HapticLock: Eyes-Free Authentication for Mobile Devices

Gloria Dhandapani
University of Glasgow
Glasgow, Scotland
2252533D@student.gla.ac.uk

Jamie Ferguson
University of Glasgow
Glasgow, Scotland
jamie.f.ferguson@glasgow.ac.uk

Euan Freeman
University of Glasgow
Glasgow, Scotland
euan.freeman@glasgow.ac.uk

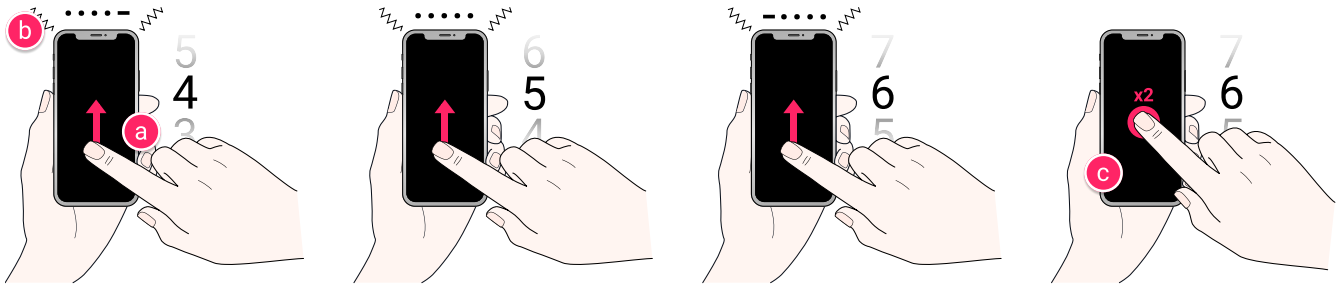

Figure 1: HapticLock uses non-visual interaction modalities for discreet eyes-free PIN entry. Users select PIN digits by swiping up or down (a), with Morse Code vibration patterns (b) for feedback about the currently selected digit. Users confirm selection with a double tap (c), to move to the next digit, continuing until the PIN is complete.

ABSTRACT

Smartphones provide access to increasing amounts of personal and sensitive information, yet are often only secured using methods that are prone to observational attacks. We present HapticLock, a novel authentication method for mobile devices that uses non-visual interaction modalities for discreet PIN entry that is difficult to attack by shoulder surfing. A usability experiment (N=20) finds effective PIN entry in secure conditions: e.g., in 23.5s with 98.3% success rate for a four-digit PIN entered from a random start digit. A shoulder surfing experiment (N=15) finds that HapticLock is highly resistant to observational attacks. Even when interaction is highly visible, attackers need to guess the first digit when PIN entry begins with a random number, yielding a very low success rate for shoulder surfing. Furthermore, a device can be hidden from view during authentication. Our use of haptic interaction modalities gives privacy-conscious mobile device users a usable and secure authentication alternative for sensitive situations.

CCS CONCEPTS

• **Human-centered computing** → **Interaction techniques**; • **Security and privacy** → *Usability in security and privacy*.

KEYWORDS

eyes-free authentication, haptic feedback, usable security

ACM Reference Format:
Gloria Dhandapani, Jamie Ferguson, and Euan Freeman. 2021. HapticLock: Eyes-Free Authentication for Mobile Devices. In *Proceedings of the 2021 International Conference on Multimodal Interaction (ICMI '21), October 18–22, 2021, Montréal, QC, Canada.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3462244.3481001

ICMI '21, October 18–22, 2021, Montréal, QC, Canada

1 INTRODUCTION

Smartphones contain increasing amounts of personal and sensitive data (e.g., photos, contact information, emails) and provide access to many services where security is crucial (e.g., bank accounts, mobile payment apps, apps with saved credit cards, etc). Protecting access to smartphones is therefore crucial, as they are a prime target for security attacks. Users secure their smartphones using a variety of authentication methods, including PINs, passwords and patterns entered using the touchscreen [21]. However, such methods are prone to observational attacks, where an attacker discovers how to gain access by viewing how a user authenticates via the screen.

Shoulder surfing (observing someone's information without their consent) can be a straightforward way of obtaining authentication secrets. Many smartphone users have experienced being shoulder surfed by strangers and have admitted to shoulder surfing others (e.g., on public transport) [19]. Authentication can be breached by simply watching or recording someone as they enter their PIN, password or lock pattern [8, 19, 23, 28]. More nuanced approaches, like smudge attacks [8, 9] and thermal attacks [1, 2], can even be used to infer a secret *after* the user has finished (e.g., using a thermal camera to identify residual heat from touchscreen PIN entry [1, 2]).

The risk of shoulder surfing has motivated significant research into alternative authentication schemes. One approach is to explore alternative interaction modalities, so that observing touch, alone, is insufficient to discover a PIN (e.g., through use of gaze [3, 25], gestures [3, 7, 35], force [26]). Another approach is to allow authentication without looking at the screen [10, 13, 16, 30, 32], so that users can obscure the device during PIN or password entry (e.g., keeping the device in their pocket or hidden under the table).

In this work, we describe HapticLock, an eyes-free smartphone authentication method that enables users to enter a PIN without looking at the device. Users enter their PIN by swiping up or down on the touchscreen to select digits from 0–9, with the currently selected digit played back using a Morse Code vibration pattern. These interaction modalities allow eyes-free authentication; users can discreetly and subtly enter their PIN with their device by their side, in a pocket, under the table, etc. We use random start digits, so that if the smartphone is not hidden from view, observers cannot simply count the number of swipes to decipher a PIN. Repetitive vertical swipes for input are also resistant to thermal and smudge attacks because their traces will overlap on the touchscreen [1].

We present two studies investigating the usability and security of HapticLock. Our findings show that HapticLock enables effective non-visual authentication and is robust against observation attacks. Our contributions include: (1) a novel smartphone authentication method using eyes-free interaction modalities; (2) an experiment investigating its usability and performance; and (3) an experiment investigating its robustness against observation attacks.

