The Anatomy of Smartphone Unlocking

A Field Study of Android Lock Screens

Marian Harbach[1], **Alexander De Luca**[2], **Serge Egelman**[1,3]
[1]International Computer Science Institute, Berkeley, CA
[2]Google, Zurich, Switzerland
[3]University of California, Berkeley, CA
mharbach@icsi.berkeley.edu, adeluca@google.com, egelman@icsi.berkeley.edu

ABSTRACT
To prevent unauthorized parties from accessing data stored on their smartphones, users have the option of enabling a "lock screen" that requires a secret code (e.g., PIN, drawing a pattern, or biometric) to gain access to their devices. We present a detailed analysis of the smartphone locking mechanisms currently available to billions of smartphone users worldwide. Through a month-long field study, we logged events from a panel of users with instrumented smartphones ($N = 134$). We are able to show how existing lock screen mechanisms provide users with distinct tradeoffs between usability (unlocking speed vs. unlocking frequency) and security. We find that PIN users take longer to enter their codes, but commit fewer errors than pattern users, who unlock more frequently and are very prone to errors. Overall, PIN and pattern users spent the same amount of time unlocking their devices on average. Additionally, unlock performance seemed unaffected for users enabling the stealth mode for patterns. Based on our results, we identify areas where device locking mechanisms can be improved to result in fewer human errors – increasing usability – while also maintaining security.

Author Keywords
Smartphone; security; Android; lock screen; field study; usability

ACM Classification Keywords
H.5.2. Information Interfaces and Presentation: User InterfacesInput devices and strategies, evaluation

INTRODUCTION
Due to increased processing power and storage capabilities, modern smartphones store a plethora of sensitive data that users want to prevent others from accessing [10]. This, together with security and usability problems of current authentication systems [1, 2, 18, 22], has motivated research on improving smartphone authentication, including improving adoption. Researchers propose new locking mechanisms on a regular basis.

As we show in this paper, the research on these systems operates on assumptions about smartphone authentication that are overly optimistic or simply not true. For instance, a system that takes 10 seconds to authenticate might be fine in a laboratory setting, but its usability in the real world is questionable when a user has to unlock her phone up to 50 times per day. Thus, in order to improve the usability and adoption of smartphone authentication mechanisms, an understanding of users' *in situ* behaviors over long periods of time is necessary.

Recent research has taken steps to better understand smartphone locking in the wild, such as by identifying basic preferences [23], usage patterns and performance [12], and attitudes towards unlocking [10]. However, the performance data from these laboratory and field studies is still not very detailed and we are unaware of any study that has looked at exact authentication times and error rates in a real world setup. Harbach *et al.* performed the only prior field study of which we are aware [12], and their instrumentation did not allow for the capture of unlocking errors nor did they differentiate between preparation and actual unlocking time, which precluded a detailed usability analysis.

We performed a field study over one month with 134 participants of mixed demographics. We collected detailed data on unlocking, authentication speed, error counts, and types of errors. We provide detailed data on the influence of errors on overall authentication times, showing that, for example, PIN and pattern users spend similar amounts of time unlocking their devices, considering the higher number of errors committed by pattern users. We furthermore show the influence of wearing a watch on overall session count, providing proof that many device activations focus on simple activities, such as checking the time. Also, those pattern users that activated the "stealth mode" (visited cells are not highlighted) performed as well as their less-secure counterparts.

We argue that our work is complementary to the previously mentioned studies and fills an important gap left by these research efforts. The main contribution of this paper is providing a benchmark of current smartphone authentication mechanisms for future research to be compared against, as users are unlikely to invest more effort into unlocking their smartphones than they currently do.

CHI'16, May 07-12, 2016, San Jose, CA, USA
ACM 978-1-4503-3362-7/16/05.
http://dx.doi.org/10.1145/2858036.2858267

RELATED WORK

In this section, we discuss the need for research on smartphone authentication and present research on replacing or extending current systems like PIN and the Android unlock pattern. We also outline work that has looked into users' real world performance and perceptions of these mechanisms.

Smartphone Authentication Systems

Currently, the most common smartphone authentication mechanisms are PINs, the Android unlock pattern, and biometrics (e.g., Apple's Touch ID or Android's Face Unlock). As they are widely used, security and usability problems become more important. These systems need to prevent unauthorized parties from gaining access to devices (security), while also minimizing the legitimate user's burden (usability), in terms of both cognitive load (e.g., remembering a PIN) and the time it takes to successfully authenticate.

PINs and patterns are both prone to guessing attacks, as users tend to choose easy to remember and consequently easy to guess secrets [1, 6, 20]. In addition, authentication codes are highly susceptible to shoulder surfing (i.e., a person spying on the user's input in order to steal the secret) [22]. Aviv *et al.* also found that smudges on the screens can easily be used to infer an Android unlock pattern [2]. Finally, side channel attacks using built-in sensors (e.g., accelerometer, microphone, etc.) have been proven to be efficient ways to infer a user's PIN or pattern [3, 18].

Biometrics have recently become a popular authentication option for smartphones [8]. While they solve many of the security and usability problems of previous mechanisms, they also have some shortcomings. Besides privacy issues, researchers showed that some biometric systems can be tricked with relatively simple methods (e.g., [11]). In addition, users may perceive these systems as being insecure [17] or awkward to use in specific situations [4, 8]. Thus, getting biometrics right is hard and important: if a mechanism is seen as low-effort *and* secure, users may be motivated to protect their devices when they otherwise would not [8, 10].

There has also been significant research effort to solve existing security problems of smartphone authentication. These include additional biometric security layers for PINs [24] and Android patterns [7], external hardware [5], or improving security by visual methods like indirect input [14, 19, 21]. This is just a small excerpt of the huge body of literature on smartphone authentication. However, for any of these alternative methods to be successfully adopted, a detailed understanding of how users interact with existing smartphone authentication mechanisms *in situ* is needed.

Real World Insights

To provide a baseline understanding of users' interactions with existing smartphone authentication systems, recent research has begun to examine the ways in which users interact with these mechanisms outside of laboratory environments.

In 2013, von Zezschwitz *et al.* [23] provided first insights on the differences between PINs and the Android pattern in a real world setting. However, in their study, users only provided one data point per day, which made real world applicability of the data tenuous at best. In order to fill this gap, Harbach *et al.* [12] provided the first real world data on the amount of time users spend unlocking their smartphones relative to the total amount of time spent using them. They also employed experience sampling to provide quantitative data on the likelihood of shoulder surfing, showing that there are very few cases when users perceive this threat. This highlights the appropriateness of context-dependent authentication systems as proposed by Hayashi *et al.* [13].

Egelman *et al.* [10] performed a series of interviews and online surveys to provide data on why (or why not) users choose to secure their smartphones with locking mechanisms. They also quantified and correlated the breadth of sensitive data stored on users' devices with users' perceptions surrounding that data. For instance, they showed that access to email is a heavily underestimated risk.

While this previous work has painted a better picture of how users interact with smartphone locking mechanisms and suggested ways in which smartphone locking mechanisms can be improved, we are unaware of any prior studies that have examined lock screen interactions in detail. The data presented in this paper contributes to this line of work by filling a gap left by Harbach *et al.* [12] and Egelman *et al.* [10] in that we provide detailed quantitative data on users' interactions with these mechanisms *in situ*.

