SlidePIN:
Slide-Based PIN Entry Mechanism on a Smartphone

Huiping Sun, Shuaiying Guo, Ke Wang, Nan Qin, and Zhong Chen

School of Software & Microelectronics, Peking University
No.5 Yiheyuan Road, Haidian District, Beijing, P.R.China
sunhp@ieee.org,soft87@126.com,{k3wang,qinn,chz}@pku.edu.cn

Abstract. SlidePIN is a PIN entry mechanism based on slide input method combined with a random numeric keypad. As slide input method ensures higher usability and security, a random numeric keypad introduced, at a slight cost of usability, conspicuously enhances the security of SlidePIN. As an indirect entry mechanism, SlidePIN keeps users away from additional computation or memory burden. Theoretic analysis and experiments show that SlidePIN performs effectively against one-time shoulder surfing attack and better than 4-digit PIN mechanism against multi-time shoulder surfing attack.

Keywords: Smartphone; PIN; Shoulder Surfing; SlidePIN

1 Introduction

Smartphones have globally become the most popular communication tool nowadays which gradually change our life and work. Great amounts of private as well as business information are stored in our smartphones. As the foundation of smartphone security against unwanted access, unlocking mechanisms get more indispensable. 4-digit PIN (Personal Identification Numbers) mechanism that is used most widely asks users to input the PIN directly, which makes it vulnerable to shoulder surfing attacks [1, 2].

A wide range of research tried to resist shoulder surfing attacks to minimize relevant security threats. Two main mechanisms, invisible entry mechanism [5–10] and indirect entry mechanism [11–18], are proposed and developed. However, these mechanisms need either additional hardware or extra computation to ensure security and usability, which motivates us to design a better unlocking mechanism.

The Word-Gesture Keyboard [3, 4] concept was proposed by Montgomery in 1982. The idea suggests using slide gesture to input English words on a touch screen with a soft keyboard. Some implementations of this technology have been developed into products put into market by some companies (ShapeWriter, Swype, TouchPal and etc).

Inspired by Word-Gesture Keyboard, we propose SlidePIN as an indirect entry mechanism with a random numeric keypad and slide input method. SlidePIN inserts users' 4-digit PIN into a slide sequence. Even if attackers captured the slide sequence, extracting the 4-digit PIN from the sequence would still be a conundrum.

To the best of our knowledge, SlidePIN is the first 4-digit PIN entry mechanism that combines slide input method with a random numeric keypad. Generating PIN indirectly based on user's gestures, SlidePIN can effectively prevent one-time shoulder surfing attack and withstand multi-time shoulder surfing attack with better performance, compared with 4-digit PIN.

In following sections of this paper, the concept of SlidePIN will be introduced first. Afterward, theoretical and experimental analysis will be provided to illustrate the improvement on security of SlidePIN and slight compromise of usability, compared with 4-digit PIN.

2 Related Works

In order to withstand shoulder surfing attacks, current entry methods mainly adopt invisible entry mechanism and indirect entry mechanism.

Invisible Entry Mechanism. An invisible entry mechanism [5–10] is to utilize special human-computer interaction methods to implement the input process of the PIN or password. Typical examples are eye tracking [5], tactile sensor [6], pressure sensor [10], vibration sensors [7], back-of-device interaction [8] and physical block [9], etc. It is difficult for the attacker to visually capture the interactions between computers and humans, thus this mechanism has great capability to resist shoulder surfing attack. However, an invisible entry mechanism is not suitable on smartphones because of its dependency on additional hardware followed by extra deployment costs.

Indirect Entry Mechanism. An indirect entry mechanism [11–18] is built with a human-computable challenge-response mechanism. Colors [11, 15], symbols [12, 14] or directions [13, 16–18] are added as additional authentication factors based on a traditional keypad or even the layout of a keypad, as a challenge, which is rebuilt with these factors. Generally in these methods, users use PINs kept in mind to compute the response against the corresponding challenge. The computation always involves collection attribution, color or symbol matching, orientation comparison, table looking up, etc. Users input the response instead of the PIN and it is hard to extract the PIN from the response, which leads to an improvement on security of the mechanism. Additionally, less dependency on auxiliary devices or hardware makes it compatible on smartphones, but additional computation or memory burden for users inevitably reduces its usability.

As an indirect entry mechanism, SlidePIN depends on no additional hardware or devices, which hence avoids extra deployment costs. What is more, with slide input method and a random numeric keypad introduced, SlidePIN accordingly generates the input response when a user slides to unlock. No additional computation or memory burden is imposed to users, which makes security potentially improve as users of SlidePIN slide to unlock their smartphone habitually and conveniently.