2 BACKGROUND

2.1 Observational Attacks

Shoulder surfing refers to viewing someone's information without their consent, e.g., by glancing at their smartphone screen. Widely used smartphone authentication methods (e.g., PINs, passwords, lock patterns) are susceptible to observational attack by shoulder surfing, because of the visible coupling between touchscreen input and output. An observer simply needs to view a person unlocking their device to discover the PIN/password/pattern/etc. Kwon et al. [28] described a variety of methods that are used by shoulder surfers to more adeptly detect authentication, revealing that simple observational attacks can yield a high success rate. These could potentially occur surprisingly often; Eiband et al. [19] found a significant proportion of people were aware of being shoulder surfed, or have shoulder surfed others (though not with malicious intent).

Observational attacks may occur after the time of authentication. Video recordings can be used later to observe user input and touching the screen itself leaves residual clues for attackers. For example, smudge attacks use fingerprint smudges left on screens [8, 9] and thermal attacks use residual heat transferred from fingers to the screen [1, 2], both with surprising efficacy. Such attacks are likely to become more prevalent, with increasingly miniature and cheaper recording devices [2] and the increased use of smartphones to access and store sensitive information, make payments, etc.

These risks have naturally inspired a significant body of research into methods for combating shoulder surfing. Work in this area takes many approaches, including attack prevention and attack detection (i.e., detecting shoulder surfing as it occurs). Attacks can potentially be prevented by obfuscating the screen [4], such that an attacker cannot make sense of the user's actions. Alternative interaction modalities have also been used to try fool attackers, e.g., by incorporating other input modalities like back-of-device touch [17, 37], gaze [3, 25], pressure input [26], mid-air gestures [7] and touchscreen gestures [35, 39]. These aim to make shoulder surfing more difficult, through use of additional input sensing.

Recent work has investigated the use of additional sensors to detect shoulder surfing as it occurs, e.g., using gaze estimation to detect bystanders glancing at the screen [34]. Detecting shoulder surfers raises concerns about wrongly implicating bystanders and infringing privacy through gaze tracking [24]. Camera-based attacks would not go detected by such approaches and will become increasingly challenging as cameras become more discreet.

In this work, we investigate a novel authentication method that aims to reduce the efficacy of observational attacks, both at the time of authentication and through recording. Whilst HapticLock uses the screen for input, it does not show any information and all communication with the user takes place through the haptic modality. Swipe gestures will overlap on the touchscreen, mitigating smudge and thermal attacks, and each authentication attempt starts with a random digit, aiming to make counting difficult. Furthermore, users can perform authentication with the device out of sight, e.g., in a pocket or under a table.

2.2 Eyes-Free Authentication

Eyes-free authentication methods have been developed to allow inconspicuous and discreet authentication. These use interaction modalities where input actions are not coupled with visual cues on screen, making it difficult for observers to discover how to authenticate. For example, users could provide input using back-of-device touch sensing [17, 37] where finger movements are occluded, or using pressure input [26] where the amount of pressure being applied cannot be deciphered. Another approach is to infer the user's identity through input characteristics; e.g., Nguyen et al. [32] used stroke gestures for invisible PIN input, analysing the drawing behaviour to help authenticate the user. Others have investigated temporal passwords, e.g., Lin et al. [29] and Nguyen et al. [31] used rhythmic tapping for input, where users tap their "tapword" [29].

Non-visual output modalities also support eyes-free authentication, e.g., using audio through headphones or vibration through a handheld device. For example, BlindPass [16], Colorlock and Timelock [13], Phone Lock [10], Secure Haptic Keypad [11], SpinLock [12], Tactile Authentication System [27], and Vibrapass [18] all used audio or vibration to discreetly present non-visual signals to the user during authentication. These would only be perceived by the user, making it more difficult for an observational attack.

HapticLock uses vibration for non-visual output; similar to these works, its vibration patterns will only be perceived by the user, making it difficult for an observer to decipher a PIN. Our work builds on similar ideas to BlindPass [16], which used four touchscreen gesture modes with audio feedback for eyes-free PIN entry. Three of their gesture modes are at risk of smudge and thermal attacks, since the touchscreen taps and strokes can be connected to the digits in a PIN; the fourth mode, 'Scroll', is more resistant to these attacks due to its overlapping touchscreen swipes [1]. When using the Scroll mode, users swiped up/down to increase/decrease a PIN digit, receiving audio playback of the entered digits. We use the same swipe gestures for PIN digit entry in HapticLock, although with vibrotactile output instead of audio. This has the advantage of not requiring headphones for secure PIN entry, making it more readily available. It is also appropriate for other mobile devices where vibration is a common output modality (e.g., smartwatches).

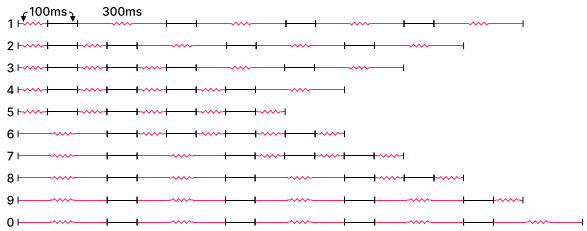

Figure 2: Morse Code vibration patterns for 0–9; dashes are 300ms, dots and pauses are 100ms.

2.3 Vibrotactile Number Encoding

Vibration can encode PIN digits in a variety of ways [20]. Bianchi et al. [13] used vibration counting, where vibration quantity encodes a PIN digit (e.g., four vibrations for the digit four). ActiVibe [14, 15] used a combination of vibration counting and duration, counting in multiples of five with a sequence of short vibrations for 1–4 and a longer vibration for 5 (inspired by Roman numerals). Users can achieve good recognition performance with such approaches, especially when a single digit is presented at a time (like with PIN entry). However, a concern from a security perspective is the linear relationship between vibration pattern duration and digit value. Observational attackers could potentially infer digits from the interval between input actions whilst perceiving a vibration.

In this work, we use Morse Code to encode digits instead. Morse Code encodes digits from 0–9 using combinations of five short ('dots') and long ('dash') signals, in our case using short and long vibrations, as shown in Figure 2. These vibration patterns are easy to learn and interpret [33, 36, 38]. From a security perspective, Morse Code may be more robust against observation attacks because of similarity between pattern duration. All digits have five vibrations, and several digits have the same pattern symbols in different orders, e.g., digits 3 and 7 both have three 'dots' and two 'dashes', so digits are more difficult to infer if a user pauses to perceive a pattern.

