SwiPIN - Fast and Secure PIN-Entry on Smartphones

Emanuel von Zezschwitz[1], **Alexander De Luca**[1,2], **Bruno Brunkow**[1], **Heinrich Hussmann**[1]
[1]Media Informatics Group, University of Munich (LMU), Munich, Germany
[2]DFKI GmbH, Saarbrücken, Germany
{emanuel.von.zezschwitz, alexander.de.luca, hussmann}@ifi.lmu.de, brunkow@cip.ifi.lmu.de

ABSTRACT
In this paper, we present SwiPIN, a novel authentication system that allows input of traditional PINs using simple touch gestures like *up* or *down* and makes it secure against human observers. We present two user studies which evaluated different designs of SwiPIN and compared it against traditional PIN. The results show that SwiPIN performs adequately fast (3.7 s) to serve as an alternative input method for risky situations. Furthermore, SwiPIN is easy to use, significantly more secure against shoulder surfing attacks and switching between PIN and SwiPIN feels natural.

Author Keywords
Authentication; Mobile Devices; Shoulder Surfing

ACM Classification Keywords
H.5.m. Information Interfaces and Presentation (e.g. HCI): Miscellaneous

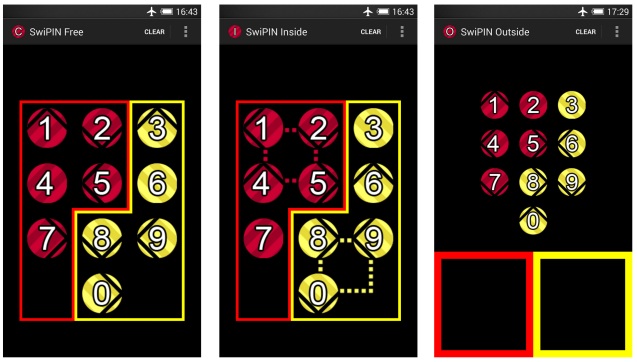

Figure 1. Three different versions of SwiPIN: *free* (left), *inside* (center) and *outside* (right). The currently assigned gesture is indicated by black arrows, no arrow meaning *tap*. To enter a "1" in SwiPIN (*free*), an *up*-gesture within the red-bordered area of the screen would be performed.

INTRODUCTION
As mobile devices can store and access various potentially sensitive data, user authentication on such devices has become indispensable. Despite the advance of alternative biometric mechanisms like fingerprint scanners, knowledge-based systems still represent the primary way of authentication for most users and serve as a fallback solution whenever alternative approaches fail.

PIN is commonly recognized to be a fast and easy way for daily authentication. However, the observability of the PIN-entry can open serious security holes. While some systems like ATMs can partially counter this problem by providing privacy shields and regulated environments, mobile interaction often takes place in uncontrolled (semi-)public situations which increases the threat of shoulder surfing.

A shoulder surfing attack refers to the action of intentionally getting someone's information by direct observation. While countermeasures have been researched for several years, none of these concepts has been widely deployed. Harbach et al. [3] found that observation attacks exist, but are rarely perceived critical. Therefore, we assume that most users are not willing to invest much effort for better protection. As it has already been suggested [5], we argue that novel approaches need to be usability-optimized, easy-to-deploy and should support the entry of traditional passwords (e.g. PIN) to have the chance of a wide user acceptance.

Bianchi et al. [1] proposed several mobile PIN-entry concepts which utilize audio or haptic cues. Roth et al. [6] developed two concepts which display digits in distinct sets. Users repeatedly indicate the respective target set, the intersection of these sets is used to determine the PIN. Similar approaches were proposed by De Luca et al. [2] and Lee [5], exploiting the limitations of the human short-term memory. Most concepts are adequately secure against human-based attacks, but authentication times are usually high (> 8 s). Kwon and Na [4] propose TinyLock which utilizes simple gestures to protect the Android pattern (un)lock from shoulder surfing and smudge attacks. TinyLock shows that simple changes in the user interface can bring significantly more security and at the same time obtain usability.

