Hindawi Publishing Corporation
Advances in Human-Computer Interaction
Volume 2016, Article ID 8762892, 11 pages
http://dx.doi.org/10.1155/2016/8762892

Research Article

Extending the Touchscreen Pattern Lock Mechanism with Duplicated and Temporal Codes

Ashley Colley,[1] Tobias Seitz,[2] Tuomas Lappalainen,[1] Matthias Kranz,[3] and Jonna Häkkilä[1]

[1]*University of Lapland, 96300 Rovaniemi, Finland*
[2]*LMU Munich, Munich, Germany*
[3]*University of Passau, Passau, Germany*

Correspondence should be addressed to Ashley Colley; ashley.colley@gmail.com

Received 3 August 2016; Accepted 1 November 2016

Academic Editor: Thomas Mandl

We investigate improvements to authentication on mobile touchscreen phones and present a novel extension to the widely used touchscreen pattern lock mechanism. Our solution allows including nodes in the grid multiple times, which enhances the resilience to smudge and other forms of attack. For example, for a smudge pattern covering 7 nodes, our approach increases the amount of possible lock patterns by a factor of 15 times. Our concept was implemented and evaluated in a laboratory user test ($n = 36$). The test participants found the usability of the proposed concept to be equal to that of the baseline pattern lock mechanism but considered it more secure. Our solution is fully backwards-compatible with the current baseline pattern lock mechanism, hence enabling easy adoption whilst providing higher security at a comparable level of usability.

1. Introduction

For an authentication scheme, the balance between its ease of use and its security is a critical factor determining its suitability for a particular application. Smartphones hold a large amount of private information, from personal photographs, to text messages, email, social media, and the possible access to the user's finances. Even considering that physical access to the device is needed to operate the device lock mechanism, ignoring remote vulnerabilities, the need for a secure lock mechanism is clear. This paper focuses on lock mechanisms for mobile devices, particularly touchscreen smartphones. Research in the domain of usable security in general [1] acknowledges that there is a tension between security, user needs, and acceptance of these mechanisms and suggests design guidelines. One recommendation is to use the "path of least resistance," that is, to match the most comfortable way to do tasks.

The typical usage context of smartphones has the user unlock their phones many times a day. Harbach et al. [2] report a daily average of 47.8 unlocks. Often this unlocking occurs in situations where the user is physically encumbered or cognitively loaded with other tasks; thus one-handed use, as shown in Figure 4, is desirable. In fact, in most cases the unlocking action is an annoying precursor to the user's actual goals. This, along with many other factors, limits acceptance of complex yet secure locking mechanisms [3, 4]. Koved et al. [5] state that, "When end-users' perceptions of risk are not aligned with those on which the system is based, there is a mismatch in perceived benefit, leading to poor user acceptance of the technology." In practice the situation may be somewhat more complex, as users' perceptions of the security provided by a password mechanism may be somewhat different from the actual level provided [6].

Though, it is generally accepted that for most users raising awareness on security, especially on mobile devices, is a challenging, yet important task [7]. Put simply, if an action is too much effort for the expected outcome, acceptance is low [4]. Asking for usable and secure authentication seems, as De Luca and Lindqvist [8] state, too much. We therefore argue that a minimal extension to a well-accepted method could increase security without lowering perceived usability.