3 SlidePIN Concept

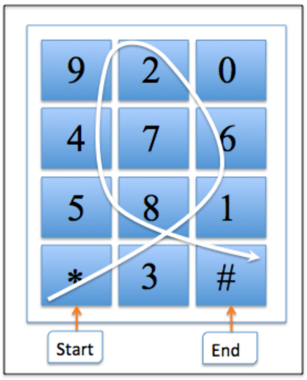


Fig. 1. SlidePIN Concept

Design. SlidePIN (Slide-based PIN entry) is one of sliding PIN entry methods which is inspired by the concept of slide input English words introduced in Word-Gesture Keyboard [4]. Two key mechanisms are introduced into our design compared with 4-digit PIN: *1)* Click input is replaced by slide input. *2)* The fixed layout keypad is substituted by the randomly distributed keypad.

We adopt these two mechanisms mainly because:

1. **Slide input is faster.** It has been proved [20, 21] that click input method is more difficult to use, which suggests that slide input will help improve input efficiency.
2. **Slide input is more secure.** As an indirect entry mechanism, a user's PIN is concealed in a slide sequence, which makes SlidePIN more secure than traditional click input method against shoulder surfing attack.
3. **Input with a random numeric keypad is more secure.** A random numeric keypad helps SlidePIN perform better against replay attack.

Implementation. There are two implementation phases of SlidePIN as follows:

- Setup Phase: Like 4-digit PIN, the user chooses 4 ordered digit numbers as the master secret, usually referred to a PIN, between him/her and the smartphone.

- Unlocking Phase: The user touches and slides over a keypad passing all digits of the PIN in order. As Fig.1 shows, the keypad is a random numeric keypad. In addition, the sliding process should be started from '*' and ended up with '#'. If the slide sequence contains the PIN as its subsequence, the authentication will be passed.

For instance, '1245' as a user's PIN, the user needs to slide and generate a trace starting from '*' and ending with '#' and subsequence '1245' has to be contained in an exact order in the slide sequence. As Fig.1 shows, '*381629458#' is one of the valid slide sequences which can unlock the smartphone.

4 Theoretical Analysis

4.1 Model Definition.

Firstly, we will introduce some concepts.

- **PIN(P):** A PIN is an ordered sequence consisting of integers from 0 to 9. $\{p_1, p_2 ... p_n\}$, $p_i \in [0, 9]$, $i \in [0, n]$. In this paper, we set n=4.
- **Layout(L):** L is the distribution of keys in a keypad as Fig.1. A new layout is generated randomly every time before a user inputs.
- **Trajectory(T):** A trajectory is the trace that is formed when the user slides.
- **Sequence(S):** S represents the keys a user slide over by fingers, which forms an ordered sequence $\{s_1, s_2 ... s_m\}$. S starts from '*', containing PIN (p_1, p_2, p_3, p_4) and inserted numbers $(i_1, i_2, i_3, i_4, i_5\ )$ and then end with '#', as Fig.2 shows.

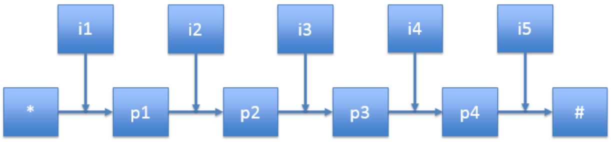


Fig. 2. Model of Input Sequence

- **Slide Map Function** f**:** f defines the process that user creates the sequence by sliding on the layout according to PIN:

$$f(L \times P) \rightarrow S \tag{1}$$

- **Attack Function** f^{-1}**:** f^{-1} defines the reverse process or the attack process based on keypad layouts and trajectories captured by an attacker who aims at obtaining the PIN. Since a specific sequence is determined by a specific random keypad layout and a specific trajectory, the relationship can be described as follows:

$$L \times T \rightarrow S \tag{2}$$

Accordingly, the attack function is defined as:

$$f^{-1}(L \times T) \rightarrow f^{-1}(S) \rightarrow P \tag{3}$$

4.2 Sequence Length Analysis.

The sequence length has a direct influence on security and usability of SlidePIN. Longer slide sequences make users' PIN more secure but the usability is impaired. Shorter ones lead to higher usability yet lower security. Therefore, sequence length analysis needs to be conducted to searching for a good tradeoff between security and usability. First, we estimate the distance between keys of keypad. Afterward, we estimate the distribution of PIN and finally the sequence length.

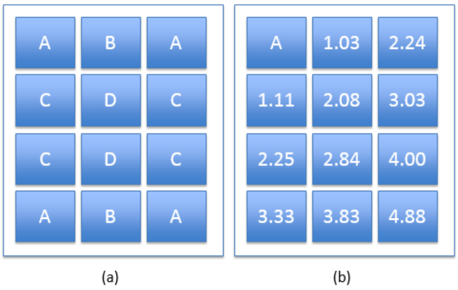


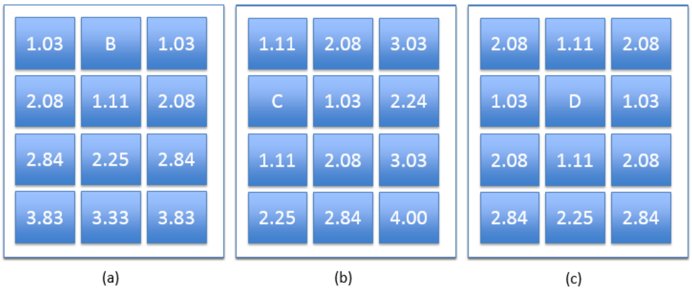
Fig. 3. Category of Keypad and Distances Between Category A and Other Keys