We present SwiPIN, a concept which allows PIN-entry based on simple touch gestures. Similar to TinyLock [4], it requires only little change in the user interface. SwiPIN was designed to be a fast, easy and secure input method which empowers the user to protect from shoulder surfing whenever such risk is perceived [3]. The results of two user studies show that SwiPIN is fast (M=3.7 s; SD=0.9) and secure against human-based observations. Switching between traditional PIN and SwiPIN felt natural and both approaches showed equally low error rates.

CHI'15, April 18 - 23 2015, Seoul, Republic of Korea.

http://dx.doi.org/10.1145/2702123.2702212

THREAT MODEL

According to our threat model, a smartphone user enters her PIN in a (semi-)public setting. An observer (attacker) standing in the user's vicinity has perfect sight on the display (no reflections, no occlusions). In the first scenario, the input is based on a traditional PIN pad. In a second scenario, the user's device is equipped with SwiPIN. As the user recognizes the potential observer, she switches to the secure mode and enters the PIN using touch gestures. After observing the PIN-entry, the attacker gets in possession of the device and tries to replicate the observed input. The observation attack takes place without additional equipment (e.g. camcorder).

SWIPIN CONCEPT

The SwiPIN concept is the result of a brainstorming with four usable security experts. We presumed that a feasible mechanism would require to support (a) fast and (b) easy input of (c) standard PINs on (d) off-the-shelf smarphones in a way that assures (e) adequate shoulder surfing protection.

SwiPIN is based on a random assignment of simple touch gestures to specific digits. To enter a digit, instead of tapping a specific button, the user performs such a gesture, usually at a different location than the respective button. As the required gesture is displayed as a black arrow on the respective digits (see Figure 1), the user does not need to memorize any additional information. The gesture mapping changes after each input. As soon as the user performs a gesture, the assignment disappears and the gestures are newly mapped. Therefore, the entry of a digit comprises two distinct steps: (a) recognizing the assigned gesture, (b) performing the recognized gesture. Thus, an attacker has to memorize the current gesture mapping to recognize an entered digit. This mapping is too complex to fit in human short-term memory, which makes the task of the observer very hard.

First designs were implemented using low-fidelity paper prototypes and high-fidelity software versions. We performed a preliminary user study and a focus group with the goal to find the best design and the right gesture set. We tested small sets with five distinct gestures and full sets with ten gestures (e.g. multi touch). Overall, the highest potential was found for the systems shown in Figure 1. The three designs use a redundant small set of five gestures (*up*, *down*, *left*, *right*, *tap*) and are based on the traditional PIN pad layout. To map ten digits to five gestures, the PIN pad was divided into two distinct areas, indicated by the colors *red* and *yellow*.

While supporting identical gesture sets, the three illustrated versions of SwiPIN provide different areas to start gesture input (gestures can end anywhere). SwiPin (*free*) (Figure 1,left) allows to start gestures anywhere in the respective half of the PIN pad. SwiPIN (*inside*) forces the user to begin the input within a certain region of the screen, which is indicated by a dotted border. Finally, SwiPIN (*outside*) requires the user to start the gesture at the bottom of the screen. Gestures were illustrated using black arrows, where no arrow meant the tap gesture. Figure 2 illustrates a participant of the user study entering a "2" on SwiPIN (*outside*). For this, a simple "down" gesture is performed. As soon as the gesture is completed, the gesture assignment in the red section changes.

USABILITY AND SECURITY STUDY

Before we compared SwiPIN against traditional PIN-entry, we analyzed the three design alternatives according to their performance, security and user perception.

The study was based on a repeated-measure within participants design. The independent variable was *system* with three levels (*free*, *inside*, *outside*). Each system was tested with three different randomly generated PINs. The sequence of *system* was counterbalanced; PINs comprised each digit only once. We measured authentication time (divided in preparation and input time), error-rate and shoulder surfing success. In addition, we collected qualitative data via questionnaires.

Procedure and Participants

The user study was held in an isolated room at our premises. We started each session by explaining the goal of the study and the SwiPIN concept. For each system, we followed the same procedure: (a) explain the characteristics of the specific design, (b) free training, (c) input of three different PINs and (d) user feedback. During training, the system allowed unlimited input of digits. Entered digits were displayed on top of the screen. Whenever the participant felt ready, the authentication task began. The user entered three different PINs. Each PIN was entered five times with a maximum of three attempts (five runs), resulting in a minimum of 15 correct and a maximum of 45 failed attempts per system. The current PIN was displayed via a pop-up message and dismissed when the user pressed start. Participant answered small questionnaires between the different concepts and filled in a final questionnaire after all systems were tested.