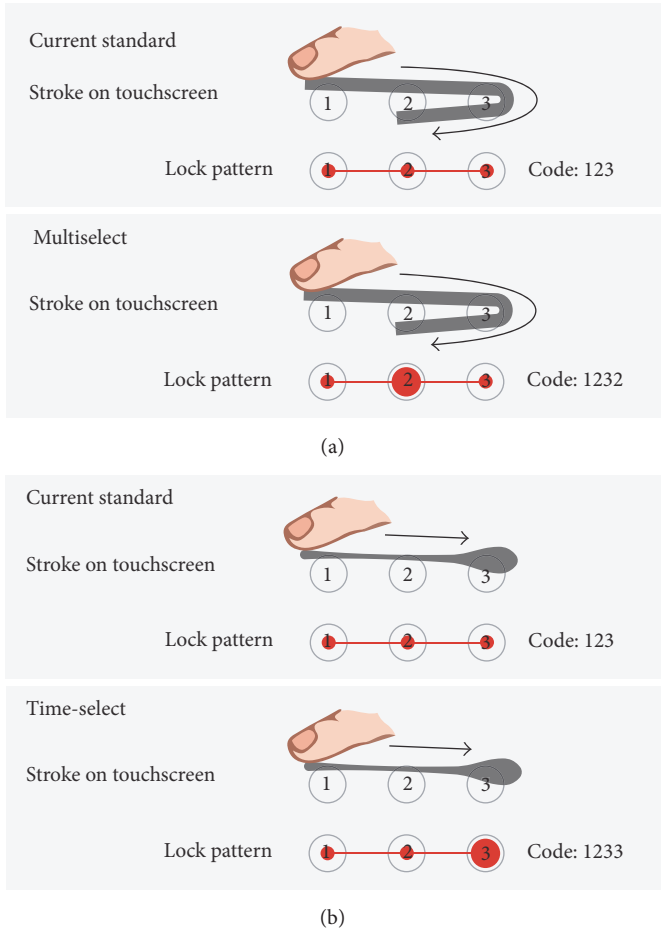

Figure 1: Pair-wise comparison of the proposed lock mechanisms to the standard pattern lock. In each illustration, the unlock action starts at node "1" and ends at the large red dot. (a) In the multiselect condition, the "2" is added to the pattern by moving over a previously selected node. (b) In the time-select condition, a second "3" is added to the pattern. "Long-press" allows the current node to be reselected and added to the pattern.

Prior work both evaluates the performance of pattern lock mechanisms and proposes improved or alternative locking procedures. The acceptance and performance of a locking mechanism are highly sensitive to small conceptual changes, if we consider the high number of unlocks that users perform each day. The unlocks often occur in split-attention scenarios or second-task conditions, for example, when users quickly read an instant message whilst rushing to a train on the way to work. Touchscreen interaction itself is in general subject to a certain level of errors and accidental touches, which will also play a role in any touchscreen based unlocking process [9].

Our concept is illustrated in Figure 1. As the basis of our concept, we extend the pattern lock mechanism by enabling each node in a 3×3 grid to be used multiple times, including the repetition of a node directly after it has been used. This enhances the current baseline pattern lock mechanism that allows each node to be included in the pattern only once. We therefore propose a subtle variation in the task, which introduces a significant conceptual change. In contrast to other variations of stroke-based passwords, we specifically address one-handed input, acknowledging the context of a primary task as trigger for the unlock action.

In our research we implement and evaluate improvements to the pattern lock mechanism that aim to increase its resilience to a variety of attack methods, whilst at the same time maintaining the full usability of the baseline method. Our particular focus is on evaluating the method in a typical usage context. We specifically,

(i) introduce novel enhancements to the current pattern lock mechanism; it becomes more resilient particularly against smudge attacks, whilst fully preserving the usability benefits of the current mechanism, for example, concerning one-handed use; additionally, the method provides a fully backwards-compatible user experience with the standard pattern lock procedure, requiring only little learning or adaption from the user;

(ii) evaluate the enhanced mechanism in a user study, focusing on both the advances in usability and the perception of security improvements. We also explore

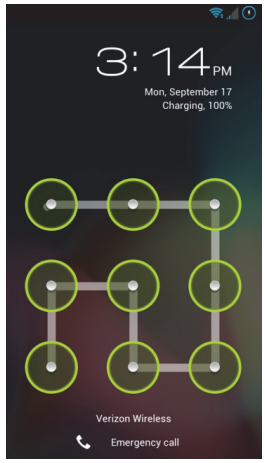

Figure 2: The default Android grid lock, showing a lock pattern for a 3 × 3 grid.

if this is achieved by the actual user codes within the scheme.

The paper is structured as follows. We first give an overview on the state of the art on lock mechanisms and then relate our approach to other touch-based, drawmetric approaches. We then describe the experimental setup and the qualitative and quantitative results. We close with a discussion of implications and future lines of work.

2. State of the Art and Related Work

We begin with a short summary of the state of the art on lock mechanisms in general and stroke-based unlocking in specific. Also, we address potential attacks for these methods. We then discuss and relate to existing work on the design space of pattern lock mechanisms to situate our proposed extension.

