The friction of the mobile workflow—transient, salient prompts on a constrained surface and a tight interaction loop—acts as a context break that interrupts habituated "approve" behavior and forces attention onto the comparison task for some users, partially offsetting switch costs. Rather than treating mobile friction as a uniform negative, we argue it can be deliberately shaped: unify the comparison surface via overlays or deep links, surface lightweight origin/time binding, and escalate selectively based on device and risk. Our analysis quantifies *who* is most vulnerable during number-matching, on *which* device, and *why*, and translates these insights into mitigations aligned with current guidance [1]–[3].

Contributions. This paper makes three contributions:

- **Segment-aware risk in mobile number matching.** Using a within-subjects dataset (N=65; 24 trials/participant), we quantify Device×Segment effects on attack-trial correctness, decision time, and timeouts in push compare-and-confirm (number matching).
- **Mobile Paradox & Forced Focus.** We show that medium-skill users experience a mobile improvement on attack trials (≈9 percentage points phone>PC), advance the *Forced Focus* hypothesis, and provide correctness,

timing, and qualitative evidence consistent with this mechanism, while acknowledging its interpretive limits.
- **Deployable mitigations.** We derive practical guidance—one-screen compare flows (deep-link/overlay), lightweight origin/time context, and device- and segment-aware escalation—that treats interface friction as a design parameter and aligns with public recommendations on number matching and phishing-resistant MFA [1]–[3].

II. Background and Related Work

Push-based multi-factor authentication (MFA) balances stronger security with everyday convenience, but simple "approve/deny" flows are vulnerable to push-fatigue and push-bombing. Policy and provider guidance has converged on stronger session binding: CISA urges adoption of phishing-resistant authenticators and, where push remains, enabling compare-and-confirm (number matching), and major providers now surface richer context and enforce number matching in authenticator prompts [1], [2], [5]. In effect, these deployments intentionally introduce a small amount of workflow friction on the assumption that it will interrupt reflexive approvals without making authentication unusable.

This hardening runs alongside a usability mandate. NIST SP 800–63–4 stresses that authentication should make it "easy to do the right thing and hard to do the wrong thing" and be evaluated with representative users [3]. On mobile devices, compare-and-confirm often requires switching between a requesting app (or browser) and an authenticator on a small screen. Usable-security and HCI work highlights human effort and deployability as first-class criteria [6], and studies of comparison-based tasks (e.g., secure pairing, two-step methods on smartphones) document both perceived burden and performance differences driven by presentation and workflow [7], [8]. Together with task-switching and speed–accuracy results, this yields a standard expectation: additional steps and context switches typically incur "switch costs"—slower responses and more errors—especially for older or lower-skill users [9], [10]. Under this conventional view, mobile friction is primarily a liability to be minimized.

System-level work has also shown that notification-based flows remain exploitable when binding or context is weak. HIENA demonstrated that near-concurrent push prompts can be human-indistinguishable without strong session binding [11], and the Concurrent Login attack showed that even compare-/select-and-confirm variants can be abused when an adversary controls the terminal side and context is sparse [12]. In the wild, adversaries combine credential theft and session-cookie capture with push-fatigue techniques to obtain account access [13]–[16]. These papers establish exploitability of notification flows under weak binding; they do *not* quantify which user segments are most vulnerable *within* a correctly implemented number-matching flow or how device context shapes behavior. Our work is scoped to this behavioral question and complements prior attack demonstrations with a segment-aware view of where mobile friction helps and where it hurts.

III. Methodology

Study design and data. We reanalyzed data from a controlled, within-subject experiment evaluating compare-and-confirm ("number matching") push authentication [4]. We adopt a standard first-factor-compromise threat model: an adversary who has obtained the user's password can trigger plausibly timed push requests from a benign-looking terminal, but *cannot* compromise the user's possession device, notification channel, or authenticator app. The attacker may control the login terminal sufficiently to block or delay screens, open decoy pages, and initiate background authentication requests (same device, same IP), reflecting timing/concurrency conditions seen in notification-based MFA attacks [11], [12]. We do not assume cookie theft, network-layer MITM, or compromise of the mobile OS; phishing-resistant authenticators remain out of scope and are discussed separately in §V. Sixty-five participants each completed 24 trials balanced across three factors: *Device* (PC: login on desktop, confirm on phone; Phone: login and confirm on the same smartphone), *Condition* (benign vs. attack), and *Code length* (4 vs. 6 digits). In the Phone setting, participants switched between the requesting app (or browser) and the authenticator to compare codes.