METHOD

Our goal was to answer the following research questions in order to provide a deeper understanding of real world unlocking performance, as well as to collect benchmarks and ideas for the development of future novel locking mechanisms:

1. How much time do users spend interacting with lock screen UI elements each day?
2. Does wearing a watch influence how frequently the phone is activated or unlocked?
3. How much time do users spend on the lock screen before beginning to unlock?
4. How long does a successful unlock attempt take?
5. How do code length and line visibility impact successful unlock time?
6. How many errors do users commit when unlocking their phones?
7. How long do additional attempts take and how much time is wasted through errors in total per day?
8. Which types of errors occur most frequently?
9. How frequently does the lock screen actually prevent someone from accessing the phone?
10. What are users' subjective views of the time it takes to unlock and the errors committed?

In order to obtain the fine-grained field data about users' unlocking behaviors needed to answer these questions, we had to instrument users' primary smartphones. This also required a modification of the Android operating system and thus flashing a new operating system onto users' devices. To facilitate such studies, the University of Buffalo offers aca-

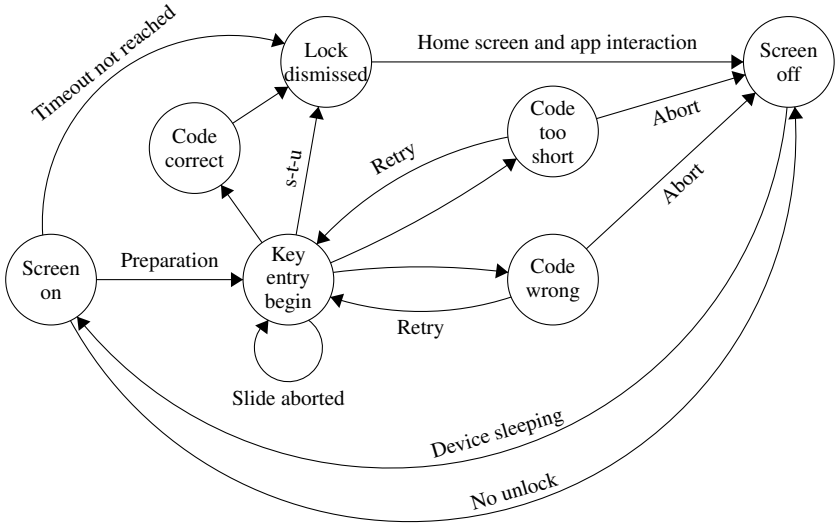

Figure 1. The events and transitions logged during data collection. Sessions are framed by screen on and off events. After key entry has begun with the first touch on the unlocking UI elements, the entered code can either be correct, wrong, or too short. If the code was correct, code entry was not yet again necessary, or no code is needed (slide-to-unlock), the lock screen is dismissed.

demic researchers access to the PhoneLab panel[1] [15] of more than 200 participants who have received customized smartphones. These LG Nexus 5 phones run a modified Android Open Source Project (AOSP) build and are instrumented to send log entries to a central server. The phones are periodically updated over-the-air so that data collection and experimentation modifications can be deployed in a flexible manner. Participants primarily include students and staff from the University of Buffalo.

We extended the instrumentation available on PhoneLab to collect the necessary data, which was transparently distributed to the panel as an over-the-air update of their phones. At the end of the study period, we also deployed a survey instrument to participants to collect qualitative information on their unlocking behaviors. In the next sections, we describe our instrumentation and the exit survey in more detail.

Logging

To gather detailed data on unlocking events, we created log entries that were modeled based on a finite state machine (Figure 1): a session usually starts with the screen being turned on. Following that, a user may interact with notifications and other widgets on the lock screen, she may further unlock the phone, or turn the screen back off. Turning the screen off involves no interaction with lock screen elements (i.e., the user can either press the power button or allow the screen to timeout), whereas further unlocking the device involves interaction with lock screen UI elements. This initial interaction can amount to entering the first digit or character, inputting the first pattern stroke, or simply beginning to slide across the screen to unlock the phone. We denote this point in time as `KeyEntryBegin`. Previous work referred to the time before beginning key entry as preparation time [21].

After key entry begins, the entered code can either be correct, incorrect, or too short. If the code is entered incorrectly five times, the user will be prevented from trying again for 30 seconds. If a code was too short (three digits/strokes or less), Android does not count the attempt towards the number of failed attempts. After the key was entered correctly, the `Keyguard` (i. e., the lock screen) is dismissed. We also log this event to detect dismissals without code entry, e. g., when the lock screen timeout (the time before a re-authentication is necessary) was not yet reached or when the user is not using a code-based lock. The latter case means that the slide-to-unlock method (s-t-u) is used, which does not involve a code, but simply sliding the finger across the screen to dismiss the lock. Android also offers the option to entirely disable the lock screen, which causes the phone to immediately show the home screen when pressing the power button.

After a failed or too-short unlock attempt, a user can either retry (if any attempts are left), or abort unlocking altogether, which will be logged as a screen off event. Finally, if the phone was unlocked successfully and subsequently interacted with, the screen will eventually be turned off again (by manually pressing the power button or reaching an idle timeout).

To capture the necessary data, we relied on log output captured by the PhoneLab instrumentation. The custom AOSP build already included log output for certain system events. We reused output generated for screen-related events. `SCREEN_ON` and `SCREEN_OFF` actions are broadcasted when the user presses the hardware button to turn on the screen and when a timeout or another button press causes the screen to turn off respectively. As in the previous study by Harbach *et al.* [12], these events frame the user's interaction with the device. We then modified several classes in `com.android.keyguard` to capture the lock-related events. With each event, we captured additional information, such as the type of lock screen being used (i.e., PIN, pattern,

[1] http://phone-lab.org – last access 08/31/15

or slide-to-unlock), the length of the entered code, whether or not a drawn pattern was visible (the so-called "stealth mode"), the reason for turning the screen off (by the user or a timeout), and how many attempts had already been made. The accuracy of the collected data, especially the timestamps, will be discussed in the Limitations section.

Finally, in post processing, we converted the sequence of events in the log into statistics about the respective transitions of the state machine.

Survey

At the end of our data collection period, PhoneLab staff emailed participants a link to our online exit survey. As an incentive to complete the survey, we created a drawing for a $100 Amazon gift card. The survey asked for demographic information and subjective views of the unlocking process. These views were collected by proposing statements (e. g. "unlocking my phone is easy") to which participants were asked to indicate their agreement on a 7-point Likert scale. We also asked them open-ended questions to tell us about situations in which they were particularly happy to have a lock screen or particularly annoyed to have a lock screen, and whether there are any situations in which they usually struggle to unlock their phones.