Participants were told to enter the PINs as fast and as error-free as possible. We did not control for hand posture (e.g. one handed), but users could use whatever felt natural. In addition, the PIN-entry was filmed for later security analysis. The whole procedure took about 20 minutes, participants were compensated with a 5 Euro shopping voucher.

We recruited 18 participants (13 male) via the university mailing list and social networks. The average age was 25 years (20-32). Eleven participants had already heard about shoulder surfing attacks. All participants were experienced touch screen users and stated to use smartphones on a daily base.

Results

We excluded the first three runs (authentication attempts) of each PIN $\times$ System combination as training. Therefore, the analysis is based on 108 samples (2 runs $\times$ 3 PINs $\times$ 18 users) per system.

Performance

The performance analysis is based on correctly entered PINs. Our data was normally distributed and allowed for parametric tests. A repeated measures ANOVA comparing the average authentication times of the two runs for each of the three PINs revealed a significant main effect for system ($F_{2,34} = 7.53, p = 0.002$). Bonferroni corrected post-hoc tests showed that SwiPIN (*inside*) (M=5.04 s; SD=0.94) was significantly slower than *outside* (M=4.32; SD=0.94) and *free* (M=4.48 s; SD=0.96; $p < 0.05$). A detailed analysis of authentication

speed distinguishing preparation and input time, revealed almost equal preparation times for all systems ($p > 0.05$). After clicking start, users needed between 1.2 s (SD=0.36; *outside*) and 1.3 s (SD=0.35; *inside*) until they started with the first input. In contrast, input times differed significantly ($F_{1.44,24.50} = 13.48, p < 0.001, \epsilon = 0.72$), ranging from 3.15 s (SD=0.62; *outside*) to 3.79 s (SD=0.70; *inside*). Error-rates were low: participants failed five times (4.6%) using the SwiPIN (*free*), twelve errors (11,1%) were counted using *inside* and six (5.6%) attempts failed using *outside*.

Security

The security analysis was performed by three experienced SwiPIN users (1 female). The shoulder surfers attacked the first correct input of each PIN. Therefore, 54 (3 PINs $\times$ 3 systems $\times$ 6 users) attacks were performed by each participant. Shoulder surfers were paid 20 Euros plus 40 Cents per successfully attacked PIN. They performed shoulder surfing attacks, followed by video attacks. Therefore, we showed them videos that were cut to the respective PIN-entry. Figure 2 shows screenshots of such a video. Per attack, a maximum of three guesses was permitted. Shoulder surfing attacks were based on a one time view of the input followed by three guesses. Video attacks allowed the participant unlimited control of the video (e.g. pause and rewind). For each PIN-entry, we computed the binary success (true/false) and the relative success rate (overlap of correct digits) based on the best of the three guesses. As none of the designs was significantly secure against video attacks, only human attacks are discussed.

Outside performed best with only one exposed PIN (binary: 1 of 54; overlap: 35.6%). *Inside* was successfully attacked in five cases (binary: 5 of 54; overlap: 44.4%). In all cases, the input was slow, allowing the attacker to observe parts of the gesture mapping and guessing the remaining digit(s) by chance. *Free* was most vulnerable to observations (binary: 8 of 54; overlap: 49.5%). The main reason was that participants started their gestures on the intended digit.

Perception

Each system was rated directly after the test. We used Likert scales ranging from 1 (strongly disagree) to 5 (strongly agree). Most of the participants agreed that *free* and *outside* were easy (Mdn = 4) and fast to use (Mdn = 4). SwiPIN (*inside*) was rated lower for both aspects (Mdn=3).

In addition, users were asked to rank the respective systems. Eleven participants (61%) preferred *outside* and rated it to be the fastest and easiest system. The rest of the participants preferred *free*. SwiPIN (*inside*) was not mentioned.