Apparatus and procedure. Desktop sessions ran on Windows 10 with Google Chrome (stable); phone sessions used a recent Android device with an authenticator app that presented a foreground number-matching notification. UIs used consistent typography and contrast across devices; the only intentional difference between Device conditions was the app-switching step on Phone. Each participant completed 2–3 practice trials per Device, then 24 experimental trials. On each trial the participant initiated a login, received a number-matching notification, and was instructed: "Approve only when the numbers match and the request is expected; otherwise deny." Trials with no response within 30 s were marked as *Timeout*. Attack trials used the same flow but could include an intermediate/decoy page while a background request was issued.

Randomization and counterbalancing. The $2\times2\times2$ design (Device$\times$Condition$\times$CodeLen) yielded eight cells, each repeated three times (3 trials/cell). Trials were emitted in randomized blocks of eight containing one instance of each cell; the Device order in the first block was Latin-square counterbalanced across participants. This within-subject scheme balances all factors while dispersing Conditions and CodeLen across the session to limit simple order and fatigue effects.

Measures and segmentation. From the logs we derived three metrics: (i) *Correctness*, coded as 1 for correct approvals on benign trials and correct denials on attack trials, and 0 otherwise; (ii) *Decision time*, measured in seconds from prompt to action (winsorized at the 1st and 99th percentiles); and (iii) *Timeout*, flagged when no response occurred within 30 s (timeouts are excluded from correctness models and summarized descriptively to avoid conflation with incorrect responses).

Segment-aware analyses used self-reports from a post-study

questionnaire completed by 47 of the 65 participants. We defined two segmentation schemes: — *Skill* bins: Low = Poor, Medium = Fair, High = Good/Excellent; — Age bands: 18–24, 25–34, 35–44, 45+.

All summaries that marginalize over segments (by Device / Condition / Code length) use the full sample $N = 65$; any analysis that conditions on skill or age uses the subset with complete demographics ($N = 47$).

TABLE I
QUALITATIVE THEMES AND SAMPLE QUOTES.

Theme	Count	Example quotes
No challenges	22	"I had no challenges."
Switching between apps	9	"Switching between apps was frustrating."
Difficult to compare codes	8	"It was difficult to compare the codes."
Unsure whether to approve/deny	3	"I was unsure whether to approve or reject notifications."
Other	1	Non–

Statistical analysis. *Correctness.* We fit population-average binomial GLMs (logit link) at the participant×Device×Condition×CodeLen cell level with participant-clustered (sandwich) standard errors. Fixed effects included Device, Condition, CodeLen, Segment, and their interactions (e.g., Device×Segment). We initially attempted logistic GLMMs with participant random intercepts to capture within-subject correlation, but quasi-/complete separation in the very small Low-skill bin led to non-convergence. Following standard remedies [17], [18], we therefore report GLMs with cluster-robust covariance, and present odds ratios (ORs) with 95% CIs plus predictive margins (with CIs) for figures and contrasts.

Latency. To examine speed on attack trials, we analyzed time to *correct rejection* by Device×Skill using Kaplan–Meier curves and a Cox proportional-hazards (PH) model. Because only per-cell averages were available for each participant, the KM curves should be interpreted as qualitative timing signatures rather than full survival estimates, but they still reveal relative differences and heavy-tail behavior across groups [19].

Qualitative coding. Two researchers independently coded open-ended responses about challenges and likes, resolving disagreements by discussion. The most frequent themes were *switching between apps* and *difficulty comparing codes*; representative quotes and counts appear in Table I.