2.1. Unlock Mechanisms. An overview of the wider area of graphical password mechanisms is provided by Biddle et al. [11]. The design space for graphical passwords is described by Schaub et al. [12], who evaluate the key parameters of several approaches and offer guidelines for the designers of such mechanisms.

A variety of unlocking mechanisms exist for granting access to smartphones. This includes PIN code entry, textual passwords, action-based unlocking, such as tapping [13], shaking [14, 15], or transferring the lock-state from another device [16], and biometric authentication such as fingerprint or facial recognition [17]. Biometric methods especially lack widespread acceptance amongst users due to various reasons, such as the inability to change biometric attributes [18]. We therefore especially aimed to sustain acceptance whilst increasing the security of the lock mechanism.

The pattern lock mechanism (see Figure 2), which we focus our work on, is a so-called graphical password. It was amongst the first mechanisms to substitute the PIN entry, offering an acceptable balance between security and usability [2]. In Figure 2, the standard grid consists of 3×3 nodes ($n \times m$ in the general case) on which the user makes a single stroke gesture starting from one node and passing through a number of other nodes. Each node can be included only once, and hence the maximum pattern length includes 9 nodes. Lifting the finger from the screen completes the stroke gesture and enters the passcode, without the need for further interaction. It should be noted that although it is possible to pass over a node several times, it is only included in the pattern the first time it is passed over. For example, a pattern may start on node 2 (top row, middle), move to node 3 (top row, right), and then end on node 1 (top row, left), passing over the previously selected node 2 in the process. In this case the entered code will be "2 3 1".

The mean unlock time for the standard Android pattern lock has been measured as 1.4 seconds, for user selected patterns [19]. Studies considering its usability are generally rather limited; in particular we were unable to find detailed studies that evaluated the mechanism in one-handed use or real-world usage contexts. It may be noted that in many of the concepts presented as improvements to the baseline pattern lock, relatively high unlock times have been reported. For example, Chiang and Chiasson [20] report 15 seconds and De Luca et al. [21] ≤ 4 seconds. It may be speculated that, for current users of the standard pattern lock mechanism, such increases would make migration to the proposed concepts unlikely.

2.2. Potential Attacks. Any specific lock mechanism, or combination thereof, has its specific attack vector. Basic attacks such as brute force and dictionary based guessing, where a list of higher probability passwords is used by attackers to reduce the number of attempts needed to guess a password, can also be applied to lock patterns. Brute force exhaustive search, dictionary-based explorations, overlooking, social engineering, and recovery from postinspection have been presented and partially are applicable to pattern lock, too. Whilst the theoretic "password" space for lock patterns might appear larger than a 4-digit PIN, this does not withstand reality checks [12]. Depending on restrictions (e.g., can a node appear twice in a pattern), the array (e.g., $m \times n$ dots), and sociocultural aspects (e.g., starting at the top left for many western users), the actual password might be weak, like "0000" as a PIN number [22]. Moreover, Biddle et al. [11] discuss 25 mechanisms concluding that graphical passwords are in general more vulnerable to shoulder surfing attacks than alphanumeric passwords. Additionally, they point out that many of the reviewed graphical password systems lack rigorous evaluation in security and usability. In addition to shoulder surfing attacks [21, 23], the pattern lock mechanism is vulnerable to smudge based attacks [19, 24–26], where physical residues from finger strokes on the touchscreen provide attackers with clues as to the password.

The theoretical password space refers to the total set of all possible password combinations which can be produced by a password scheme and hence the inherent strength of the scheme. The total is reported in binary as the number of bits. The password space of the standard Android 3 × 3

grid lock is 18 bits [24], which is approximately equal to a 5-digit PIN code. Based on a simulated study, average lengths of pattern lock codes were found to be between 6.19 and 6.64 nodes, for "easy-to-remember" and "hard-to-guess" patterns, respectively [10].

Several authors have made detailed studies of the variation of patterns used by users [10, 27, 28], concluding that the variety of patterns is relatively limited. Here, for example, studies have reported that many users started their patterns from the top left node, Andriotis et al. [10] reporting 52% and, in a paper based study, Uellenbeck et al. [28] 44%. This limited range of used codes makes this lock mechanism susceptible to dictionary based guessing attacks. Following up this work, Aviv and Fichter [27] identify particular password pattern elements that users perceived as contributing to create a high security password.