Estimate of Distance between Keys. Keys on a SlidePIN keypad can be divided into four categories (A, B, C, D) and distances between keys in category A and others can be calculated as showed in the Fig.3 [1]. Then the average distance between key A and others is $D(A) = (1.03 + 2.24 + 1.11 + 2.08 + 3.03 + 2.25 + 2.84 + 4.00 + 3.33 + 3.83 + 4.88)/11 \approx 2.78$

Distances between keys in category B, C, D and others, as Fig.4 shows, are D(B)=2.39, D(C)=2.25, D(D)=1.87, which can be calculated using the same method.

As the keypad showed in Fig.1, the sequence in category A is started at * and ended with #. As defined in the model, the average distance is D(A) at position i1 and i5. Other than that, the average distance among 10 numbers is $D_{avg} = (D(A) \times 2 + D(B) \times 2 + D(C) \times 4 + D(D) \times 2)/10 \approx 2.31$, which is the distances at the position i2, i3, i4.

[1] The distances are calculated based on data from sequence length experiments.



(a)

1.03	B	1.03
2.08	1.11	2.08
2.84	2.25	2.84
3.83	3.33	3.83

(b)

1.11	2.08	3.03
C	1.03	2.24
1.11	2.08	3.03
2.25	2.84	4.00

(c)

2.08	1.11	2.08
1.03	D	1.03
2.08	1.11	2.08
2.84	2.25	2.84

Fig. 4. Distances Between Category B,C,D and Other Keys

Estimate of PIN Distribution. the Layout of a SlidePIN keypad can be divided into area Z1, Z2, Z3. According to the method in the previous section, we can separately calculate the average distance when all of the digits in a PIN are in the area of Z1, Z2, Z3 , which is D(Z1)=8.08, D(Z2)=10.82, D(Z3)=11.55.[2]

We could calculate probability of P(Z3)=1, if PIN in the area of Z3. When PIN in the area of Z2, $P(Z2) = \frac{7}{10} \times \frac{6}{9} \times \frac{5}{8} \times \frac{4}{7} = \frac{1}{6}$ and $P(Z1) = \frac{4}{10} \times \frac{3}{9} \times \frac{2}{8} \times \frac{1}{7} = \frac{1}{210}$ when PIN in the third and fourth row except '*' and '#'.

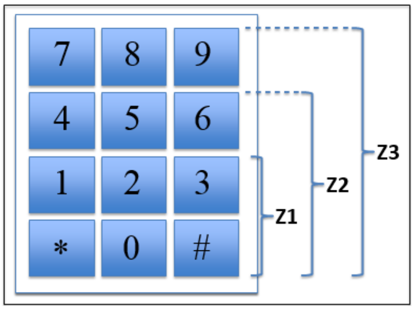


Fig. 5. Distribution of PIN

$Z3'$ refers to the case in which at least one digit of PIN is distributed in the first row. Then the probability of $Z3', P(Z3') = P(Z3) - P(Z2) = 1 - \frac{1}{6} = \frac{5}{6} \approx 0.8333$. Similarly, $Z2'$ represents one or more digits of PIN are distributed in the second row and $P(Z2') = P(Z2) - P(Z1) = \frac{17}{105} \approx 0.1619$. $P(Z1') = P(Z1) = \frac{1}{210} \approx 0.0048$. Therefore, the probability of $Z3', Z2', Z1'$ is approximate to 0.8333, 0.1619, 0.0048.

In case of Z3', the average length of sequence, $D(Z3') = D(Z2) \times P(Z2) + D(Z3') \times P(Z3') = 11.39$. Similarly, D(Z2')=10.52, D(Z1')=8.08.

[2] The calculating process is exemplified in following section, Estimate of Sequence Length.

Estimate of Sequence Length.

Mean Value of Sequence Length. As we have demonstrated above, the average of sequence length when a PIN can be distributed randomly on the entire keypad is $(D(A) \times 2 - 1) + D_{avg} \times 3 = 11.55$.[3]

Lower Threshold of Sequence Length. If the lower threshold is set to 8, approximate to the average length of Z1, there are only $\binom{8}{4} = 70$ possible PINs in an 8-bit-length sequence, making SlidePIN subject to guessing attack. However, longer sequences lead to lower usability and SlidePIN cannot stand the impairment on usability by 10 or more. Thus we set the lower threshold to 9 as a ideal tradeoff with 126 possibilities to resist guessing attack.