IV. RESULTS

We present a segment-aware analysis of attack-trial performance. We first quantify correctness differences across device context and user skill using a binomial GLM with a logit link; model-predicted probabilities (with 95% CIs) are shown in Fig. 1a, alongside observed attack-trial rates with Wilson CIs in Fig. 1b. We then examine decision-time dynamics on attack trials via Kaplan–Meier estimators and a Cox proportional-hazards (PH) model; survival curves appear in Fig. 2a and Fig. 2b. Finally, we relate participant comments to these behavioral patterns.

A. Correctness: The Mobile Paradox

The GLM reveals a statistically significant *Device* × *Skill* interaction on the probability of correctly rejecting an attack (*OR* = 0.58, 95% CI [0.34, 0.99], $p = 0.045$), indicating that the phone (vs. PC) effect depends on self-rated skill. As visualized in Fig. 1a, *High*-skill users are essentially flat across devices (predicted correctness: 40.4% on PC vs. 36.5% on phone). For *Medium*-skill users, correctness improves from 44.4% (PC) to 49.0% (phone), a counter-intuitive **Mobile Paradox**. Observed attack-trial rates with Wilson 95% CIs show the same pattern (Fig. 1b). For the small *Low*-skill bin, near-ceiling behavior yields wide uncertainty (quasi-separation), so we interpret those estimates cautiously. *Overall, correctness increases monotonically with self-rated skill (Fig. 1a), but the* shape *of the device effect differs by segment, which is the focus of our analysis.*

Interpretation. Contrary to a straightforward task-switching prediction (uniform mobile penalty), the phone context appears to introduce a *beneficial context break*: the additional step and focused prompt disrupt reflexive "approve" habits and elicit a more attentive compare-and-confirm action, particularly for medium-skill participants.

B. Decision-Time: Evidence for Forced Focus

Kaplan–Meier curves (Fig. 2a, Fig. 2b) show that *High*-skill users make faster correct decisions across devices, while at-risk segments exhibit heavier right tails. A Cox PH model confirms that lower-skill users are significantly slower overall to make a *correct* attack rejection (Hazard Ratio = 0.62, 95% CI [0.45, 0.86], $p = 0.004$). Notably, the PC × Low survival curve is the shallowest, indicating prolonged latencies and more late decisions/timeouts.

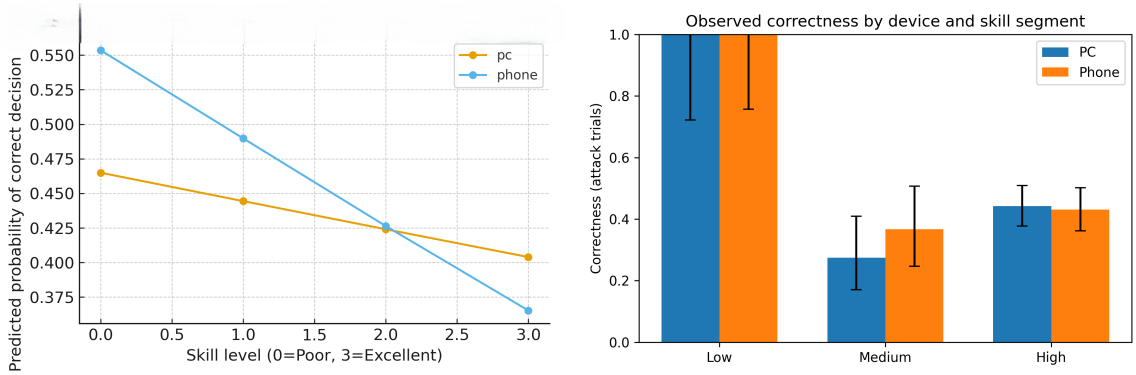
Interpretation. These latency signatures align with the *Forced Focus* hypothesis: on phones, the app switch and full-screen prompt create a brief context reset that increases attentional engagement. The interaction loop is not always faster, but for *Medium*-skill users it appears more deliberate and, ultimately, more accurate—mitigating reflexive errors induced by habituation in a more seamless PC environment.