There is a large body of related work motivated by the susceptibility of the standard pattern lock mechanism to smudge attacks, for example, [10, 19, 24, 26, 27]. Various approaches to overcome this problem have been explored, an overview of which is presented later in this section.

2.3. Design Space for Pattern Lock Mechanisms. We discuss selected extensions and variations of the pattern lock mechanism, each addressing specific limitations thereof with the overall intention to increase the security of the pattern.

2.3.1. Strength Meters. Based on their analysis of perceived and actual pattern strength, Andriotis et al. [22] and Sun et al. [29] investigated the display of a strength meter, known from text-based passwords, and its effect on the passwords selected by users. Here, the users increased the complexity of their patterns when the strength meter was present, hence leading to an increased level of security.

2.3.2. Alternative Node Patterns. A rather straightforward approach is simply to increase the number of nodes used in the scheme; for example, Chiang and Chiasson [20] used a 5×7 grid as part of the solution they evaluated. However, this increased density may lead to a reduction in the one-handed usability of the unlocking mechanism.

In their tiny lock concept Kwon and Na [19] present a minimally sized 3×3 grid input area and a larger copy of the grid that provides visual feedback. When combined with a final circular unlocking stroke that creates a masking smudge, Kwon and Na report that an attacker was unable to deduce any passwords from the smudge patterns on the screen. However, we assume that the increased visual feedback makes the system more susceptible to shoulder surfing attacks.

With the aim of providing a wider range of user-defined password patterns, in their paper based study Uellenbeck et al. [28] evaluated a wide variety of alternative node grid layouts, for example, a 3×3 square layout with the top left node missing, as well as circular patterns. Shin et al. [30] also briefly introduced a circular pattern arrangement of 6 nodes. In this scheme, each node may be used up to 7 times, changing color at each reuse. Although this has some similarities with our concept regarding reselection, few details of the implementation are given and no user evaluation is presented.

2.3.3. Moving Patterns. Concepts where the position of the pattern grid is translated to a different position, scale, and rotation on the screen for each unlock attempt have been explored by Schneegass et al. [25] and von Zezschwitz et al. [26]. However, they reported that users experienced difficulty in locating the grid, due to its varying location.

2.3.4. Rhythm-Based Approaches. To address the smudge and shoulder surfing attack problems Kim et al. [31] and Lee et al. [32] propose mechanisms based only on the amount of time the finger is in contact with the touchscreen. For example, code entry may consist of a rhythmic sequence of touch events lasting 3 sec, 2 sec, 3 sec, and 1 sec, followed by pressing an enter button to enter the code. Here, the taps may be made at any position on the device screen, for example, at a single position. Whilst this solution clearly solves the smudge issue and may offer advantages in terms of semiblind use, its practical usability has not been established as currently no user evaluation has been presented. Somewhat similar solutions have been proposed using the accelerometer contained in the device to recognize a sequence of rhythmic taps on the device body [1]. The rhythm-based enhancement that we put forward offers similar security and usability whilst being resilient against shoulder surfing attacks.

2.3.5. Other Approaches. Chiang and Chiasson [20] present a multilayered drawing lock mechanism. Here warp cells at the corners of the grid enable more complex patterns by using multiple layers. For example, when a warp cell is touched as part of pattern entry, a second empty grid layer is displayed obscuring the original grid layer, on which the pattern entry can continue. When evaluated in a comparative user study, Chiang and Chiasson [20] conclude that their mechanism outperforms the "Draw a Secret" lock mechanism. However, as earlier noted, the density of the 5×7 grid pattern and unlock times of 15 to 18 seconds make its usage in realistic one-handed contexts questionable.

Acknowledging the users' reluctance to check for threats, Riedl et al. [33] propose to have different zones on their mobile phone. Each zone is basically equivalent to a virtual machine that is separated from the others. This approach divides everyday activities, such as surfing the web, from sensitive activities, such as home banking. As a response to shoulder surfing attacks De Luca et al. [13] add a touch panel on the back of smartphone, such that the user can split their unlocking gestures between the two sides of the device. Here, unlock times ≤ 4 seconds are reported.