Upper Threshold of Sequence Length. Excessively long sequences are vulnerable to replay attack and with lower usability. As to replay attack, the attacker has averagely at least 8 (D(Z1) = 8.08) target characters to input. According to the previous section, a target character generally leads to about 2 input characters (D(A)=2.78, D(B)=2.39, D(C)=2.25, D(D)=1.87). To further improve capability of resisting replay attack, we conservatively use 1.87 as the factor to calculate: $8.08 \times 1.87 \approx 15.11$. Therefore, we set the upper threshold to 15, which is a reasonable upper threshold ensuring both security and usability. An illustration will be provided later in the experiment section.

4.3 Security Analysis

Threat Model. Shoulder surfing attack refers to using direct observation techniques, such as looking over someone's shoulder, to get information. It can also be done by camera (known as video attack) or other vision-enhancing devices. Shoulder surfing attack can be catalyzed as *one-time shoulder surfing attack* and *multi-time shoulder surfing attack*. Excluding threats like phishing attack or malware attack, we only focus on attacks targeting weakness of human-computer interface in this paper.

Besides, we also take guessing attack and replay attack into consideration in this paper. A attacker conducts guessing attack to obtain users' PIN by brute-force attack or dictionary attack based on partial knowledge on users or not. In this paper, replay attack refers to obtaining the a user's complete slide sequence when he successfully unlocks his smartphone and reusing this sequence or one of its subsequences to illegally get access to the user's smartphone.

Guessing Attack. Similar to traditional 4-digit PIN mechanism, a 4-digit PIN is adopted in SlidePIN. Thus both traditional 4-digit PIN mechanism and SlidePIN have the same PIN space, 10000, and the capability of withstanding guessing attack is equivalent.

[3] We excluded the start and the end point when estimating distance between keys.

One-time Shoulder Surfing Attack. Given that the attacker has obtained one valid unlock sequence. Because several invalid numbers are concealed with the user's PIN in a longer slide sequence, the attacker have to take major efforts to extract the PIN. In the worst case, the attacker has successfully conducted one-time shoulder surfing attack and obtained the exact slide sequence and the length of sequence is 9, the lower threshold value, the probability of getting the PIN will be $1/\binom{9}{4} \approx 0.79\%$. However when confronting shoulder surfing attack, input process of traditional 4-digit PIN mechanism or even the PIN is directly exposed to the attacker. Therefore SlidePIN is better at resisting one-time shoulder surfing attack.

Multi-time Shoulder Surfing Attack. Assuming in the worst-case situation, an attacker may capture several valid slide sequences when a user slides to unlock, and PIN can be calculated by using statistical methods. In this paper, we calculate the longest common sequence (LCS) of slide sequences in a simulated attack experiment to validate the security of SlidePIN against multi-time shoulder surfing attack. Longer sequences are more secure against multi-time shoulder surfing attacks. In SlidePIN, the lower threshold of sequence length is set to 9 and due to that, SlidePIN owns capability of resisting such attacks, which will be detailed in the attack experiment.

Replay Attack. In order to resist replay attack, a random numeric keypad is introduced in SlidePIN. Every time before users input PIN, it generates a new layout. So the layout and the sequence are indispensable when replay attack is conducted. Moreover, merely reentering the entire slide sequence with another randomly distributed layout leads a prominent prolonging on slide sequence that will exceed the length limitation discussed in the sequence length analysis and experiment.

Additionally, shoulder surfing attack has negative relevance with replay attack. Longer sequences have stronger capability of resisting shoulder surfing attack but weaker capability against replay attack. Therefore, it is an effective solution and a reasonable tradeoff to set thresholds of sequence length discussed above for SlidePIN.

4.4 Usability Analysis

In this section, usability is evaluated from perspectives of cost of learning, unlock time, orientation time and error rate.

Cost of Learning. SlidePIN is easy to learn and use.

SlidePIN is built based on 4-digit PIN. A SlidePIN keypad is similar to the keypad used in 4-digit PIN. However, clicking and sliding can both be processed in SlidePIN and no mode-switch is demanded. Therefore SlidePIN is easy to learn and provides a smooth transition for users from click input to slide input.

SlidePIN is easy to use. As mobile devices (e.g. smartphones, tablet computers), particularly the ones with touch-screen, rapidly pervade the world, sliding has become one of most common gestures for users. With slide input method, SlidePIN provides an easy and comfortable way for users to unlock their devices.

SlidePIN is interesting to use. Doodling or scrawling being human's inborn preference, it is more interesting to "doodle around" with slide input method than to click fixed buttons mechanically.

Orientation Time. Users need to observe the keypad layout every time before an unlocking movement and the duration of this process is defined as *orientation time.* Since a random numeric keypad is adopted in SlidePIN, orientation time in SlidePIN will be longer than that in traditional 4-digit PIN mechanism.

Unlock Time. Sliding is faster than clicking, however, as a random numeric keypad is introduced and input sequences become longer, unlock time is increased.

Sliding is faster. Clicking input can be considered as a task sequence consisting of multiple single tasks. Each of these tasks can be described by Fitts' Law [20, 21] as entering a single character by clicking. Similarly, slide input in SlidePIN can be regarded as a task sequence including tasks that require sliding over a character. Accot and Zhai have already proved that Fitts' Law is applicable in the case of slide input and sliding across target characters is faster than clicking.[22]

Input Sequences Become Longer. An input sequence contains 6 characters rather than 4, including not only 4-digit PIN but a start point '*' and a end point '#', which partly counteracts the advantage in input speed brought by slide input method.