Statistical Analysis

There are two ways in which we can look at the log data we collected: (a) summarizing the logged data by user and comparing users' performances; and (b) using the raw information. The latter case allows for a detailed view on the entire distribution of all collected events, while the former results in a loss of information. However, most statistical tests operate under the assumption that individual data items are independent of each other and thus require summarizing per user first. When looking at differences between lock screen types, we will thus be relying on data that is first summarized per user. However, to also give a sense of the total distribution of the timing data we gathered, we will also be discussing some of the raw values. To clarify which data was summarized per user first and which was not, per user data will be marked with a section sign (§) and raw data with a double dagger (‡).

When conducting significance testing, we use non-parametric tests, as many of the distributions we encountered were skewed. Unless otherwise noted, we use Kruskal-Wallis rank sum tests for between-groups analyses with more than two groups and Wilcoxon rank sum tests for data between two groups as well as post-hoc pairwise testing with Holm corrections. As there is no immediate effect size measure for Kruskal-Wallis tests, we report the effects of the significant pairwise tests using the $r = Z/\sqrt{N}$ metric.

RESULTS

We collected data from July 15 to August 31, 2015 using the PhoneLab panel, yielding data from 202 individuals. Before analysis, we cleaned the data to exclude participants from whom we received less than 30 full days of data (57 users). These participants are likely to either not have installed the AOSP update during the period of study or did not connect their phones to WiFi on time to upload the logging data. We trimmed the remaining data to only include full days from midnight to midnight. We also cut down the data to the first 30 full days, so that we can base the analysis on an equal amount of time per participant. Finally, we removed a further 8 participants who used multiple types of lock screens during the study period, 2 who used a password, and 1 who disabled the lock screen altogether, as their small numbers preclude comparative analysis.

For these remaining 134 participants, we preprocessed log data to discard events that were out of order. This was likely caused by threading issues in AOSP, causing a log entry for a later event to be written before an earlier event's entry. This and data accuracy are discussed in the Limitations section. Overall, cleaning amounted to 1.2 % of events being discarded on average§.

N_{survey}	71	
Age	19 – 70 years, Mdn = 35 years	
Gender	36	female
	34	male
	1	n/a
Ethnicity	38	white
	5	hispanic/latino
	4	black/african american
	1	native american
	14	asian/pacific islander
	6	other
	3	n/a
Highest degree	11	high school diploma or less
	40	bachelor/master degree
	18	doctorate/professional degree
	2	n/a
Annual household income	34	less than $50k
	19	less than $100k
	13	more than $100k
	5	n/a
$N_{logging}$	134	
Lock screen type	32	PIN
	35	pattern
	67	slide-to-unlock
Avg. PIN length	4.25	digits (sd = .84, range: 4-8)
Avg. pattern length	5.9	cells (sd = 1.81, range: 4-9)
Pattern strokes invisible	8	participants

Table 1. Participant demographics.

Of the 134 participants we received log data from, 71 completed our survey. We suspect that the remaining 63 participants were not motivated to respond, as prior PhoneLab data collection has been effortless for them. Table 1 gives an overview of the available demographics. The type of lock screen used did not differ based on demographic properties.

In total, we observed an average of 246.4 events per user per day (σ = 181.7, Mdn = 197.7, ranging from 33.4 to 1,170.9)§. In the following, we will present data to answer the research questions introduced above.

Total Time Spent Using and Unlocking

Our first goal was to examine the actual time participants spend interacting with the security mechanism per day to get an idea of how much effort they are investing. Previously,

Method	# sessions/day	# unlocks/day	Time spent unlocking/day [s]	Locked Session Length [s]	Unlocked Session Length [s]
Slide	75.0 (56.2, 58.4)	42.6 (29.6, 35.8)*	8.0 (5.2, 6.8)$^{*+}$	87.1 (178.4, 44.6)	287.3 (225.8, 224.6)*
Pattern	76.8 (54.7,62.5)*	44.1 (32.0, 36.5) $^{+}$	64.1 (87.8, 48.7)*	58.1 (65.4, 32.0)	316.4 (177.4, 264.2)
PIN	53.2 (44.7,46.1)*	29.5 (22.6, 22.3) $^{*+}$	58.3 (69.2, 36.8)$^{+}$	59.6 (59.5, 38.7)	537.0 (762.3, 331.3)*
Overall	70.3 (53.8, 57.1)	39.9 (29.2, 31.8)	34.7 (61.8, 18.3)	73.0 (133.9, 39.1)	354.6 (423.6, 259.5)
	$\chi^2_2 = 7.0$, $p = .03$	$\chi^2_2 = 7.5$, $p = .02$	$\chi^2_2 = 89.6$, $p < .0001$	$\chi^2_2 = 2.43$, $p = .30$	$\chi^2_2 = 6.75$, $p = .034$
	$r^{*} = .32$	$r^{+} = .47$, $r^{*} = .41$	$r^{+} = .79$, $r^{*} = .75$		$r^{*} = .25$

Table 2. Session statistics by type of lock screen used (mean, σ, Mdn)[§]. Pairwise significantly different cells in each column are marked with an asterisk* or plus sign^{+} (holm-corrected Wilcoxon rank sum tests $p < .05$).

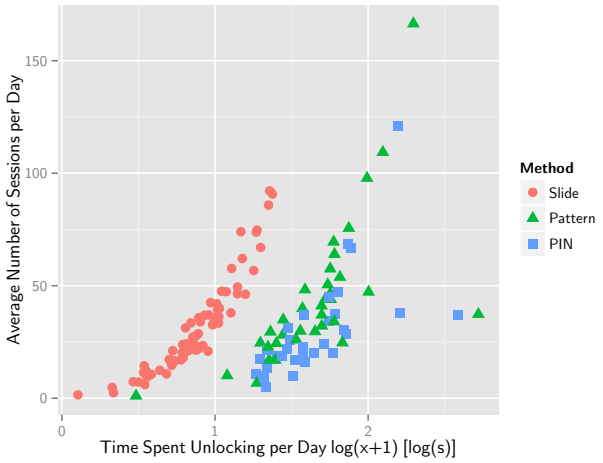

Figure 2. Relationship between time spent unlocking per day and average number of sessions per day given different lock screen types[§].

Harbach *et al.* [12] reported that users spend an average of 2.6 minutes per day unlocking their devices. However, their analysis considered the total time from turning the screen on to dismissing the lock screen. It stands to reason that only a fraction of that time is actually devoted to entering a code. Table 2 gives an overview of sessions and unlocks per participant per day. Both sessions and unlocks differ significantly by type of lock screen used. The number of activations and unlocks per day align with the findings of Harbach *et al.* [12]. However, our more detailed data show that much less time is actually spent unlocking the device per day on average.

Furthermore, pattern and PIN users predictably spent significantly longer unlocking their devices than slide-to-unlock users (Table 2). Interestingly, the overall effort in terms of time spent for the code-based locking mechanisms is similar. It is important to note, that while most participants spent less than 100 seconds per day, a few participants spent almost 400 seconds or more (Figure 2). The figure also shows that our set of participants included both inactive and very active users.

To put the time spent unlocking into perspective, Table 2 also provides an overview of participants' average session lengths by lock screen type. The length of sessions with unlocks differed significantly between lock screen types: PIN users' sessions took almost twice as long as sessions by slide-to-unlock users. Pattern users' session lengths fell in between. Session lengths of non-unlock sessions did not differ significantly between lock screen types. So, while PIN users spend significantly more time after having unlocked the device, they also unlock less frequently. This suggests a different usage pattern for these participants.