FEASIBILITY STUDY

Based on the results of the first user study, we decided to select SwiPIN (*outside*) for a feasibility study. *Outside* was preferred by most participants and had the best usability and security features. The goal of this study was to compare SwiPIN (*outside*) to traditional PIN-entry in terms of performance, security and likeability. In addition, we were interested in the effects of switching between these concepts.

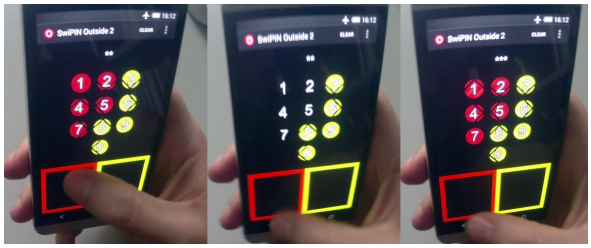

Figure 2. PIN-entry was filmed with a head-mounted camera. This sequence shows a participant entering a "2" using SwiPIN (*outside*).

The study was conducted using a repeated measure within-participants design. The independent variable *system* had the two levels SwiPIN (*outside*) and PIN (*traditional*). The dependent variables were measured at the same levels as in the previous user study. *System* was counterbalanced.

Procedure and Participants

The procedure was similar to the previous study. There were three important changes: 1) we tested only one PIN per system; 2) we had a guided training phase; 3) we added a system switching task. Therefore, we repeated the following procedure for each system: (a) introduction, (b) free training, (c) guided training, (d) PIN-entry and (d) feedback. In the guided training period, the participants were required to enter their PIN ten times, before the actual test was done. After the concepts were tested separately, we assigned a new PIN for the system switching task: (a) PIN training (10 times); (b) traditional PIN-entry (5 times); (c) SwiPIN-entry (5 times); (d) traditional PIN-entry (5 times). Users switched the methods by pressing a software button in the top right corner of the touch screen. At the end of the study, participants were given a questionnaire. The whole procedure took about 15 minutes and was compensated with a 5 Euro shopping voucher.

We used mailing lists to invite 16 participants (12 male) to our study. Their average age was 27 years (21-37). All were experienced smartphone users, eight had used SwiPIN before (within the first user study). No significant differences were found between experienced and novel users.

Results

As for the first study, the results are based on the last two PIN-entries of each device.

Performance

A two-tailed dependent t-test comparing the average authentication times of both runs of PIN and SwiPIN revealed highly significant main effects ($t_15 = 9.79, p < 0.001, r = .93$). Traditional PIN performs significantly faster (1.34 s; SD=0.3) than SwiPIN (3.66 s; SD=0.9). Error-rates were low. Within both tasks, only one failed attempt was logged (3.1%).

A repeated-measures ANOVA on the average authentication times of PIN-entry before and after switching the input method reveals that switching from PIN to SwiPIN and vice versa resulted in significantly slower authentication times ($p < 0.05$). While switching to PIN was error-free, two (12.5%) participants failed their first input on SwiPIN after PIN was used.

Security

The security analysis was equivalent to the one of the previous study. Again, the binary and relative success rates (overlap) were computed based on the best of three guesses.

Video attacks were successful in all cases and will not further be discussed. With respect to shoulder surfing attacks, PINs were correctly identified in almost all cases (binary: 14 of 16; overlap: 92.2%). The two instances that were not correctly shoulder surfed were due to very fast input. SwiPIN was correctly identified in only two instances (binary: 2 of 16; overlap: 35.9%). It has to be noted that in both cases, the attackers guessed the input based on observing the input elements. For instance, in one case, an attacker observed that the participant used the following input areas: red, yellow, red, yellow. This leaves a chance of $(\frac{1}{5})^4 = 0.16\%$ to guess the correct PIN. When observing traditional PIN-entry, we would assume a guessing probability of $\frac{1}{1}^4 = 100\%$ as an attacker could easily observe every button press.

Perception

PIN was rated very easy (Mdn=5) and very fast (Mdn=5) by most participants. SwiPIN was rated easy (Mdn=4) by most participants, but neutral (Mdn=3) in terms of speed. Most participants stated that they liked PIN (Mdn=4) and SwiPIN (Mdn=4). When asked, if they could imagine using SwiPIN in critical situations, all participants agreed. When we asked if switching to SwiPIN in front of others could be perceived as a mistrustful action, most participant disagreed (Mdn=1), one participant agreed (Mdn=4) and one was neutral (Mdn=3).