Random Numeric Keypad Increases Unlock Time. When using a random keypad, some users tend to skip this observation process and input immediately as they get the keypad layout. In this case, it will take them longer to search for next target character after one has already been input.

A random numeric keypad apparently cannot help users form fixed gesture and relevant memory to faster unlock after use experience is accumulated. However, movement that is easy to be transformed into stable memory is subject to attack.

Error Rate. SlidePIN system is more complicated than 4-digit PIN system, which leads to a slight increment on the error rate of unlocking by using SlidePIN. Error rate increases mainly because: 1) Certain range of input sequence length has been set up in SlidePIN. 2) Start point '*' and end point '#' is required during each slide. 3) Participants in our experiments are not familiar enough with this novel mechanism.

5 Experimental Analysis

To evaluate the usability and security of SlidePIN, we recruited 20 students as volunteers to conducted relevant experiments including sequence length experiment, unlock experiment and attack experiment. Based on data obtained from these experiments, we analyzed the reasonable range of slide sequence, distance between two digits of a PIN, orientation time, unlock time, error rate and the capability of SlidePIN against multi-time shoulder surfing attacks.

5.1 Experiment Design

Settings. We designed and implemented an app as experiment software to conduct sequence length experiment and unlock experiment. The app is deployed on a ZTE U930 smartphone (4.2 inches screen, 960 * 540 resolution, Android 4.0).

Volunteers. we recruited 20 students from different majors as volunteers, ranging in age from 18 to 26 (average: 24). 8 were female and 12 were male and all of them have 2 more years experience of using a smartphone and a unlock system.

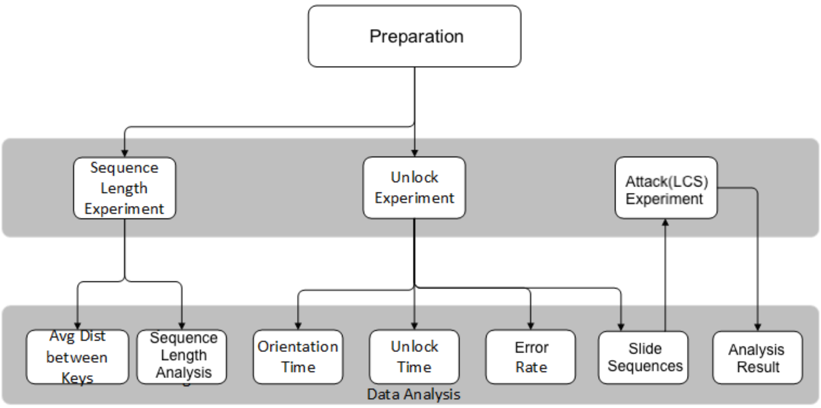


Fig. 6. Experiment Flow Chart

Description.

Preparation.

1. A quick training on how to correctly use SlidePIN is provided to volunteers. They could practise for 5 minutes before experiments formally started.

2. Some essential operation protocols, for example, sliding from '*' and ending with '#', were additionally emphasized.
3. PINs used in the experiments are randomly generated to avoid the negative effects brought by the preference of users.

Sequence Length Experiment Process. As mentioned above, the sequence length has effects on security and usability. So we carried out an experiment to analyze sequence length and distribution.

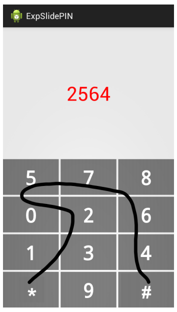


Fig. 7. Sequence Length Experiment UI

In this experiment, according to a 4-digit PIN randomly generated on the screen, each volunteer slid to input the PIN six times without any limitation on slide sequence length, as showed in Fig.7. Due to different keypad layouts generated randomly, one PIN maps to 6 different slide sequences with different sequence length. The slide sequences, sequence length, PINs and layouts are recorded by experiment software in real time for further analysis.

Table 1. Group Setting of Unlock Experiment

Method	Traditional Keypad	Random Keypad
Click	Group 1 (Traditional 4-digit PIN)	Group 2
Slide	Group 3	Group 4 (SlidePIN)

Unlock Experiment Process. In unlock experiment, each volunteer needs to finish inputing each PIN in 4 different groups and for each PIN, 6 more times are conducted. Limitation of sequence was set up ranging from 9 to 15. The slide sequences beyond this limitation were identified as a invalid slide sequence leading to a failure unlocking.

A 2-second countdown is set before keypad appears. As soon as the countdown ends, a timer will start to record *orientation time* till volunteers touch the keypad and begin to input.

The duration from the screen being touched to unlocking being successfully finished would be recorded as *unlock time*. Otherwise, times of failure would be recorded. Application calculates the *error rate* and exits after 6 successful unlocking.