Finally, for each session, we also collected the termination reason, that is, how the screen was turned off. On average, participants terminated sessions themselves (using the power button) in 61.9 % of sessions (σ = 26.8 %, Mdn = 71.8 %, ranging from 1.8 % to 95.7 %)[§]. As expected, this proportion did not differ significantly between lock screen types (Kruskal-Wallis $\chi^2_2 = 2.4$, $p = .3$)[§].

Wearing a Watch

When designing this study, we realized that there could be a mitigating factor for the difference between activations and unlocks observed in previous studies: users regularly wearing watches may rely less on their phones to tell the time. Also, having a smartwatch would further reduce this need by showing notifications on the watch. So, as a side note to the main aim of this paper, we asked in the exit survey whether participants regularly wear a watch or smartwatch. Out of the 71 participants who responded to the survey, 31 indicated they wear a watch, and two a smartwatch, on a regular basis. While this precludes looking at the specific effect of smartwatches on activations, we found that the 33 participants with any type of watch (61.3 activations per day, σ = 56.3, Mdn = 46.6) activated their phones significantly less frequently (W = 456, $p < .05$, $r = .23$) than participants without watches (68.1 activations per day, σ = 31.0, Mdn = 64.6). The number of unlocks per day, however, did not differ significantly between these two groups.

Time Needed Before an Unlock Attempt

As already shown above, our data provides a more detailed look at the unlock process. So next, we wanted to see whether the time before key entry begins differs for lock screen types. This first step after turning the screen on has been referred to as "preparation." Depending on a user's intention, he or she may look at notifications first and then unlock the phone, or unlock the phone straight away; the preparation phase measures how long she spent on the locked screen prior to initiating an unlock attempt (i. e., starting to draw a pattern, enter a PIN, or sliding her finger). In Figure 1, this is the transition between "screen on" and "key entry begin."

We observed that preparation times vary widely. Averaging per participant, the mean preparation time is 71.5 seconds per attempt (σ = 131.9, Mdn = 38.7)[§] when the participant does not unlock the device. If an unlock follows the preparation, however, mean preparation time per participant and attempt is

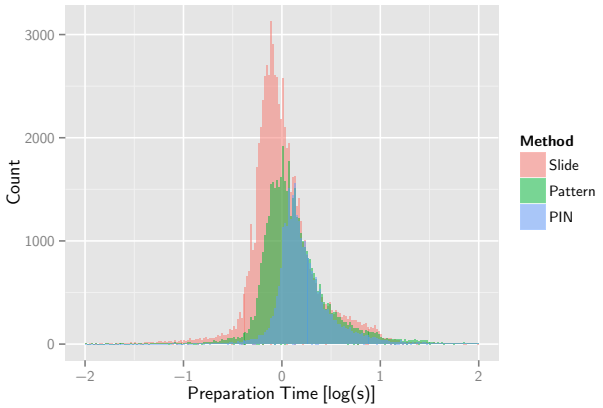

Figure 3. Distribution of preparation time between lock screen types‡. The log-scale x-axis shows sub-second intervals to the left of zero and longer intervals to the right.

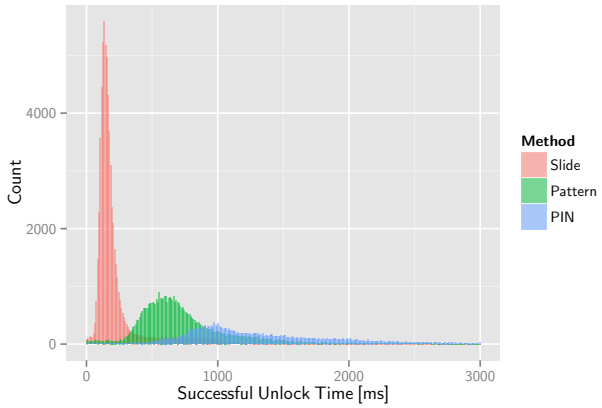

Figure 4. Distribution of successful unlock attempt time between different lock screen types‡.

22.7 seconds (σ = 84.1, Mdn = 3.4)§. As the median value suggests, the majority of preparation intervals is briefer than four seconds and likely represents the cases where users do not check notifications first.

Figure 3 shows a log-scale histogram of preparation times separated by lock screen types used when an unlock followed‡. Preparation times significantly differ between lock screen type (K-W χ^2_2 = 22.2,p < .0001)§, with pairwise tests yielding significant results (p < .05, .24 < r < .42) between all lock screen types. Median preparation time for slide-to-unlock is 2.2s, 4.3s for pattern, and 9.8 s for PIN.

One explanation for the differences we observed in preparation times are recall processes. For instance, preparation times for slide-to-unlock might be so brief because there is almost no recall involved; one can slide the finger across the device in the right spot without even looking at the screen. Pattern might be quicker than PIN because the user needs to recall the PIN first, while with a pattern, the user only needs to find the starting position. However, there is an alternative hypothesis: as we cannot prove causation (an additional study is needed), slower people may simply be more predisposed to using PINs rather than patterns.

Duration of Successful Unlocks

Next, we look at successful unlock attempts to establish a baseline that future mechanisms can use to evaluate their performance. The data we collected includes 160,746 unlock sessions, of which 75,298 (46.9 %) involved entering an unlock code. The remaining 53.1 % pertained to users without code-based lock screens or unlock attempts where entering the code was unnecessary (timeout not yet reached). Of these code-based unlock sessions, 68,739 (91.2 %) were immediately successful (i. e., unlocking the device on the first try) and took 1.18 seconds on average (σ = 33.98, Mdn = .82)‡. In the remaining code-based unlock sessions, participants aborted their unlock attempts in 768 (1.0 %) cases, and another 5,790 (7.7 %) unlock sessions involved at least one error. These will be discussed in more detail in the next section.

The time a successful unlock takes is very different between lock screen types (K-W χ^2_2 = 107.6, p < .0001, pairwise tests all p < .0001, pairwise effect sizes .74 < r < .82)§: slide-to-unlock users take 230ms on average (σ = 43,Mdn = 224), pattern users 910ms (σ = 625, Mdn = 739), and PIN users 1,963ms (σ = 1,665,Mdn = 1,535). The differences also become apparent in the histogram of all successful unlock attempts (Figure 4)‡. Given the long tails of the distributions, Table 3 lists the quantiles for each unlock method‡.

Method	25 %	50 %	75 %	90 %
Slide	120	150	210	420
Pattern	510	660	880	1,225
PIN	920	1,200	1,729	2,380
Overall	158	480	880	1,397

Table 3. Quantiles of successful unlock times in milliseconds‡.

Influence of Code Length and Line Visibility

Another aspect worth looking at is the correlation between code length and successful unlock time. Unfortunately, only 2 PIN users did not have a PIN of length four, which precludes statistical analysis for this lock screen type. However, pattern users showed more variety: eight participants chose a pattern of length 4, thirteen of length 5, two of length 6, four of length 7, two of length eight and seven of length 9, allowing us to take a closer look at timing differences between those groups. To this end, we fit a linear model using pattern length to explain average time for a successful unlock (Figure 5, $F(1, 33)$ = 36.53, p < .0001, adjusted R^2 = .51)§. Per additional cell used in the pattern, the successful unlock time increases by 147 ms on average.