3 HAPTICLOCK

We propose HapticLock, an eyes-free authentication method for mobile devices. HapticLock allows users to discreetly enter their PIN by swiping up/down to increase/decrease the current digit, respectively, with Morse Code vibration patterns given as feedback.

Consider a user entering the PIN '1357'. First, the user swipes until they reach the digit 1 (Figure 3a); a double-tap gesture confirms this digit to move to the next selection (Figure 3b). Next, the user swipes up twice (to the digit 3), confirming it with a double-tap. This continues until digits 5 and 7 are entered correctly, at which point authentication is complete. If a user makes a mistake, they can use a two-finger tap to remove the most recent digit (Figure 3c). If the user is unsure how many PIN digits have been entered, they can long press to have the number played back as vibration (Figure 3d). This simple gesture vocabulary can be performed discreetly and nothing is shown on screen, as all feedback is given through vibration.

We use 300ms and 100ms vibrations for Morse Code 'dashes' and 'dots', respectively (same as Seim et al. [36]). A 100ms pause (the 'dot' duration) is inserted between signals. Figure 2 shows the vibration patterns for each digit. Each vibration pattern begins

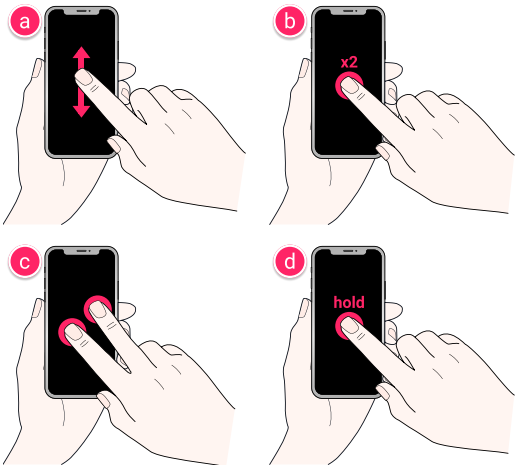

Figure 3: HapticLock touchscreen gestures: (a) swipe up or down to increase or decrease digit, respectively; (b) double tap to confirm digit; (c) two-finger tap to remove most recent digit; (d) long-press to check how many digits are entered.

playing immediately when the user swipes, interrupting any previously active vibration. This was implemented in Android using the VibrationEffect API [5]. We used two additional vibration patterns to give other feedback. When the user performs a two-finger tap to remove the most recent digit, a short 100ms vibration is given as confirmation. When the user has entered an incorrect PIN, a long 600ms vibration is given to indicate an error. These patterns allow all interaction to take place eyes-free and use a single vibration so are distinct from the Morse Code.

Each authentication attempt starts from a random digit, making it more difficult for attackers to discover a PIN by counting swipes. Users can continue swiping in the same direction to loop through the numbers (e.g., 8→9→0→1→2), so there are no changes in direction corresponding to the end digits. HapticLock uses eyes-free input and output modalities which allow users to obscure their device during PIN entry for further security (e.g., in a pocket, in a bag, under a table). It is also robust against thermal and smudge attacks because swipes can be performed anywhere on the screen and will overlap, ensuring that smudges and heat traces are distorted. Furthermore, overlapping swipe interactions have been found to be more robust against thermal attacks than overlapping tapping interactions [1]. Finally, our use of Morse Code makes it difficult to infer digits from the time spent receiving the vibration patterns.

4 USABILITY EXPERIMENT

4.1 Experiment Design

We conducted an experiment to investigate the usability and PIN entry performance of HapticLock. We evaluated HapticLock on its own rather, rather than compare to other PIN entry methods. HapticLock allows haptic-only PIN entry and is resistant to many observational attacks and so addresses a gap in the literature; whilst simpler existing methods may lead to faster PIN entry or better usability, they do not offer the same benefits or a fair comparison. We thus measure its performance in this context.

We used a within-subjects design with two factors: (1) PIN length (4 or 6 digits); and (2) start digit (random or zero), giving four conditions in total. We selected 4-digit and 6-digit PIN lengths as both are commonly used and allowed us to evaluate how longer PINs might affect usability and cognitive load. We evaluated PIN entry with each attempt starting at zero (as a baseline) and at a random digit (as the intended design). We evaluated the zero start digit to help assess the impact of using the Morse Code; i.e., when starting at zero, users could simply count swipes, but the random start digit requires use of vibration.

For each task, participants had to use HapticLock to enter a given PIN. We used a set of predefined PINs, created such that there were no duplicates and subsequent digits were at least two numbers apart. Tasks started when the user first touched the screen and ended when the PIN was entered correctly; if an incorrect PIN was entered, users had to try again and timing continued until the PIN was successfully entered. Tasks were presented in two blocks (random start digit, zero start digit), in counter-balanced order. There were 24 total tasks (six per condition).

We measured time taken per task and the number of incorrect attempts per task. After each block, participants completed the NASA-TLX survey [22]. At the end of the experiment, participants completed a survey asking about their use of HapticLock. We used ANOVA with post hoc t-tests to analyse task time as this data met the test assumptions (continuous data, normal distribution); we used Wilcoxon tests to analyse number of incorrect attempts and the TLX scores, which did not satisfy these assumptions.

We conducted this study remotely by deploying the experiment as an Android app. On the two days before the experiment, we asked participants to complete a game where they had to identify numbers through their Morse Code vibration pattern; this trained them in recognising these simple vibration patterns prior to the HapticLock experiment. The experiment itself also had training tasks that introduced the HapticLock interactions and allowed them to practice PIN entry. These training activities established a baseline competence in Morse Code and increased our confidence that participants were capable of using it effectively. Each experiment session lasted one hour, including training tasks and surveys. Our experiment app played white noise and we asked participants to use headphones during the experiment, to mask any vibration sounds.

We recruited using an email list for user study recruitment in Glasgow. Participants needed to be regular smartphone users who owned an Android 11 device. Twenty participants were recruited (11 female, 9 male, mean age 26.2 years, SD 6.4 years). None knew Morse Code prior to the study. They used PIN (14), password (12), fingerprint (9), face recognition (3) and lock patterns (2) on their own devices.

4.2 Results

4.2.1 PIN Entry Attempts. There were 28 incorrect PIN attempts (5.8% of tasks). Figure 4 shows total number of errors per condition. Wilcoxon tests found no significant difference in incorrect attempts between start digits (zero vs random: Z = .98, p = .33) and PIN lengths (four vs six digits: Z = .77, p = .44).