C. Qualitative Corroboration

Post-study survey responses explicitly reference mobile friction: "*Switching between apps was frustrating*" and "*It was difficult to compare the codes*." As summarized in Table I, *switching between apps* (9 mentions) and *difficulty comparing codes* (8 mentions) were the dominant challenges. We interpret this reported "annoyance" as the subjective correlate of the same mechanism that underlies the Mobile Paradox: friction that interrupts habituation and nudges users into a verification-first mindset.

V. DISCUSSION

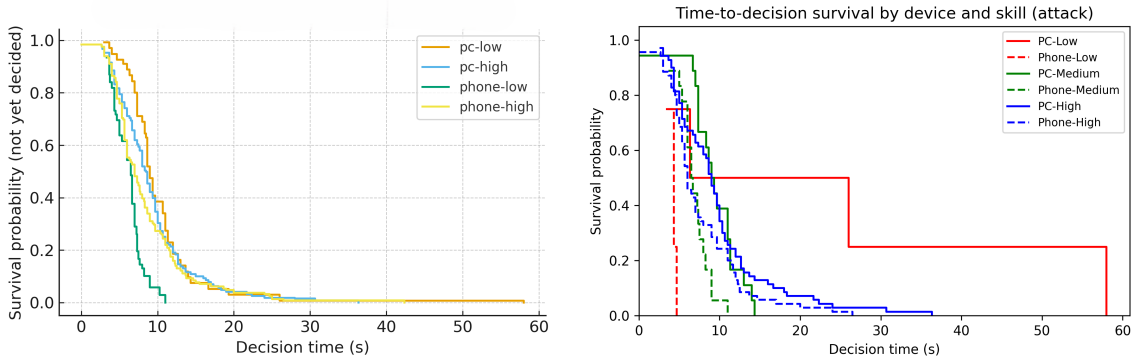
Our results surface a counterintuitive **Mobile Paradox**: medium-skill participants were *more* likely to correctly reject attacks on phones than on PCs, while high-skill users showed little difference. This runs against the straightforward prediction that mobile app-switching friction should uniformly harm performance. We interpret this pattern through a **Forced Focus** hypothesis: the mobile workflow—transient foreground prompts on a constrained surface with a tight interaction



(a) Model-predicted (GLM margins) with 95% CIs.

(b) Observed rates (attack trials) with Wilson CIs.

Fig. 1. Correctness by device and self-rated skill. Panel (a) shows the **Mobile Paradox**: medium-skill users improve on phones; high-skill users are stable across devices. Panel (b) shows the corresponding observed pattern.



(a) Four-curve overview (Low/High per device).

(b) Six-curve detail (Low/Medium/High per device).

Fig. 2. Time to *correct* attack rejection (Kaplan–Meier). The shallowest curve corresponds to PC × Low, indicating the slowest correct decisions and more late decisions/timeouts; higher-skill users respond faster overall.

loop—creates a brief context break that disrupts reflexive "approve" behavior. In contrast, the desktop flow can be *too* seamless, with the phone notification appearing as just another window on an already busy display, encouraging glance-and-approve habits. We emphasize that we do not directly observe attentional state; Forced Focus is a theoretically grounded behavioral interpretation of the observed Device×Skill interaction, supported by timing signatures and qualitative reports rather than a definitive causal proof.

A. Design Implications: Putting Forced Focus into Practice

Taken together, our findings suggest that interface friction should be treated as a design parameter to be *managed*, not eliminated. The goal is targeted, beneficial friction—enough to disrupt habituation for at-risk segments without degrading overall usability.

a) Preserve the Context Break, Reduce the Frustration: Because the dedicated, foreground confirmation step appears to curb reflex approvals, the design goal should not be to remove all friction, but to remove needless navigational hassle while preserving the beneficial "context break." A *one-screen compare* flow, using OS-level overlays or deep links, would allow the user to see the challenge and comparison cue in a single, focused view before being automatically returned to the requesting app. This preserves the attention benefit we observe on mobile while lowering reported frustration, and is compatible with guidance to strengthen push flows via number matching and richer context [1], [2].