Finally, there was no significant difference between the successful unlock times for participants with hidden pattern strokes (i.e., “stealth mode”; 8 participants) and those without (27 participants, W = 135, p = 0.30)§. While this indicates that our stealth mode participants did not observably take more time to correctly enter their patterns, this data is confounded by the fact that these two groups were self-selected: there is no evidence that forcing all participants to use stealth

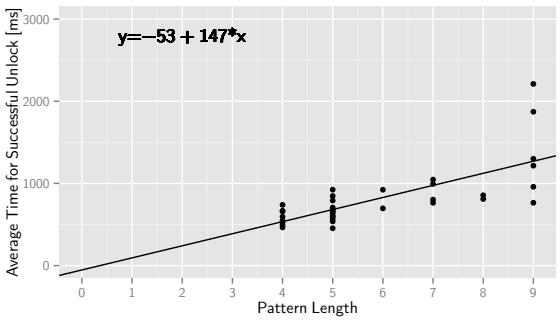

Figure 5. Regression model built to explain successful unlock time based on pattern length[§].

mode would not increase their unlock times. That is, those who chose to enable stealth mode were likely already comfortable entering their patterns.

Number of Unlock Errors Committed

In this section, we examine the number of errors the 67 participants of code-based lock screens committed. In 6,558 instances (11.5 %), unlock attempts did not result in success immediately or at all. Pattern users committed errors much more frequently (5,667 of 46,921 attempts, 12.1 %) than PIN users (891 of 28,376 attempts, 3.1 %)[‡]. On average, pattern users needed 1.13 attempts (sd = .06, Mdn = 1.11) to successfully unlock their devices, while PIN users only needed 1.04 attempts (sd = .03, Mdn = 1.03, W = 1064, $p < .0001$, r = .88)[§]. Thus, given the average number of unlocks per day presented above, a PIN user would commit 1.3 errors (σ = 1.2, Mdn = 1.0, ranging from .07 to 4.1), while a pattern user would commit 8.0 (σ = 6.6, Mdn = 4.9, ranging from 0.4 to 26.4)[§].

Time to Recover from Errors

Unlock errors make the unlocking process take longer, as the user needs to recover. This includes realizing that an error was made, potentially remembering the correct code, and then retrying. Table 4 gives an overview of the time additional unlocks take for PINs and patterns. Pattern users spend twice as much time on successive attempts, whereas PIN users spend only about a third more. Given the average error rate, unlocking errors contribute 13.4 seconds (20.9 %) to each pattern user's daily unlocking time of 64.1 seconds on average[§]. For PIN users, this time amounts to 3.5 seconds (6.0 %) of the overall average of 58.3 seconds[§]. Pattern users thus need to invest more than three times as much effort in terms of time to compensate for errors.

Method	First attempt		
Pattern	791	σ = 371	Mdn = 703
PIN	1,952	σ = 1,765	Mdn = 1,500
Method	**Further attempts**		
Pattern	1,699	σ = 2,805	Mdn = 1,069
PIN	2,668	σ = 1,141	Mdn = 2,413

Table 4. Mean unlocking time per user based on attempt and lock screen type in milliseconds[§].

Besides using their phones less frequently, the lower error rate provides another explanation of why the overall time PIN users spend unlocking is similar and even a little lower than the pattern users', even though PIN users take longer to unlock with each session and additional attempts also take longer. This also provides an opportunity for improvement: reducing the error rate of pattern users would allow them to save up to a fifth of their unlocking time per day on average. This may also make the pattern lock more attractive for users currently not using a code-based locking mechanism.

Types of Errors Committed

Looking at the types of errors committed, PIN and pattern users show similar behaviors. Table 5 lists the most common errors for both mechanisms. Single errors are the most frequent, accounting for more than 70 % of errors in both unlocking methods[‡]. Interestingly, for both methods, single errors without successful unlocks are in the top five. Those most likely represent accidental inputs without the true intention to unlock (e.g., "pocket dialing"). Truly critical errors – those attempts where the lock screen was not dismissed after five failed attempts – are discussed in the next section.

Method	Error type	Count	Proportion
Pattern	fs	2736	48.3 %
	ts	1353	23.9 %
	t	385	6.8 %
	ffs	269	4.8 %
	tts	192	3.4 %
	fts	135	2.4 %
	f	120	2.1 %
PIN	fs	477	53.5 %
	ts	291	32.7 %
	f	30	3.4 %
	tts	27	3.0 %
	t	21	2.4 %

Table 5. Types of errors by unlocking method, explaining more than 90 % of total errors. f=failed (code wrong), t=too-short (code too short to be counted as proper attempt), s=success[‡].

Too-short errors are responsible for 43.8 % of pattern users' failed unlock attempts. While not counted towards the lockout threshold (i.e., the number of failed attempts the user can make before the device prevents additional attempts), these 2,487 (5.3 %)[‡] of all pattern entry attempts are delaying the unlock process, likely due to slipping up while entering the pattern by lifting the finger too early. Last, enabling stealth mode (i. e., when the pattern strokes are invisible) did not observably influence the number of additional attempts necessary to unlock the device (W = 73, p = .17)[§].

Access Prevented

The last aspect of our data analysis concerned whether we would actually be able to observe instances where the lock screen prevented access to the device. Such unlock attempts have been called critical errors in previous work (e. g., [9]) and are usually assumed to happen after three attempts. On Android, however, critical errors occur after having entered a PIN or pattern incorrectly 5 times, excluding attempts that are too short. The user then has to wait 30 seconds before being able to attempt key entry again. Critical errors in a field

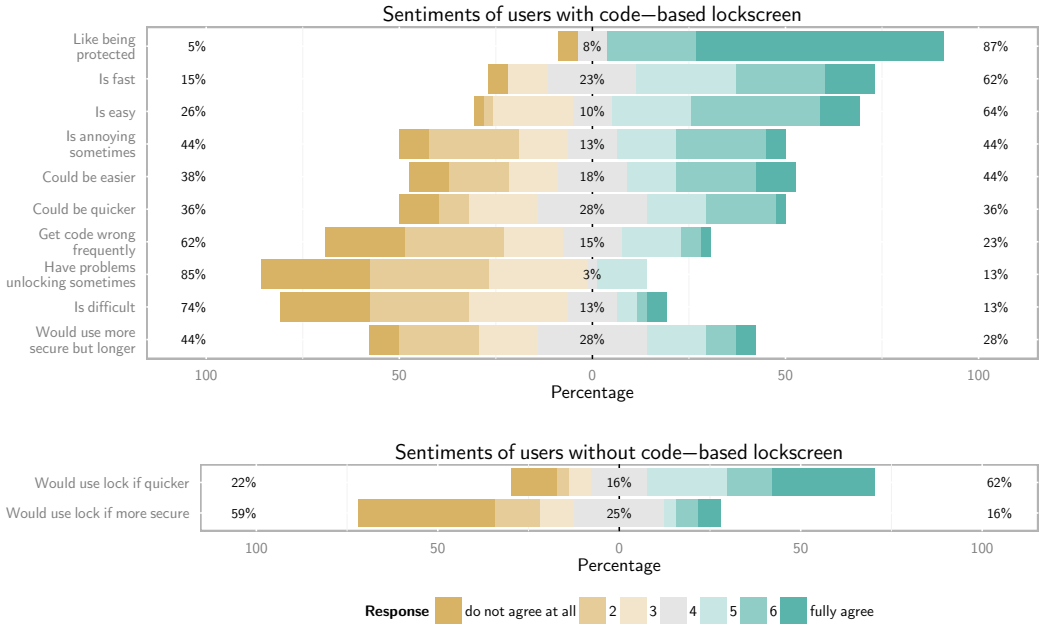

Figure 6. Sentiments of survey participants. They were asked to rate their agreement to the respective statements on a 7-point numerical scale.