4.2.2 PIN Entry Time. Mean task time was 26.5s (SD 11.1s, 95% CI [25.47s, 27.45s]), with the fastest PIN entry taking 8.9s. For fair comparison, we calculated task time-per-digit for analysis, since 6-digit PIN entry should take longer than 4-digit PIN entry because there are more digits. Figure 5 shows mean task time-per-digit for each condition; the 95% CI for mean time-per-digit is [5.12s, 5.45s].

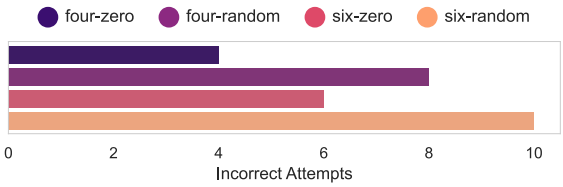

Figure 4: Total incorrect PIN attempts per condition.

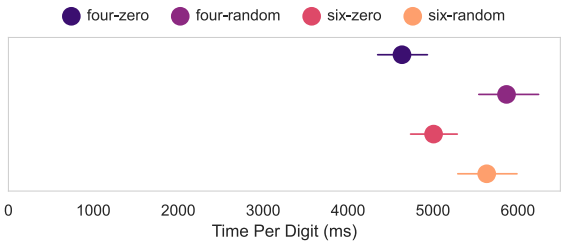

Figure 5: Mean task time normalised by number of digits. Error bars show 95% CI.

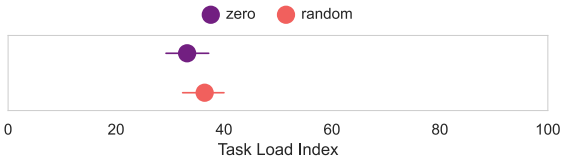

Figure 6: Mean TLX scores. Error bars show 95% CI.

A repeated-measures ANOVA found a significant main effect of start digit on time-per-digit: F(1, 19) = 22.2, p < .001, η_p^2 = .54. A post hoc t-test found significantly faster time for the zero start digit: t(19) = 4.71, p < .001, d = .78. There was no significant main effect of PIN length: F(1, 19) = .51, p = .48.

4.2.3 TLX Scores. Raw TLX score [22] was computed from the NASA-TLX surveys. Mean TLX score was 34.8 (SD 9.4). Figure 6 shows the mean score for each block of tasks (note: score was out of 100). A Wilcoxon test did not find a significant difference between TLX scores for the 'zero' and 'random' start digit task blocks: Z = 1.78, p = .08.

4.2.4 Survey. We asked participants to rate agreement with three statements using a 7-point Likert scale: Q1 I found the HapticLock interface easy to use; Q2 I found the vibration patterns easy to understand; and Q3 I found it easy to orient myself in the random start digit tasks. Figure 7 shows the scale ratings.

Responses show agreement with the statements in Q1 and Q2 (median 6.0 and 5.5, respectively). There was less consensus for Q3 (median 4.5), although half of participants showed agreement.

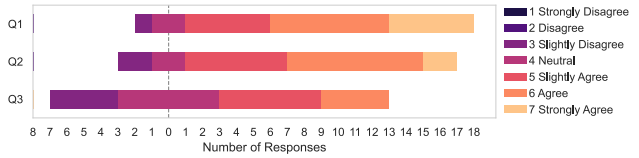

Figure 7: Likert scale ratings for survey questions (aligned around the neutral response at $x = 0$).

We asked participants to describe how they used HapticLock. Most described keeping count in their head as they swiped up/down. This was easy when digits started at zero, but less straightforward for the random start (*"the hard part was knowing where you started"*). Several responses said they searched for a *"reference point"*, most often zero or five (*"those were the easiest ones to find because they had the same vibrations"*).

We asked participants to describe how they used and perceived the vibration patterns, since all were novice users who learned the Morse Code digits for this experiment. These responses reveal strategies used by unfamiliar users. Some focused on either dashes or dots: *"long vibrations ... were more noticeable"*, *"dots were easier to notice"*. One focused on whichever came first: *"I didn't need to wait for the whole pattern because I could just count the number of dots or dashes"*. Another used the relative difference to identify symbols: *"you just need to feel the difference between fast and slow"*.

Seven responses said they mostly ignored the Morse Code once they knew which digit they had selected (most relevant for the random start): e.g., *"I just ignored most of them once I knew my start point"* and *"after you know you have the first number, you can just ignore them when you count in your head"*. However, several said they still used those vibration patterns as feedback about the swipe gestures: e.g., *"when you felt it vibrate you know you did the right thing because it was kind of confirming your swipe"* and *"I ignored the actual numbers but used the vibration to know it was responding"*.

Vibration was useful for helping users keep track of which digit was selected: e.g., *"I used them to check I had the right number"* and *"a few times I got lost or missed my swipe so I had to figure out what number I was at using the vibrations"*. One user described a strategy where they only waited for the final pattern, as the most relevant: *"if I had to go over lots of numbers at once I would skip the vibration and only feel for the last one - that's the only one that matters anyway"*.

4.3 Discussion

We compared HapticLock's random start digit with a zero start digit as a baseline for PIN entry. Starting at zero was indeed faster (mean 24.3s vs 28.6s), although this was expected: this difference is the time cost of using the vibrotactile feedback to find a known start digit. Whilst there was an associated time cost with locating a known start digit from random, we did not observe significant increase in cognitive demand via the TLX responses. There were different strategies for finding that starting point. Some used the Morse Code vibrations to recognise the currently selected digit, whilst others swiped until they found a reference digit – typically zero or five, as they consisted of all long, or short, vibrations, respectively. With more experience, we would expect users to develop their own strategies for efficient PIN entry, just like our participants did within the duration of this study.

Once users knew which digit was currently selected, they kept count in their head as they swiped between digits. This prompted a change in how they used the vibrotactile feedback. Many started to ignored the Morse Code itself after the first digit, using subsequent vibrations as simple confirmation that their swipe gestures were recognised. This allows faster input because skipping the complete pattern and swiping was quicker than waiting for a full vibration (ranging from 900ms for '5' to 1900ms for '0'). We believe these strategies contribute to the lack of significant workload increase.