Attack Experiment Process. Based on slide sequences obtained from unlock experiment, attack experiment is conducted to evaluate SlidePIN's security against multi-time shoulder surfing attack. The minimum quantity of slide sequences that can help one get the PIN statistically is evaluated by comparing LCS (Longest Common Subsequence) of them with the PIN.

5.2 Experiment Analysis.

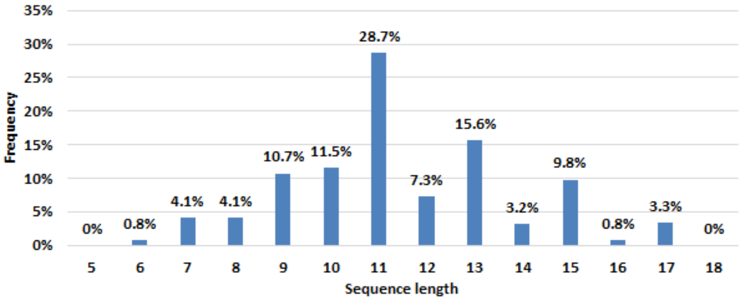


Fig. 8. Distribution of Sequence Length

Sequence Length Experiment. Based on statistical calculation, the average of sequence length is 11.46, approximate to the theoretical value 11.55. Meanwhile, according to the previous section, the average distance between keys is 2.49 and the standard deviation is 1.16.

As Fig.8 shows, 9.0% of sequences are shorter than 9 (lower threshold), which means that setting lower threshold to 9 leads to a loss on usability.

Additionally, all sequences are equal or shorter than 17. Only 4.1% are longer than 15 (upper threshold), while sequences with 15 has reached 9.8%. So it's reasonable to set the upper threshold 15 to improve usability.

Based on theoretical and experimental analysis, it is a good tradeoff to set 9 as the lower threshold and 15 as the upper threshold. Nearly 13.1% of valid sequences are excluded in this case, which compromised usability a little, however, the security is greatly improved.

Unlock Experiment.

Orientation Time. We collect 4 groups of data about orientation time from unlock experiment and put corresponding points on both sides of a time axis. Then a diagram (Fig.9) describing distribution of orientation time is generated. On each axis, ignoring the points with large error by setting up a threshold value marked on the axis leads to a result of higher accuracy.

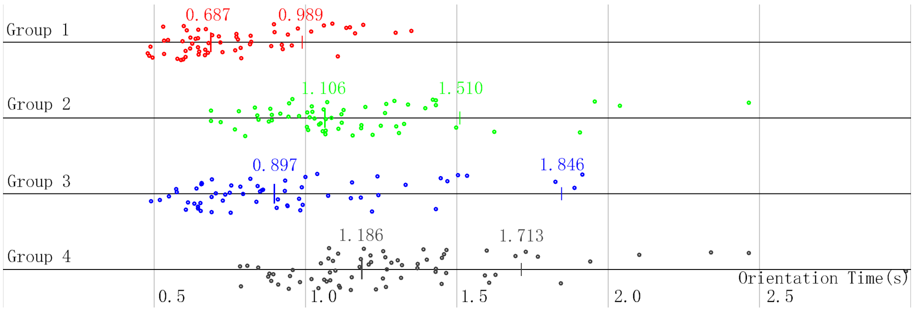


Fig. 9. Orientation Time

According to Fig.9 and Table 2, average orientation time in Group 2 and 4 is longer than in Group 1 and 3, which demonstrates that random keypad increases orientation time. However, orientation time in click input is approximate to that in slide input, which means slide input method has almost no effect on orientation time. Finally, SlidePIN has limited impact on orientation time, compared with 4-digit PIN mechanism.

Table 2. Orientation Time (s)

Groups	Average	Standard Deviation	Threshold Value
1	0.687	0.133	0.989
2	1.064	0.199	1.510
3	0.798	0.293	1.846
4	1.186	0.225	1.713

Unlock Time. According to Table 2, we can conclude that ”random keypad” leads to additional unlock time. Volunteers are more familiar with traditional 4-digit PIN mechanism. Besides, more digits, start point ’*’ and end point ’#’ included, need to be input in SlidePIN. Consequently, the slide input method is not the main reason that increases the unlock time.