study scenario are either cases where a legitimate user has trouble remembering or entering the code (false positives), or an unauthorized user was trying to get access (true positives).

We observed no critical errors for PIN users and only six critical errors committed by five different pattern users. These included a large number of additional too short attempts, suggesting that they may result from unintentional input or someone like a child playing with the device.

On the other hand, we can look at all unlock attempts for code-based lock screens that were aborted with at least one failed entry attempt as an upper bound of access attempts that were prevented. We observed such cases for 28 pattern users and for 12 PIN users. Individual PIN users had as many as 8 such events while one pattern user had aborted attempts in 30 instances. These numbers would suggest that lock screens are effective at keeping others from using the phone. However, it is very likely that they include instances where our participants aborted themselves. Of course, this does also not account for cases where simply seeing an active, code-based lock screen deters an unwanted person from using the device. A dedicated investigation is necessary to determine lock screen efficacy in terms of its security goals.

Altogether, our dataset seems to suggest that truly critical errors are very rare events for most users: they do not happen within a given month. Yet, lock screens may still be effective in deterring unwanted use.

Qualitative Data

After finishing log data collection, we sent out survey invitations to all PhoneLab participants. As mentioned before, 71 completed the survey. Of those, 39 used PIN or pattern as their lock screen. Figure 6 gives an overview of sentiments toward code-based lock screens that the 71 respondents provided. Participants appreciate the protection that their lock screens afford them and they mostly think their lock screens are fast and easy. However, they also indicated that lock screens can be annoying sometimes and that they would like to have easier and quicker alternatives. Most participants did not feel that they get codes wrong frequently or have problems unlocking their devices. Code-based locks were also not perceived as being difficult. The responses of participants with a code-based lock screen did not differ significantly based on their respective type of lock screen.

The remaining item asked participants with code-based lock screens if they were ready to invest more time into using a more secure alternative. The majority of respondents did not agree with this statement. Similarly, respondents without code-based locks would not adopt a secure lock screen if it was just more secure. However, most of them would be ready to adopt one if it took less of their time.

We also asked participants to describe situations in which they were particularly happy to have a code-based lock screen on their device. The 18 respondents who gave accounts of such instances mostly described situations where a phone was misplaced or lost and the code prevented attackers from either abusing or keeping it in the participant's view. One participant said: *"Dropped my phone while walking on [a beach]. I think the phone was left on a railing because it did have a lock feature."* The other type of situation several participants mentioned was when visitors, such as dates, family members or children are present and the code then affords them protection: *"When I went to visit my 5 year old cousin and he always wanted to play with my phone."* Another participant

put it differently: *"Always [happy to have it]. I just wish it worked properly."* These statements also implicitly communicate the trade-off between security and usability: while the events during which the code protected their device and data are rare, the level of security it affords them is appreciated while the effort is accepted. It could also be that encountering such events is an important precursor or motivating factor for adopting or keeping a code-based lock screen.

Finally, we asked our exit survey respondents to describe situations in which they particularly struggle to unlock their devices. The 32 open-ended responses all describe contexts that can make interacting with mobile devices more difficult in general. These situations involve either being distracted (*"while driving"*), being in a rush (*"when I want to take a video/picture fast"*), having limited dexterity (*"when I have to use my left hand instead of my right hand"*), or when environmental conditions make recognizing touch input difficult (*"rain, sweat"*, *"heat, humidity"*, *"when I have gloves, in winter"*). Two respondents also pointed out limitations of the available user interface. They mentioned that it is difficult to hit the correct keys in difficult lighting conditions (*"when it's bright out"*) and in general (*"I just accidentally hit [numbers] next to the ones I want frequently"*). Another said, *"the last digit of my password and the enter button are in close proximity to each other and I often press enter instead of the digit, which is annoying."*

Therefore, based on respondents' subjective points of view, the potential for improved solutions lies in providing quicker alternatives to users who currently do not have lock screens and improving the performance and UI of existing lock screens in difficult situations and contexts.

DISCUSSION

In this paper, we provided deeper insights into the individual parts of the smartphone unlocking process. Going beyond the insights previous studies have provided, we were able to provide detailed benchmarks for the currently available lock screens on the Android operating system. Overall, the data presented shows that smartphone use in general and unlocking behavior in particular varies widely between and within users. The large ranges of measured data and the corresponding standard deviations, as well as the long tail of the distributions are a testament to that.

However, clear patterns also emerged that show differences based on lock screen type: PIN users interact with their phones for longer, but less frequently. PIN users also need longer to prepare unlocking and successful unlocks take longer, but are also less error-prone. On the opposite end of the spectrum, slide-to-unlock users interact with their phones more frequently but for less time, while also needing very little time to unlock due to the lack of a code. Pattern users appear to be using their phones as frequently as users without a code-based protection and for similar amounts of time. However, they spend about the same overall amount of time unlocking their devices as PIN users because they commit almost six times as many errors.

Thus, looking at our data, it appears that the three most commonly used mechanisms to lock smartphones allow users to make a tradeoff between security and unlocking speed or convenience given their unlock frequency. The subjective views of the unlocking process appear to support this: participants with a code-based lock screen were generally satisfied with their unlocking experience and few considered the process problematic. Participants who do not currently have a code-based lock screen indicated that they would be most motivated to adopt a more secure alternative if it took less time.

We also provided a code-length based model for pattern unlocking that will allow a comparison of the time it takes to unlock given the pattern length. Our data additionally showed that whether or not the lines are visible during pattern entry did not observably influence unlocking time or error rates (though this may be confounded by self-selection bias). Approximations for this were previously only available from lab studies. In addition, a dedicated, more detailed deconstruction of factors influencing the usability of the pattern lock screen could be of interest, as the work of von Zezschwitz *et al.* [22] suggests that several other properties of patterns (for example knight-moves or corners) impact security.

Another interesting aspect to note is that most PIN users limited themselves to four digits. The prevalence of PINs of that length in other applications likely influence this. As the relatively small search space for four-digit PINs and the lack of strong rate limiting on Android lock screens are security concerns, future work should look at avenues for improving the security of these users while maintaining acceptable usability. One possibility could be an improved PIN lock screen that uses spaced repetition techniques to effortlessly "teach" users one or two additional digits over time, similar to the approach presented by Schechter and Bonneau [16].