Arguably the interaction could be simplified by giving a single confirmatory vibration for each swipe, after the first digit has been entered. Alternatively, no vibration could be given after the first digit so users (whom often ignored subsequent vibrations) can skip through digits quickly. However, users would often rely on the vibration if they lost count or were unsure if a swipe gesture was recognised, so there is value in continuing to deliver the full pattern. Some would also pause at their reference digit (e.g., five) when moving past, just to reassure themselves they were keeping track correctly. Alternative haptic designs like these are a compelling topic for future work as this could improve HapticLock's usability.

A longer PIN will naturally take longer to enter via HapticLock, just like a longer password will take more time to type. There is a time and workload cost associated with finding the first digit from a random start, although this only happens once per authentication attempt; thus, adding more digits will, indeed, increase the overall time, but this is the simplest part of the interaction, so we believe this would scale well to longer secrets if desired.

Our findings also give insight into the general use of vibrotactile Morse Code as a means of encoding digits in vibration patterns. All of our participants were novice Morse Code users, who learned the digits in the days before the experiment. When using HapticLock, they did not need to recognise digits spontaneously. Context (i.e., prior digits) and relative change (i.e., increasing number of short vibrations going from 1–5) simplified pattern recognition. This led to effective interaction here and suggests the potential ease of using vibrotactile Morse Code in other application contexts.

Our intention is not for HapticLock to be the primary method of authenticating with a mobile device, as it is slower than widely used methods like visual PIN and password entry. However, it offers a usable and robust alternative for privacy-conscious users in scenarios where shoulder surfing is prevalent [19]. Its interaction time is modest when weighed against privacy concerns (e.g., when accessing sensitive information or authenticating around others); a four-digit PIN starting at a random start digit took novice users 23.5s and the first quartile time of 18.6s shows potential for faster interaction. Long-term use of HapticLock would be supported by the interaction strategies reported by our participants to simplify their use of Morse Code vibration and facilitate efficient PIN entry; a tutorial could teach these strategies to support rapid adoption.

5 SHOULDER SURFING EXPERIMENT

5.1 Experiment Design

We conducted an experiment to assess how robust HapticLock was against observational attacks. We used a within-subjects design

with two factors: attacker view (unobscured, obscured) and start digit (random or zero), giving four conditions. We used four-digit PINs in this experiment. Our threat model assumes the attacker knows our authentication method and knows the starting point (zero or random), but does not know the PIN. We considered video based shoulder surfing attacks from two attacker views, which are each feasible with our eyes-free interaction design: **Unobscured:** an attacker has a clear over-the-shoulder view of the smartphone screen; **Obscured:** an attacker has a side-on view of the user holding their smartphone in a jacket pocket.

For each task, participants (the attackers) viewed a video of a user entering a PIN on a smartphone using HapticLock. They were free to watch the videos an unlimited number of times and could pause, zoom, and adjust playback speed and volume. Their task was to guess the PIN, with three guesses allowed. We allowed three guesses as this increases the likelihood of successful attack, e.g., by allowing multiple guesses of the first digit. Many PIN systems also lock out after three failed attempts and similar studies also allowed three attempts [25]. There were 12 videos (three per condition), each showing a different PIN being entered.

We measured successful attacks and the Levenshtein distance between the PIN and the closest guess from the three attempts (as in [25]). The Levenshtein distance indicates PIN similarity, where smaller distances indicate greater similarity. Since the first digit is essential for guessing subsequent digits, we also recorded if the first digit was correctly identified. We used Wilcoxon tests to analyse these metrics, as this non-parametric test is appropriate for these data types. After the experiment, participants completed a survey asking about their attack strategy.

We conducted this study using an online survey platform with participants recruited via a local mailing list. Fifteen participants were recruited from the UK (mean age 23.2 years, SD 1.7 years). None participated in the usability experiment.

5.2 Results

A total of 180 attacks were completed. Only 14 were successful (8%): 13 for unobscured view with start digit zero (29% for this condition), and one for unobscured view with a random start digit (2% for this condition). No attacks on the obscured view were successful.

The first digit was guessed correctly in 67 attacks (37%): 41 for the unobscured view with start digit zero (91% for this condition), 13 for the unobscured view with a random start digit (29% for this condition), 6 for the obscured view with zero start digit (13% for this condition) and 7 for the obscured view with random start digit (16% for this condition). Wilcoxon tests found significantly more correct first digits with start digit zero ($Z = 5.53$, $p < .001$), and for the unobscured view ($Z = 4.02$, $p < .002$).

Mean Levenshtein distance was 2.79 (SD 1.14), as seen in Figure 8. Wilcoxon tests found significantly lower distance for the zero start digit (2.37 vs 3.22, $Z = 4.52$, $p < .001$) and for the unobscured video view (2.48 vs 3.11, $Z = 3.65$, $p < .001$); in this case, lower Levenshtein distance suggests less secure, as guesses were closer to the real PIN.

5.3 Discussion

This experiment showed that eyes-free PIN entry is still susceptible to observational attack. However, it also showed that security can be increased significantly by using a random start digit (as intended for HapticLock) or obscuring the device (as enabled by non-visual interaction modalities). Whilst it might seem facetious to attempt an observational attack in these conditions, no authentication method is perfect and there are still cues that an attacker could exploit.

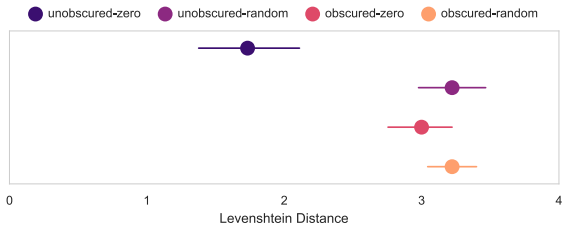

Figure 8: Levenshtein distance between target PIN and best guess from the three attempts. Smaller numbers mean greater similarity. Error bars show 95% CI.

The most common attack strategy described in the survey was observing finger swipes and internally counting the entered digit. It was not surprising that attacks were more successful with zero as the starting digit, since counting will correctly reveal the first digit. Only a small proportion of those first digits were converted into successful attacks, however. When the starting digit was random, participants were—in their own words—guessing where to start.