b) Introduce a Forcing Function on PC: Our finding that the seamless desktop environment may encourage insecure habits motivates a small, targeted *forcing function* for at-risk segments or high-risk sessions. For example, requiring the user to type the first digit of the on-screen code or click a highlighted number before approval would simulate the Mobile Paradox "Forced Focus" effect on PC. This reframes friction as a potential security feature—introducing a brief moment of deliberate engagement—provided it is evaluated for accessibility and fatigue so that added effort remains proportionate to risk.

c) Enrich Prompts with Context by Default: To capitalize on moments of forced focus, prompts should improve discriminability between legitimate and fraudulent requests, especially on small screens. Binding the number-matching code to human-salient cues—service name/domain, approximate location, and time-since-request—raises attacker cost and complements session-binding measures emphasized in contemporary standards that couple security and usability by design [3].

d) Implement Device- and Segment-Aware Escalation: Our findings support a move away from one-size-fits-all authentication policies. Because risk depends on both device and user segment, systems should escalate sooner when signals

suggest higher susceptibility. For example, a mobile login from a lower- or medium-skill segment, or a burst of prompts over a short window, could trigger mandatory number matching with rich context by default, while lighter prompts are reserved for lower-risk scenarios. A natural next step is to embed such escalation rules in campaign-level simulations that model persistent, bursty push-bombing rather than isolated prompts.

B. Limitations and Future Directions

This work has several limitations. First, it uses a laboratory setting, synthetic attack prompts, and a convenience sample (N=65, of which 47 provided demographics); the small Low-skill bin yields wide uncertainty for that segment. Second, time-to-decision analyses rely on per-cell averages, so we do not capture full within-trial variability or detailed censoring effects. Third, we treat attack prompts as independent events and do not explicitly model adaptive, campaign-style adversaries who can tune timing, volume, or social-engineering content over time. Finally, we study behavior *within* a correctly implemented compare-and-confirm flow; our results complement, but do not replace, system-level defenses such as strong session binding and phishing-resistant authenticators.

These limitations point to concrete directions for future work. One is *field validation*: deploying one-screen compare flows and PC forcing functions at scale to test whether Forced Focus improves security without harming long-term usability. A second is *personalized, campaign-aware policies*: learning per-user thresholds and escalation rules that combine behavioral history with real-time risk signals, then evaluating them against realistic push-bombing campaigns. A third is *richer binding on small surfaces*: systematically testing which minimal context elements (origin, time, recognizable branding) most improve discriminability on phones and watches. These directions align with public guidance on phishing-resistant MFA and enhanced push flows, as well as the usability mandate in modern identity standards [1]–[3].

VI. Conclusion

This paper examined how mobile device context shapes the security of compare-and-confirm push authentication. In a 65-participant within-subjects study, we found a **Mobile Paradox**: medium-skill users were significantly more likely to correctly detect attacks on a phone than on a PC, while high-skill users showed little device difference. We proposed a **Forced Focus** explanation—mobile friction acting as a beneficial context break that interrupts habituated, reflexive approvals—and supported it with converging correctness, timing, and qualitative evidence.By moving beyond aggregate failure rates to a segment-aware analysis, we show that friction is not uniformly harmful and challenge the assumption that usability and security are always in opposition. Our design implications—one-screen compare flows, PC forcing functions, and device-/segment-aware escalation with lightweight origin/time binding—treat interface friction as a tunable security control rather than a flaw. Taken together, these results offer an evidence-based path toward push-based authentication systems that are more secure *and* more thoughtfully aligned with the realities of human attention and behavior.

References

[1] Cybersecurity and Infrastructure Security Agency (CISA), "Implement number matching in mfa applications," Cybersecurity and Infrastructure Security Agency, Fact Sheet, 2022. [Online]. Available: https://www.cisa.gov/sites/default/files/publications/fact-sheet-implement-number-matching-in-mfa-applications-508c.pdf

[2] ——, "Implementing phishing-resistant mfa," Cybersecurity and Infrastructure Security Agency, Fact Sheet, 2022. [Online]. Available: https://www.cisa.gov/sites/default/files/publications/fact-sheet-implementing-phishing-resistant-mfa-508c.pdf

[3] National Institute of Standards and Technology (NIST), "Digital identity guidelines," https://pages.nist.gov/800-63-4/, 2025, special Publication 800-63-4 (Web Portal).