In terms of unlocking errors, the users in our dataset seem to be well trained on their chosen locking mechanisms, as relatively few errors were committed, and virtually none could be considered critical. We only observed six attempts that caused the user to wait before attempting further key entries across the one month data collection period. This implies that novel lock screen proposals need to be able to achieve similar error rates in field environments, given the amount of time users are spending unlocking their phones on an average day. Additionally, our analysis demonstrated a potential to reduce the time wasted by pattern users: reducing the number of errors they commit every day could save them up to a fifth of their total daily authentication time.

The qualitative insights we gathered from our participants furthermore provide starting points for improving their experience. One example could be to detect when the phone is held in the less-used hand and then displaying the pattern frame in a smaller space to make it more readily reachable. Another example would be to disable locking after the phone was unlocked once while being in a car or riding a bike. Lastly, improved touch screen hardware could help to improve on environmental influences, such as limited contrast due to ambient lighting conditions or misdetection of touch input from moist hands.

Overall, one could argue that the available mechanisms suit the needs of smartphone users. Room for improvement appears to exist for the error rates of pattern users, which could save a considerable portion of their daily unlocking time. Also, the relatively few participants in the long tail of the time spent unlocking distribution could be a worthwhile target for either improved security (better use of the time spent) or usability (reducing the time). Finally, given the overall reluctancy to adopt code-based lock screens that has been observed in this and several other studies to this day, it is unlikely that a novel mechanism needing more time to unlock than the currently available options would see much adoption.

Thus, an important research challenge to tackle is to create novel designs that take less or at most the same amount of time as current methods, while providing protection against some of the threats current methods offer no protection for. At the same time, research on lock screens needs to shift the focus more towards real world usability (authentication time and error rates *in situ*) instead of, for example, mainly aiming to reduce the risk of shoulder surfing. Therefore, using longitudinal field studies to validate proposals seems paramount, as we observed great variance in our baseline data. A possible way forward would be to focus more on understanding what makes current lock screens fast (e. g. other than motor memory), what real-world security problems exist, and with what frequency they occur.

LIMITATIONS

The data presented in this paper is limited in a few ways. First, due to the non-trivial nature of collecting this data in the field, all our participants came from the PhoneLab panel. The panel is primarily recruited from students and staff of the University of Buffalo and is thus not representative of all smartphone users. Lock screen interactions and subjective views may be different for users not represented by our sample, although there is no indication in previous work that this is a concern for smartphone security.

Furthermore, all participants were using the same type of device, an LG Nexus 5. It is likely that performance of especially the pattern lock can differ based on device size. Also, our data collection was limited to a 30 day interval. Thus, we cannot be sure of the influence of infrequent events – such as travel, mental state, or illness – on the collected data. Last, our insights pertain primarily to the lock screens available on Android 4.4 and the Nexus 5 in July and August 2015. While the previous work by Harbach *et al.* has shown that these are the most used lock screens for Android users, other platforms provide alternatives. For example, Apple's TouchID provides a very fast yet also secure locking mechanism. If it was possible to overcome the restrictions imposed by iOS, repeating this study on Apple devices would certainly be worthwhile. The performance of slide-to-unlock and PIN users likely translates to users of devices of similar size on iOS and Windows, as the user interfaces for unlocking are very similar. However, it is also possible that the groups of users that decide to use different operating systems also exhibit different performance characteristics.

Another limitation concerns accuracy. The accuracy of the timestamps in our log files is limited by the temporal resolution of the AOSP runtime and by its threading and process scheduling. While the logging infrastructure offers timestamps at millisecond resolution, context switches between threads and processes may cause the creation of a log entry to be delayed from the actual user action. By collecting timestamps twice in several parts of the logging instrumentation, we found that context switches occur during the logging of 9.8 % of events and caused an average delay in logging of 494 milliseconds ($sd = 238$, $Mdn = 496$)[‡], in similar proportions for all events we collected. This results in events being out of order if they are shorter than the delay, in which case we excluded them from the analysis during preprocessing. If events are longer than the delay, these instances appear to be shorter than they actually were in our data. As the extent of this bias is limited and occurring with similar frequencies for all events and users, we argue that the conclusions we draw are still valid, although reported averages are likely to be about 50 ms briefer than the actual duration of the event.

CONCLUSION

This paper provided the first detailed look at in situ performance of Android lock screen implementations. We believe that we were able to provide a benchmark against which novel mechanisms can be evaluated. Given the preferences our users expressed, it is unlikely that solutions requiring users to spend more time per day unlocking their devices will find significant adoption. Our data also showed potential for improvements that can inspire new solutions.

We were able to show that unlocking itself takes much less time than previously approximated. Our data also suggests that users interacting with their devices more frequently did so for shorter amounts of time and chose a quicker lock screen type. However, overall, both PIN and pattern users spent the same amount of time unlocking their devices on average. Yet, PIN users need more than twice as long before beginning the unlock process, possibly to recall their PIN. We found that users wearing a watch on a regular basis activate their phones less frequently and that using stealth mode does not influence the unlocking time of participants using the pattern lock. We provide a model for the influence of code-length on successful unlock time for the pattern lock screen and show how errors contribute a large amount of overall unlock time for this method. Finally, we found little direct evidence that lock screens prevent someone else from accessing the phone and provided insights into participants' subjective views of the unlocking process.

ACKNOWLEDGEMENTS

We would like to thank the PhoneLab team at University of Buffalo for their support in gathering the data. This work was partly supported by a fellowship within the FITweltweit program of the German Academic Exchange Service (DAAD), as well as by the U.S. National Science Foundation under award CNS-1318680.

REFERENCES

1. Panagiotis Andriotis, Theo Tryfonas, and George Oikonomou. 2014. Complexity Metrics and User Strength Perceptions of the Pattern-Lock Graphical Authentication Method. In *Proceedings of the Second International Conference on Human Aspects of Information Security, Privacy, and Trust - Volume 8533*. Springer-Verlag New York, Inc., New York, NY, USA, 115–126. DOI: http://dx.doi.org/10.1007/978-3-319-07620-1_11

2. 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, Berkeley, CA, USA, 1–7. http://dl.acm.org/citation.cfm?id=1925004.1925009

3. Adam J. Aviv, Benjamin Sapp, Matt Blaze, and Jonathan M. Smith. 2012. Practicality of Accelerometer Side Channels on Smartphones. In *Proceedings of the 28th Annual Computer Security Applications Conference (ACSAC '12)*. ACM, New York, NY, USA, 41–50. DOI: http://dx.doi.org/10.1145/2420950.2420957

4. Chandrasekhar Bhagavatula, Kevin Iacovino, Su Mon Kywe, Lorrie Faith Cranor, and Blase Ur. 2014. Poster: Usability Analysis of Biometric Authentication Systems on Mobile Phones, In Symposium On Usable Privacy and Security Poster (SOUPS 2014). *SOUPS Poster* (2014).