We believe the success rate was not higher for the {unobscured, zero start digit} condition because it was not always obvious to the attackers when the user had double-tapped to move to the next digit, causing subsequent digits to be guessed incorrectly. Some said they also looked for a change in direction to indicate moving to the next digit; however, this was not reliable, since not all digit sequences had a change in direction and the user could loop digits (e.g., from 9→0→1 by swiping up, or 1→0→9 by swiping down).

Similar strategies were used for the obscured view conditions, where the video showed a side-on view of the user's hand in their pocket. Although the hand was not directly visible, hand motions were subtly noticeable and attackers tried to interpret these cues. These attempts were unsuccessful, with 10 correct first digits but no correct PIN guesses. These cues are likely difficult to interpret as it would be difficult to differentiate between a swipe (to change digit) and a double tap (to confirm digit).

A small number of participants reported attempting to listen to the vibration patterns, especially in the obscured conditions where vibrations against the clothing material were sometimes audible with the volume increased. These cues were not clear enough to decipher the Morse Code patterns, however, since body movements resulted in barely audible cues that masked the vibration sounds. Moreover, the user did not need to pause to check vibrations (i.e., skipping over digits), interrupting the vibration and making it more difficult to try and decipher audible cues.

6 OVERALL DISCUSSION

In this paper we presented HapticLock, a novel authentication method for mobile devices. HapticLock was designed to be resilient to observational attacks through its use of non-visual interaction modalities. Users enter their PIN via touchscreen swipes and tap

gestures, with Morse Code vibrations for the currently selected PIN digit. Each attempt begins with a random digit and all feedback is via vibration. This makes it difficult for observers to recognise a PIN by watching the user and allows PIN entry to an obscured device (e.g., in a pocket or bag, under a table, held against the torso).

6.1 Usability

Our evaluation found good PIN entry performance, considering this is an unfamiliar haptic-only PIN method. Whilst not as straightforward as daily authentication methods (e.g., lock patterns, PIN keypads), HapticLock yields good PIN entry success. Most users developed their own strategies for efficient interaction; for example, finding zero or five as an easily recognised reference digit, then internally counting as they swiped to select digits. The Morse Code vibration patterns served many purposes: helping users locate the first digit, providing implicit feedback that a swipe was recognised, and helping users orient themselves if they lost count or were unsure if a gesture was recognised. Users similarly adopted methods for easier Morse Code perception; for example, counting the 'dots' at the beginning or end of a pattern, or focusing on the relative change between subsequent patterns. This extends the small body of work investigating vibrotactile Morse Code and shows the potential for easy use in other application contexts.

6.2 Resilience to Shoulder Surfing

Much like users counted their swipes internally, so too did attackers in our shoulder surfing experiment. When the device was visible and PIN entry started at zero, most attackers correctly identified the first PIN digit and all but one successful attacks happened in this condition. When PIN entry started at a random digit, attackers had to guess the start digit, leading to low success rate. When the device was obscured in a pocket, attackers tried to interpret subtle motion cues; this was unsuccessful, as it was difficult to differentiate between swipes, taps and other gestures on the screen.

In the shoulder surfing experiment, some participants described attempted audio attacks. These participants increased volume to listen for audible cues, e.g., as a result of clothing moving whilst the user entered a PIN with their device in a pocket. These attacks were not successful, because the start digit was not known, but an interesting question for future work is whether an auditory attack would be possible if vibration sounds could be detected. This novel form of observation attack (via listening) shows that novel authentication methods may lead to unexpected attack strategies, which require mitigation. A partial solution to this could be to reduce vibration intensity such that audible side effects are reduced.

We used four-digit and six-digit PINs in our usability experiment and four-digit PINs in our shoulder surfing experiment. Four and six digits are common on mobile devices (e.g., six digits is the default on iOS) and both were usable. There was a time cost associated with the random start digit, although this only happens once per authentication attempt; HapticLock could also be used with longer PINs, facilitated by the interaction strategies adopted by our users.

6.3 Intended Use

HapticLock is slower than normal PIN entry via touchscreen keyboard, so it is understandable that people would not want to use it every time they unlock their device. Indeed, the associated time cost is a limitation of this method which makes it unsuitable for high frequency usage (e.g., each time a smartphone needs unlocked). Our intention was to explore a secure alternative for privacy-conscious users who are accessing sensitive information, for infrequent but high-risk transactions (e.g., payment, cash withdrawal), or authenticating in the presence of others (e.g., on public transport and in other situations where shoulder surfing is prevalent [19], or when thermal [2] and smudge attacks [9] may be a concern). The benefits of eyes-free PIN entry are a worthy trade-off in such scenarios.

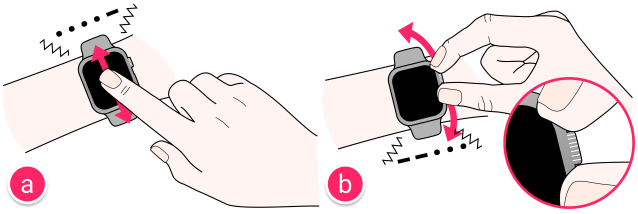

Figure 9: HapticLock could also be used for wearable devices, e.g., via a touchscreen (a) or watch crown (b).

Our authentication method is not just limited to smartphones. HapticLock could also be used by other wearable devices, e.g., like in Figure 9. Input sensing needs are simple: at minimum, detecting bidirectional input and a confirmation action. This is not limited to touchscreens; other input modalities like a smartwatch crown dial could be used for digit selection (as in Figure 9 (b)), robust against shoulder surfing as the meaning of crown adjustments would be unclear. All output is given via vibration patterns and vibrotactile actuators, which are common in most smartwatches, fitness trackers, etc. Smart rings could similarly be used for discreet PIN entry, e.g., selecting digits by using the thumb to twist a ring around the finger (like in work by Ashbrook et al. [6]).

Evaluating HapticLock on wearable devices is a compelling area for future research. Wearable devices are increasingly being used to access sensitive information (e.g., watch payment platforms like Apple Pay and Garmin Pay) so protecing access is important. These devices could also enable HapticLock to be used for PIN entry on other devices: e.g., to unlock laptops and workstations, or for PIN entry at ATMs and payment terminals.