The average distance of two numbers to input is 2.49, we can find that average elapsed time between two input in Group 4 (SlidePIN) is longer than that in Group 1 (4-digit PIN) by only about 0.14 second. Even if target characters have reached 6 after bring in start point and end point, the average unlock time of

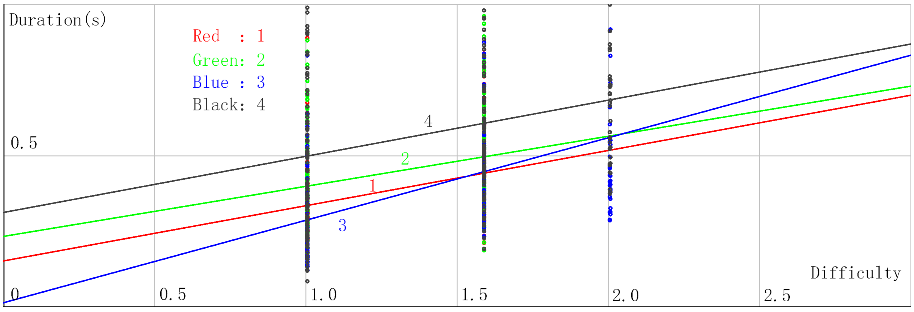


Fig. 10. Experiment Result of Unlocking Time with Fitts' Law

SlidePIN, 3.552 seconds[4], is longer than that of traditional 4-digit PIN, 1.597 seconds, by less than 2 seconds according to calculation based on data collected in this experiment.

Table 3. Regression Equation on Sliding

Groups	Regression Equation
1	T = 0.182*ID + 0.153, a = 0.153, b=0.182
2	T = 0.165*ID + 0.234, a = 0.234, b=0.165
3	T = 0.272*ID + 0.015, a = 0.015, b=0.272
4	T = 0.185*ID + 0.314, a = 0.314, b=0.185

Error Rate. Calculating the error rate, we can conclude that SlidePIN has higher error rate than traditional 4-digit PIN and that both "random keypad" and "slide input" separately have negative effects on error rate.

Table 4. Error Rate

Groups	Group 1	Group 2	Group 3	Group 4
Error Rate	1.67%	3.33%	7.69%	13.04%

In Group 4, 69 input samples are valid for evaluating and 9 of them failed, which makes the error rate reach 13.04%. And in all of these 9 samples, the

[4] For consecutive input tasks, input time can be calculated based on

$$T_n = N \times a + b \times k = \sum_{k=1}^{n-1} \log_2\left(\frac{D_{k,k+1}}{W} + 1\right),$$

where W represents the offset on the movement direction, N represents the number of characters and $D_{k,k+1}$ represents the distance between the k-th and the (k+1)-th keys [20] sequence length was less than 9, the lower threshold of sequence length. Based on the sequence experiment described above, length of 13.1% input sequences is beyond the sequence length limitation, which is of high resemblance with error rate. According to this, we believe that it is closely related to length limits of slide sequence and that as the experience of using SlidePIN is accumulated, the error rate will simultaneously increase.

Attack Experiment. We evaluate the SlidePIN's capability of withstanding multi-time shoulder surfing attacks or statistical attacks by calculating the LCS of slide sequences and when the length of LCS is reduced to 4, we believe that the PIN is exposed. In this experiment, we chose the first 10 valid slide sequences from the previous experiment and calculated the LCS of their slide sequences with incremental quantity till the PIN was exposed, as is shown in Table 5:

Table 5. LCS Analysis (”4” means PIN has been exposed)

Volunteer ID	a	b	c	d	e	f	g	h	i	j
LCS length of 2 slide sequences	6	6	6	6	7	6	6	7	6	4
LCS length of 3 slide sequences	5	5	4	4	4	4	4	5	4	
LCS length of 4 slide sequences	4	4						4		

As is showed in Table 5, a PIN was successfully extracted from 2 sequences involved in just one sample and in many cases, 3 or 4 sequences were necessary for extracting the 4-digit PIN. Thus SlidePIN performs better at resisting multi-time shoulder surfing attacks (statistic attacks) than tradition 4-digit PIN mechanism.

6 Discussion

PIN Storage and Input Sequence Verification. Hash encryption methods like MD5 are not feasible in SlidePIN for PIN storage and verification. Considering both the security of users' credentials and the practicability on smartphones, we provide a solution as follows:

PIN Storage: The device ID of a smartphone is chosen as the identity and a secret key is calculated based is as well as the user's PIN, and then the key is stored after encryption.

Verification: The plaintext of the PIN is obtained and then it will be compared with the input sequence. If the PIN is one of subsequences contained in the input sequence, then authentication is passed.

Fixed Start Point and End Point. SlidePIN demands users to slide from ’*’ and end with ’#’, which makes users input 6 target characters. Although slide input costs less time than clicking, additional digits make negative effects on duration of a complete process of unlocking. However, if terminal points ’*’ and

'#' are not demanded, part of users prefer to slide from the first digit of their PIN and to the last one, which leads to an impairment on security of SlidePIN because merely two digits are concealed during a sliding with the first one and last one exposed.

Same Adjacent Digits. PINs with same adjacent digits are supported in SlidePIN. Given that PIN is '1158', the user needs to slide away from the first '1' to others and then slide back to '1' making the input sequence '11'. However, we highly recommend users to use PIN with no same adjacent digits in SlidePIN to improve both security and usability.

Smudge Attack. A smudge attack [19] is a method to discern the PIN or password pattern of a touchscreen device such as a smartphone or tablet computer. SlidePIN adopting a random numeric keypad, different slide tracks and keypad layouts will be generated each time users slide, which makes SlidePIN effectively resist smudge attack.