Recently Apple [34, 35] has disclosed patent applications for pattern lock approaches. Ideas include changing the color and length of the visible finger trace depending on the speed, duration, and complexity of the gesture, inclusion of a strength indicator, and making certain nodes visible only after other nodes are touched. The latter concept perhaps have similarity to Chiang and Chiasson's warp cells [20].

A promising approach to protect smartphones against most attacks is the usage of behavioral biometrics. Here, an additional layer of security during authentication is added. Users show individual differences in how they enter their patterns such as speed of entry or size of the finger contact area on the screen. De Luca et al. were one of the first to put forward the idea of implicit authentication in the realm of lock patterns [36]. With two user studies, they provided evidence that distinguishing users by behavioral biometrics is feasible and significantly adds to security. Extending this idea, multimodal approaches for authentication are on the rise. For example, Google has been working on Project Abacus that targets eliminating the need for explicit authentication (https://goo.gl/G0K2bu, last accessed July 10, 2016). Moreover, as several smartphone devices on the market have introduced force sensing touchscreens, we assume that future research will investigate the performance of force touch, providing another dimension to a pattern without requiring an extension in length.

3. Concept and Implementation

We chose Google's Android platform as basis for our research. The current Android grid-based pattern lock allows each node in the pattern to be selected only once. To extend the mechanism, we introduce two solutions where nodes may be selected multiple times.

3.1. Concept. In our approach this multiple selection may be achieved in one of two ways.

(i) *Sequential Duplication*. When drawing the pattern, the path may go back over nodes that are already included. In this paper we refer to this as "multiselect."

(ii) *Time-Based Duplication*. When a user drawing the lock pattern pauses on a node for more than a threshold time, that node is again entered into the pattern sequence (cf. key repeat on keyboards). In this paper we refer to this as "time-select." In our initial implementation a time threshold of 600 ms was used.

By enabling node duplication in lock patterns we address the susceptibility to smudge and shoulder surfing attacks, which has been reported as one of the core security problems of the basic pattern lock mechanism. Additionally, we extend the possible password code length beyond the current maximum length of 9, providing more secure passwords to those users that require it. Figure 3 shows an example where the introduction of a single reselected or duplicated node increases the number of combinations that a smudge attacker would need to try from 2 to 30 (2 baseline patterns, plus 14 multiselect patterns, plus 14 time-select patterns), that is, a factor of 15 times. When multiple reselections and duplications are considered, this results in a many fold increase of the amount of permutations an attacker would need to try, essentially rendering smudge attacks ineffective.

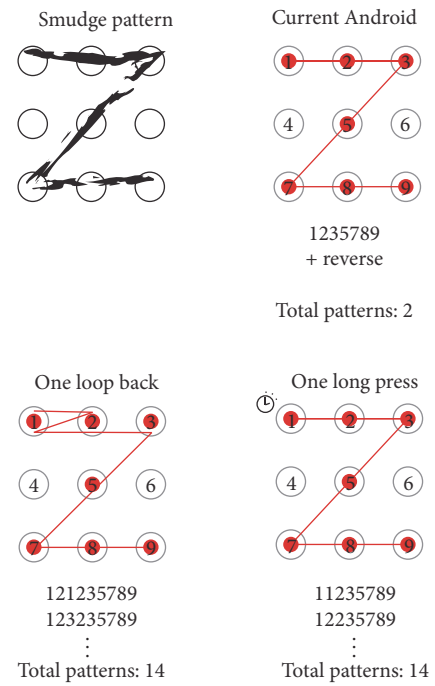

Figure 3: Additional lock code ambiguity is introduced by a single reselected or duplicated node selection based on a start smudge pattern. The red line visualizes the lock path defining the respective pattern.