7 CONCLUSION

We presented HapticLock, an authentication method for mobile devices that uses eyes-free interaction modalities for discreet PIN entry. We described two experiments investigating its usability (N=20) and resistance to shoulder surfing (N=15). Our findings show the potential of eyes-free PIN entry: HapticLock was both usable and secure against observation. We also gained insight into how users used HapticLock, uncovering efficient interaction strategies and their methods for making sense of vibrotactile Morse Code. Our authentication method is not intended to replace conventional PIN and password entry, but to give users a secure alternative when unlocking a device around other people or when concerned about being shoulder surfed—which are becoming increasingly important as mobile devices are used to access increasing amounts of personal and sensitive information.

REFERENCES

[1] Yomna Abdelrahman, Mohamed Khamis, Stefan Schneegass, and Florian Alt. 2017. Stay Cool! Understanding Thermal Attacks on Mobile-based User Authentication. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. ACM. https://doi.org/10.1145/3025453.3025461

[2] Yasmeen Abdrabou, Yomna Abdelrahman, Ahmed Ayman, Amr Elmougy, and Mohamed Khamis. 2020. Are Thermal Attacks Ubiquitous? When Non-Expert Attackers Use Off the Shelf Thermal Cameras. In *Proceedings of the International Conference on Advanced Visual Interfaces (AVI '20)*. ACM, Article 47, 5 pages. https://doi.org/10.1145/3399715.3399819

[3] Yasmeen Abdrabou, Mohamed Khamis, Rana Mohamed Eisa, Sherif Ismail, and Amrl Elmougy. 2019. Just gaze and wave: exploring the use of gaze and gestures for shoulder-surfing resilient authentication. In *Proceedings of the 11th ACM Symposium on Eye Tracking Research & Applications (ETRA '19)*. ACM. https://doi.org/10.1145/3314111.3319837

[4] Suliman A. Alsuhibany. 2020. Usability and shoulder surfing vulnerability of pattern passwords on mobile devices using camouflage patterns. *Journal of Ambient Intelligence and Humanized Computing* 11 (2020). https://doi.org/10.1007/s12652-019-01269-3

[5] Android. 2021. VibrationEffect API Reference. https://developer.android.com/reference/android/os/VibrationEffect

[6] Daniel Ashbrook, Patrick Baudisch, and Sean White. 2011. Nenya: Subtle and Eyes-Free Mobile Input with a Magnetically-Tracked Finger Ring. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11)*. ACM, 2043–2046. https://doi.org/10.1145/1978942.1979238

[7] Md Tanvir Islam Aumi and Sven Kratz. 2014. AirAuth: evaluating in-air hand gestures for authentication. In *Proceedings of the 16th international conference on Human-computer interaction with mobile devices & services (MobileHCI '14)*. ACM. https://doi.org/10.1145/2628363.2628388

[8] Adam J. Aviv, John T. Davin, Flynn Wolf, and Ravi Kuber. 2017. Towards Baselines for Shoulder Surfing on Mobile Authentication. In *Proceedings of the 33rd Annual Computer Security Applications Conference (ACSAC 2017)*. ACM, 486–498. https://doi.org/10.1145/3134600.3134609

[9] Adam J. Aviv, Katherine Gibson, Evan Mossop, Matt Blaze, and Jonathan M. Smith. 2010. Smudge attacks on smartphone touch screens. In *Proceedings of the 4th USENIX conference on Offensive technologies (WOOT'10)*. USENIX Association.

[10] Andrea Bianchi, Ian Oakley, Vassilis Kostakos, and Dong Soo Kwon. 2010. The phone lock: audio and haptic shoulder-surfing resistant PIN entry methods for mobile devices. In *Proceedings of the fifth international conference on Tangible, embedded, and embodied interaction (TEI '11)*. ACM, 197––200. https://doi.org/10.1145/1935701.1935740

[11] Andrea Bianchi, Ian Oakley, and Dong Soo Kwon. 2010. The Secure Haptic Keypad: A Tactile Password System. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '10)*. ACM, 1089–1092. https://doi.org/10.1145/1753326.1753488

[12] Andrea Bianchi, Ian Oakley, and Dong Soo Kwon. 2011. Spinlock: A Single-Cue Haptic and Audio PIN Input Technique for Authentication. In *Proceedings of the International Workshop on Haptic and Audio Interaction Design (HAID '11)*. Springer Berlin Heidelberg, 81–90.

[13] Andrea Bianchi, Ian Oakley, and Dong Soo Kwon. 2012. Counting clicks and beeps: Exploring numerosity based haptic and audio PIN entry. *Interacting with Computers* 24 (2012). Issue 5. https://doi.org/10.1016/j.intcom.2012.06.005

[14] Jeffrey R. Blum and Jeremy R. Cooperstock. 2019. Single-Actuator Vibrotactile Numeric Information Delivery in the Face of Distraction. In *Proceedings of the 2019 IEEE World Haptics Conference (WHC '19)*. IEEE, 461–466. https://doi.org/10.1109/WHC.2019.8816082

[15] Jessica R. Cauchard, Janette L. Cheng, Thomas Pietrzak, and James A. Landay. 2016. ActiVibe: Design and Evaluation of Vibrations for Progress Monitoring. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, 3261–3271. https://doi.org/10.1145/2858036.2858046

[16] Chen Chen, Soon Hau Chua, David Chung, Simon T. Perrault, Shengdong Zhao, and Wing Kei. 2014. Eyes-free gesture passwords: a comparison of various eyes-free input methods. In *Proceedings of the Second International Symposium of Chinese CHI (Chinese CHI '14)*. ACM. https://doi.org/10.1145/2592235.2592248

[17] Alexander De Luca, Marian Harbach, Emanuel von Zezschwitz, Max-Emanuel Maurer, Bernhard Ewald Slawik, Heinrich Hussmann, and Matthew Smith. 2014. Now You See Me, Now You Don't: Protecting Smartphone Authentication from Shoulder Surfers. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2937–2946. https://doi.org/10.1145/2556288.2557097

[18] Alexander De Luca, Emanuel von Zezschwitz, and Heinrich Hußmann. 2009. Vibrapass: Secure Authentication Based on Shared Lies. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '09)*. ACM, 913–916. https://doi.org/10.1145/1518701.1518840