Attacks Based on Features. Besides unlock sequences captured by shoulder surfing attack, users' precise features (e.g. angles and dwell time of single digit of slide sequences) can be captured and analyzed to attack SlidePIN. For instance, if a user tends to take a sudden turn on a specific character when sliding, it is of high possibility that this character is one digit of the PIN.

Attacks targeting weakness of human-computer interface discussed in this paper can hardly accurately capture those features. We plan to evaluate the how the attacks based on precise features will affect SlidePIN in our future work.

7 Conclusion and Future Work

SlidePIN performs better than 4-digit PIN against shoulder surfing attacks. At the same time, it has acceptable usability.

In the future work, we plan to conduct more research on numeric keypad layouts and using experiences to find out their impacts on usability and security. In addition, we will try to design SlideText based on English letters.

References

1. Goucher W.: Look behind you: the Dangers of Shoulder Surfing. Computer Fraud & Security. 17–20 (2011.11)
2. Schaub, F., Deyhle, R., and Weber, M. Password entry usability and shoulder surfing susceptibility on different smartphone platforms. In Proc. Mobile and Ubiquitous Multimedia (MUM '12), ACM (2012).
3. Montgomery, E.B.: Bringing Manual Input Into the 20th Century: New Keyboard Concepts, Computer (1982), 15(3) 11–18.

4. Shumin, Z. Per Ola, K,. The Word-Gesture Keyboard: Reimagining Keyboard Interaction. Communications of the ACM, 2012(9), 91–101.
5. M. Kumar, T. Garfinkel, D. Boneh, and T. Winograd. Reducing shoulder-surfing by using gaze-based password entry. In Proc. SOUPS07, ACM (2007), 13–19.
6. Sasamoto, H., Christin, N., and Hayashi, E. Undercover: authentication usable in front of prying eyes. In Proc. CHI08 (2008), 183–192.
7. De Luca, A., von Zezschwitz, E., and Hussmann, H. Vibrapass: secure authentication based on shared lies. In Proc CHI09 (2009), 913–916.
8. De Luca, A., von Zezschwitz, E., Dieu Huong Nguyen,N.,Maurer,M., Rubegni,E., Paolo Scipioni, M.,Langheinrich,M.,Back-of-Device Authentication on Smartphones. In Proc CHI13 (2013), 2389–2398.
9. Azenkot, S., Rector, K., Ladner, R., and Wobbrock, J. Passchords: secure multi-touch authentication for blind people. In Proc. ASSETS 12, ACM (2012), 159–166.
10. Andrea B., Ian O., Dong-Soo K., Bianchi Open Sesame Design Guidelines For Invisible Passwords. Computer (2012), 58–65.
11. V. Roth, K. Richter, and R. Freidinger. A pin-entry method resilient against shoulder surfing. In Proc. CHI 04 (2004), 236–245.
12. Tan, D. S., Keyani, P., and Czerwinski, M. Spy- resistant keyboard: more secure password entry on public touch screen displays. In Proc. OzCHI05 2005, 1–10.
13. Wiedenbeck, S., Waters, J., Sobrado, L., and Birget, J.-C. Design and evaluation of a shoulder-surfing resistant graphical password scheme. In Proc. AVI 06 (2006), 177–184.
14. M. Lei, Y. Xiao, S. Vrbsky, C.-C. Li, and L. Liu. A virtual password scheme to protect passwords. In Proc. ICC08 (2008), 1536–1540.
15. De Luca A, Hertzschuch K, Hussmann H. ColorPIN: securing PIN entry through indirect input. In Proc CHI10, ACM(2010), 1103–1106.
16. Takada, T. FakePointer: An Authentication Scheme for Improving Security against Peeping Attacks Using Video Cameras. In Proc. UBICOMM '08, 2008, 395–400.
17. Weiss, R., and De Luca, A. Passshapes - utilizing stroke based authentication to increase password memorability. In Proc. NordiCHI08, ACM (2008), 383–392.
18. Kim, D., Dunphy, P., Briggs, P., Hook, J., Nicholson, J.,Nicholson, J., and Olivier, P. Multi-touch authentication on tabletops. In Proc. CHI10, ACM (2010), 1093–1102.
19. Aviv, A., Gibson, K., Mossop, E., Blaze, M., and Smith, J.: Smudge attacks on smartphone touch screens. In Proc. USENIX 2010, USENIX Association (2010), 1–7.
20. Bi X, Li Y, Zhai S. Fitts law: Modeling finger touch with Fitts' law; proceedings of the Proceedings of the 2013 ACM annual conference on Human factors in computing systems, F, 2013 [C]. ACM.
21. Fitts P M. The information capacity of the human motor system in controlling the amplitude of movement [J]. Journal of experimental psychology, 1954, 47(6): 381–391.
22. Johnny Accot, Shumin Zhai : Performance evaluation of input devices in trajectory-based tasks: An application of the steering law. In Proceedings of ACM CHI 1999 Conference on Human Factors in Computing Systems(1999), 466–472.