[4] M. Jubur, N. Saxena, and F. A. Reegu, "Usability and security analysis of the compare-and-confirm method in mobile push-based two-factor authentication," *IEEE Transactions on Mobile Computing*, vol. 24, no. 6, pp. 4623–4638, 2025.

[5] Microsoft, "What is the microsoft authenticator app?" https://support.microsoft.com/en-us/account-billing/about-microsoft-authenticator-9783c865-0308-42fb-a519-8cf666fe0acc, 2025, accessed: 2025-07-21.

[6] J. Bonneau, C. Herley, P. C. van Oorschot, and F. Stajano, "The quest to replace passwords: A framework for comparative evaluation of web authentication schemes," in *IEEE Symposium on Security and Privacy (S&P)*, 2012, pp. 553–567.

[7] E. Uzun, K. Karvonen, and N. Asokan, "Usability analysis of secure pairing methods," in *International Conference on Financial Cryptography and Data Security*. Springer, 2007, pp. 307–324.

[8] D. N. Reese, E. Long, J. M. Ranney, and J. V. Manzo, "Usable and secure smartphone authentication: An evaluation of two-step verification methods," *Journal of Information Security and Applications*, vol. 45, pp. 1–12, 2019.

[9] S. Monsell, "Task switching," *Trends in Cognitive Sciences*, vol. 7, no. 3, pp. 134–140, 2003.

[10] R. D. Rogers and S. Monsell, "Costs of a predictable switch between simple cognitive tasks," *Quarterly Journal of Experimental Psychology A*, vol. 48, no. 2, pp. 289–304, 1995.

[11] M. Jubur, P. Shrestha, N. Saxena, and J. Prakash, "Bypassing push-based second factor and passwordless authentication with human-indistinguishable notifications (hiena)," in *Proceedings of the ACM Asia Conference on Computer and Communications Security (ASIA CCS)*. ACM, 2021. [Online]. Available: https://dl.acm.org/doi/10.1145/3433210.3453084

[12] A. T. Mahdad, M. Jubur, and N. Saxena, "Breaking mobile notification-based authentication with concurrent attacks outside of mobile devices," in *Proceedings of the 29th Annual International Conference on Mobile Computing and Networking (MobiCom)*. ACM, 2023. [Online]. Available: https://dl.acm.org/doi/10.1145/3570361.3613273

[13] J. M. Esparza, "Understanding the credential theft lifecycle," *Computer Fraud & Security*, vol. 2019, no. 2, pp. 6–9, 2019.

[14] K. Gretzky, "Evilginx - advanced phishing with two-factor authentication bypass," 2017, Accessed; Last accessed 19 April, 2019. [Online]. Available: https://breakdev.org/evilginx-advanced-phishing-with-two-factor-authentication-bypass/

[15] ——, "Evilginx2: A man-in-the-middle phishing framework for capturing credentials and session cookies," https://github.com/kgretzky/evilginx2, 2018, accessed: 2025-07-21.

[16] BeyondTrust, "MFA Fatigue Attack," 2023, accessed: January 16, 2023. [Online]. Available: https://www.beyondtrust.com/resources/glossary/mfa-fatigue-attack

[17] G. Heinze and M. Schemper, "A solution to the problem of separation in logistic regression," *Statistics in Medicine*, vol. 21, no. 16, pp. 2409–2419, 2002. [Online]. Available: https://onlinelibrary.wiley.com/doi/10.1002/sim.1047

[18] A. Zeileis, "Object-oriented computation of HC and HAC covariance matrix estimators," *Journal of Statistical Software*, vol. 16, no. 9, pp. 1–16, 2006. [Online]. Available: https://www.jstatsoft.org/v16/i09

[19] E. L. Kaplan and P. Meier, “Nonparametric estimation from incomplete observations,” *Journal of the American Statistical Association*, vol. 53, no. 282, pp. 457–481, 1958. [Online]. Available: https://www.jstor.org/stable/2281868