[19] Malin Eiband, Mohamed Khamis, Emanuel von Zezschwitz, Heinrich Hussmann, and Florian Alt. 2017. Understanding Shoulder Surfing in the Wild: Stories from Users and Observers. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. ACM. https://doi.org/10.1145/3025453.3025636

[20] Euan Freeman, Graham Wilson, Dong-Bach Vo, Alex Ng, Ioannis Politis, and Stephen Brewster. 2017. *Multimodal Feedback in HCI: Haptics, Non-Speech Audio, and Their Applications*. ACM and Morgan & Claypool, 277–317. https://doi.org/10.1145/3015783.3015792

[21] Marian Harbach, Alexander De Luca, and Serge Egelman. 2016. The Anatomy of Smartphone Unlocking: A Field Study of Android Lock Screens. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, 4806–4817. https://doi.org/10.1145/2858036.2858267

[22] Sandra G. Hart and Lowell E. Staveland. 1988. Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research. In *Human Mental Workload*, Peter A. Hancock and Najmedin Meshkati (Eds.). Advances in Psychology, Vol. 52. North-Holland, 139–183. https://doi.org/10.1016/S0166-4115(08)62386-9

[23] Tahir Musa Ibrahim, Shafi'i Muhammad Abdulhamid, Ala Abdusalam Alarood, Haruna Chiroma, Mohammed Ali Al-garadi, Nadim Rana, Amina Nuhu Muhammad, Adamu Abubakar, Khalid Haruna, and Lubna A. Gabralla. 2019. Recent advances in mobile touch screen security authentication methods: A systematic literature review. *Computers & Security* 85 (2019).

[24] Christina Katsini, Yasmeen Abdrabou, George E. Raptis, Mohamed Khamis, and Florian Alt. 2020. The Role of Eye Gaze in Security and Privacy Applications: Survey and Future HCI Research Directions. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI '20)*. ACM. https://doi.org/10.1145/3313831.3376840

[25] Mohamed Khamis, Mariam Hassib, Emanuel von Zezschwitz, Andreas Bulling, and Florian Alt. 2017. GazeTouchPIN: protecting sensitive data on mobile devices using secure multimodal authentication. In *Proceedings of the 19th ACM International Conference on Multimodal Interaction (ICMI '17)*. ACM. https://doi.org/10.1145/3136755.3136809

[26] Katharina Krombholz, Thomas Hupperich, and Thorsten Holz. 2016. Use the Force: Evaluating Force-Sensitive Authentication for Mobile Devices. In *Proceedings of the Twelfth USENIX Conference on Usable Privacy and Security (SOUPS '16)*. USA, 207–219.

[27] Ravi Kuber and Wai Yu. 2010. Feasibility study of tactile-based authentication. *International Journal of Human-Computer Studies* 68 (2010).

[28] T. Kwon, S. Shin, and S. Na. 2014. Covert Attentional Shoulder Surfing: Human Adversaries Are More Powerful Than Expected. *IEEE Trans. on Systems, Man, and Cybernetics: Systems* 44 (2014). https://doi.org/10.1109/TSMC.2013.2270227

[29] Felix Xiaozhu Lin, Daniel Ashbrook, and Sean White. 2011. RhythmLink: Securely Pairing I/O-Constrained Devices by Tapping. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology (UIST '11)*. ACM, 263–272. https://doi.org/10.1145/2047196.2047231

[30] Diogo Marques, Tiago Guerreiro, Luís Duarte, and Luís Carriço. 2013. Under the Table: Tap Authentication for Smartphones. In *Proceedings of the 27th International BCS Human Computer Interaction Conference (BCS-HCI '13)*. BCS Learning & Development Ltd., Article 33, 6 pages.

[31] Toan Nguyen and Nasir Memon. 2018. Tap-based user authentication for smartwatches. *Computers & Security* 78 (2018), 174–186. https://doi.org/10.1016/j.cose.2018.07.001

[32] Toan Van Nguyen, Napa Sae-Bae, and Nasir Memon. 2017. DRAW-A-PIN: Authentication using finger-drawn PIN on touch devices. *Computers & Security* 66 (2017). https://doi.org/10.1016/j.cose.2017.01.008

[33] Myrthe A. Plaisier, Daphne S. Vermeer, and Astrid M. L. Kappers. 2020. Learning the Vibrotactile Morse Code Alphabet. *ACM Transactions on Applied Perception* 17, 3 (2020). https://doi.org/10.1145/3402935

[34] Alia Saad, Dina Hisham Elkafrawy, Slim Abdennadher, and Stefan Schneegass. 2020. Are They Actually Looking? Identifying Smartphones Shoulder Surfing Through Gaze Estimation. In *ACM Symposium on Eye Tracking Research and Applications (ETRA '20 Adjunct)*. ACM. https://doi.org/10.1145/3379157.3391422

[35] N. Sae-Bae, N. Memon, K. Isbister, and K. Ahmed. 2014. Multitouch Gesture-Based Authentication. *IEEE Transactions on Information Forensics and Security* 9, 4 (2014). https://doi.org/10.1109/TIFS.2014.2302582

[36] Caitlyn Seim, Rodrigo Pontes, Sanjana Kadiveti, Zaeem Adamjee, Annette Cochran, Timothy Aveni, Peter Presti, and Thad Starner. 2018. Towards Haptic Learning on a Smartwatch. In *Proceedings of the 2018 ACM International Symposium on Wearable Computers (ISWC '18)*. ACM, 228–229. https://doi.org/10.1145/3267242.3267269

[37] Shaikh Shawon Arefin Shimon, Sarah Morrison-Smith, Noah John, Ghazal Fahimi, and Jaime Ruiz. 2015. Exploring User-Defined Back-Of-Device Gestures for Mobile Devices. In *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '15)*. ACM, 227–232. https://doi.org/10.1145/2785830.2785890

[38] Hong Tan, N Durlach, W Rabinowitz, Charlotte Reed, and J Santos. 1997. Reception of Morse code through motional, vibrotactile, and auditory stimulation. *Perception & Psychophysics* 59 (1997), 1004–1017. https://doi.org/10.3758/BF03205516

[39] Emanuel von Zezschwitz, Alexander De Luca, Bruno Brunkow, and Heinrich Hussmann. 2015. SwiPIN: Fast and Secure PIN-Entry on Smartphones. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*. ACM, 1403–1406. https://doi.org/10.1145/2702123.2702212