DISCUSSION

Our results show that SwiPIN (*outside*) is secure against human-based shoulder surfing and has very good usability properties. Even after very short training periods (13 inputs), SwiPIN (*outside*) users performed reasonably fast with 3.66 seconds on average. One drawback of SwiPIN lies in the fact that most people remember their PIN by muscles and this is not possible using SwiPIN. We like to note that SwiPIN was designed to be used as an additional security layer on top of PIN. That is, in low risk contexts like at home, users could opt for the faster standard PIN-entry. This way, users can still rely on muscle memory for most authentications. In addition, we assume that users are willing to accept the extra costs of SwiPIN input since it will only be activated in risky situations. In our lab setting, all participants agreed that they could imagine using SwiPIN for such scenarios. Even if this does not necessarily reflect real-world behavior, such claims are first indicators for the feasibility of the system.

The main reason for SwiPIN (*free*) being the least secure system was due to "bad lies". In these instances, the participants performed the swipes directly on top of the numbers they wanted to input. This means that the system did not take this important human factor into account. Both, SwiPIN (*inside*) and SwiPIN (*outside*) are resistant to this kind of behavior. However, with SwiPIN (*inside*) some participants reported, they felt to be forced to start on a specific position and this resulted in decreased likeability ratings. Besides preventing "bad lies", the *outside* version of SwiPIN additionally hampers observation by physically separating the presentation and the input areas. That is, the attacker had to observe two distinct areas of the screen.

Also due to this, the final system (SwiPIN *outside*) is resistant against most shoulder surfing attacks. The two instances in which it was successfully attacked in the final study were due to plain luck in combination with a clever strategy that allows to reduce the guessing space. That is, by observing the input elements, an attacker can reduce the possible digits by half. Please note that the attacker acknowledged having used that strategy and then simply guessed the PIN. Even if traditional PIN-entry would allow a guessing probability of 100%, this is a limitation as it increases the probability of a correct guess from 0.01% to 0.16%. We like to mention that using a full set of ten gestures (and only one input field) would solve this problem. However, our pre-study indicated that this strategy would also significantly reduce usability.

CONCLUSION AND FUTURE WORK

We presented SwiPIN, a concept which allows secure PIN-entry based on simple touch gestures. Fast authentication speed (3.7 s), low error-rates (3%) and shoulder surfing resistance indicate that SwiPIN is well-balanced in terms of usability and security. Therefore, we argue that it has the potential to be widely accepted as an alternative input mechanism for risky situations.

While first lab evaluations were promising, SwiPIN needs to be deployed and evaluated in the wild. By performing such field studies, we will be able to gather more insights into learning effects, real-world behavior and user acceptance.

REFERENCES

1. Bianchi, A., Oakley, I., and Kwon, D. S. Counting clicks and beeps: Exploring numerosity based haptic and audio pin entry. *Interacting with Computers 24*, 5 (2012), 409 – 422.
2. De Luca, A., Hertzschuch, K., and Hussmann, H. Colorpin: Securing pin entry through indirect input. In *Proc. CHI '10*, ACM (New York, NY, USA, 2010), 1103–1106.
3. Harbach, M., von Zezschwitz, E., Fichtner, A., De Luca, A., and Smith, M. It's a hard lock life: A field study of smartphone (un)locking behavior and risk perception. In *Proc. SOUPS 2014*, USENIX Association (Menlo Park, CA, July 2014), 213–230.
4. Kwon, T., and Na, S. Tinylock: Affordable defense against smudge attacks on smartphone pattern lock systems. *Computers & Security 42*, 0 (2014), 137 – 150.
5. Lee, M.-K. Security notions and advanced method for human shoulder-surfing resistant pin-entry. *IEEE Transactions on Information Forensics and Security 9*, 4 (April 2014), 695–708.
6. Roth, V., Richter, K., and Freidinger, R. A pin-entry method resilient against shoulder surfing. In *Proc. CCS '04*, ACM (New York, NY, USA, 2004), 236–245.