5. Andrea Bianchi, Ian Oakley, Vassilis Kostakos, and Dong Soo Kwon. 2011. 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, New York, NY, USA, 197–200. DOI: http://dx.doi.org/10.1145/1935701.1935740

6. Joseph Bonneau, Sören Preibusch, and Ross Anderson. 2012. A Birthday Present Every Eleven Wallets? The Security of Customer-Chosen Banking PINs. In *Financial Cryptography and Data Security*, Angelos D. Keromytis (Ed.). Lecture Notes in Computer Science, Vol. 7397. Springer Berlin Heidelberg, 25–40. DOI: http://dx.doi.org/10.1007/978-3-642-32946-3_3

7. Alexander De Luca, Alina Hang, Frederik Brudy, Christian Lindner, and Heinrich Hussmann. 2012. Touch Me Once and I Know It's You!: Implicit Authentication Based on Touch Screen Patterns. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. ACM, New York, NY, USA, 987–996. DOI: http://dx.doi.org/10.1145/2207676.2208544

8. Alexander De Luca, Alina Hang, Emanuel von Zezschwitz, and Heinrich Hussmann. 2015. I Feel Like I'm Taking Selfies All Day!: Towards Understanding Biometric Authentication on Smartphones. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*. ACM, New York, NY, USA, 1411–1414. DOI: http://dx.doi.org/10.1145/2702123.2702141

9. 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 (CHI '14)*. ACM, New York, NY, USA, 2937–2946. DOI: http://dx.doi.org/10.1145/2556288.2557097

10. Serge Egelman, Sakshi Jain, Rebecca S. Portnoff, Kerwell Liao, Sunny Consolvo, and David Wagner. 2014. Are You Ready to Lock? Understanding user motivations for smartphone locking behaviors. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security (CCS '14)*. ACM, New York, NY, USA, 750–761. DOI: http://dx.doi.org/10.1145/2660267.2660273

11. Rainhard D. Findling and Rene Mayrhofer. 2013. Towards Secure Personal Device Unlock Using Stereo Camera Pan Shots. In *Computer Aided Systems Theory - EUROCAST 2013*, Roberto Moreno-Daz, Franz Pichler, and Alexis Quesada-Arencibia (Eds.). Lecture Notes in Computer Science, Vol. 8112. Springer Berlin Heidelberg, 417–425. DOI: http://dx.doi.org/10.1007/978-3-642-53862-9_53

12. Marian Harbach, Emanuel von Zezschwitz, Andreas Fichtner, Alexander De Luca, and Matthew Smith. 2014. It's a Hard Lock Life: A Field Study of Smartphone (Un)Locking Behavior and Risk Perception. In *Symposium On Usable Privacy and Security (SOUPS 2014)*. USENIX Association, Menlo Park, CA, 213–230. https://www.usenix.org/conference/soups2014/proceedings/presentation/harbach

13. Eiji Hayashi, Sauvik Das, Shahriyar Amini, Jason Hong, and Ian Oakley. 2013. CASA: Context-aware Scalable Authentication. In *Proceedings of the Ninth Symposium on Usable Privacy and Security (SOUPS '13)*. ACM, New York, NY, USA, Article 3, 10 pages. DOI: http://dx.doi.org/10.1145/2501604.2501607

14. Sung-Hwan Kim, Jong-Woo Kim, Seon-Yeong Kim, and Hwan-Gue Cho. 2011. A New Shoulder-surfing Resistant Password for Mobile Environments. In *Proceedings of the 5th International Conference on Ubiquitous Information Management and Communication (ICUIMC '11)*. ACM, New York, NY, USA, Article 27, 8 pages. DOI: http://dx.doi.org/10.1145/1968613.1968647

15. Anandatirtha Nandugudi, Anudipa Maiti, Taeyeon Ki, Fatih Bulut, Murat Demirbas, Tevfik Kosar, Chunming Qiao, Steven Y. Ko, and Geoffrey Challen. 2013. PhoneLab: A Large Programmable Smartphone

Testbed. In *Proceedings of First International Workshop on Sensing and Big Data Mining (SENSEMINE'13)*. ACM, New York, NY, USA, Article 4, 6 pages. DOI: http://dx.doi.org/10.1145/2536714.2536718

16. Stuart Schechter and Joseph Bonneau. 2015. Learning Assigned Secrets for Unlocking Mobile Devices. In *Eleventh Symposium On Usable Privacy and Security (SOUPS 2015)*. USENIX Association, Ottawa, 277–295. https://www.usenix.org/conference/soups2015/proceedings/presentation/schechter
17. Hanul Sieger, Niklas Kirschnick, and Sebastian Möller. 2010. Poster: User preferences for biometric authentication methods and graded security on mobile phones, In Symposium On Usable Privacy and Security Poster (SOUPS 2010). *SOUPS Poster* (2010).
18. Laurent Simon and Ross Anderson. 2013. PIN Skimmer: Inferring PINs Through the Camera and Microphone. In *Proceedings of the Third ACM Workshop on Security and Privacy in Smartphones & Mobile Devices (SPSM '13)*. ACM, New York, NY, USA, 67–78. DOI: http://dx.doi.org/10.1145/2516760.2516770
19. Tetsuji Takada and Yuki Kokubun. 2013. Extended PIN Authentication Scheme Allowing Multi-Touch Key Input. In *Proceedings of International Conference on Advances in Mobile Computing & Multimedia (MoMM '13)*. ACM, New York, NY, USA, Article 307, 4 pages. DOI: http://dx.doi.org/10.1145/2536853.2536944
20. Sebastian Uellenbeck, Markus Dürmuth, Christopher Wolf, and Thorsten Holz. 2013. Quantifying the Security of Graphical Passwords: The Case of Android Unlock Patterns. In *Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security (CCS '13)*. ACM, New York, NY, USA, 161–172. DOI: http://dx.doi.org/10.1145/2508859.2516700
21. Emanuel von Zezschwitz, Alexander De Luca, Bruno Brunkow, and Heinrich Hussmann. 2015a. 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, New York, NY, USA, 1403–1406. DOI: http://dx.doi.org/10.1145/2702123.2702212
22. Emanuel von Zezschwitz, Alexander De Luca, Philipp Janssen, and Heinrich Hussmann. 2015b. Easy to Draw, but Hard to Trace?: On the Observability of Grid-based (Un)Lock Patterns. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*. ACM, New York, NY, USA, 2339–2342. DOI: http://dx.doi.org/10.1145/2702123.2702202
23. Emanuel von Zezschwitz, Paul Dunphy, and Alexander De Luca. 2013. Patterns in the Wild: A Field Study of the Usability of Pattern and Pin-based Authentication on Mobile Devices. In *Proceedings of the 15th International Conference on Human-computer Interaction with Mobile Devices and Services (MobileHCI '13)*. ACM, New York, NY, USA, 261–270. DOI: http://dx.doi.org/10.1145/2493190.2493231
24. Nan Zheng, Kun Bai, Hai Huang, and Haining Wang. 2014. You Are How You Touch: User Verification on Smartphones via Tapping Behaviors. In *Proceedings of the 2014 IEEE 22nd International Conference on Network Protocols (ICNP '14)*. IEEE Computer Society, Washington, DC, USA, 221–232. DOI: http://dx.doi.org/10.1109/ICNP.2014.43