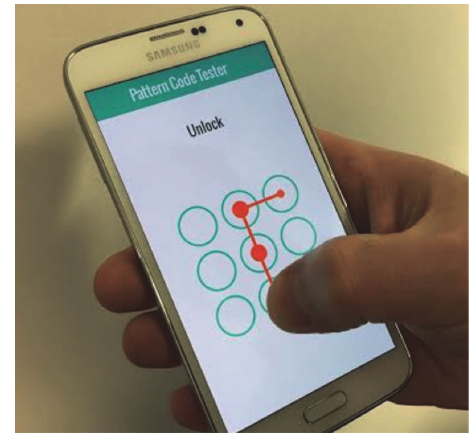

Figure 4: Our Android prototype allows users to experiment with the proposed lock mechanism. The smaller red dot (top right node) has been selected once, whereas the larger red dots (center column) have been selected twice.

A key tenet of our solution is that it is fully backwards-compatible with the existing Android pattern lock mechanism. It retains the basic 9-node grid presentation and users who do not wish to take the additional features into use can continue to use their existing patterns. In contrast to other proposed enhancements to pattern lock, such as relocating or resizing the grid, our approach maintains the

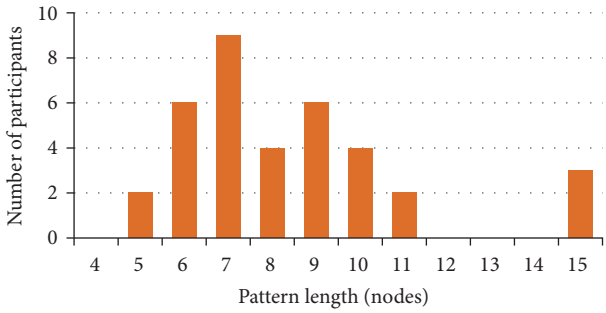

Figure 5: Length of lock patterns created by participants using the enhanced lock mechanism.

well-accepted aspects in pattern lock mechanisms. This is in line with our goal of providing usable security with as little as possible additional burden or perceived effort on the user side. Moreover, our solution maintains the simplicity of the basic lock mechanisms, supporting passwords of various lengths, without the need to confirm the entry by clicking a separate enter key.

Although our solution is extensible to larger node patterns beyond the 3×3 grid, we aim to retain this configuration. We speculate that increasing the grid size to 4×4 and above creates an increase in the required input accuracy that reduces the usability of the mechanism in realistic contexts.

3.2. Implementation. We implemented our extended pattern lock concept as a standalone Android application. The application allowed enabling both the sequential and time-based node duplication individually or in combination. The prototype enabled multiple reselections to be made in a pattern; that is, several nodes in the pattern can be selected more than once. In our implementation each node could be reselected a maximum of 3 times, each reselection being visually indicated with an increased size dot. The application included 3 modes, free interaction, set code, and unlock.

Essential aspects of our implementation are the visualization applied to duplicated node entry and haptic feedback. As we aim to allow semiblind usage, that is, without continuously looking at the device display during the entry, the purpose of the visualization is predominantly to support users' learning the mechanism. Thus we increased the size of the dots with the number of times that it had been included in the entered pattern; see Figure 5. Similarly to the current Android lock pattern, when each node was selected a pulse of vibration feedback was given using the device's inbuilt vibration mechanism. This was also identical in the case where a node was entered based on time delay; that is, when selecting the node for the first time by moving over it, a vibra pulse was given, and then if the user had not moved from the same target within the time entry window, a second vibra pulse was given and the node entered to the pattern for a second time. The application included detailed interaction logging, such that the motion and duration of each lock or unlock interaction stroke were logged to a text file stored on the device.

3.3. Research Questions. As an overall target, we aim to ascertain if users would actually like to take our extended pattern lock mechanism into use in practice, replacing their currently used lock mechanism. More specifically, we aim to answer the following research questions:

(RQ1) Are users able to understand and successfully use a pattern lock mechanism extended with multi- and time-select features?

(RQ2) Are the multi- and time-select features perceived as more secure by users?

(RQ3) Do the multi- and time-select features fully maintain the usability benefits of the baseline pattern lock mechanism, for example, speed of use, regarding memorability and one-handed and semiblind usage?

4. User Study and System Evaluation

4.1. Study Design. The test process included the following phases, adding up to about 20 minutes per participant:

(1) Signing a consent form and completing a background questionnaire, including information on their motivation for using a locking mechanism and experiences with their currently used lock mechanism

(2) Familiarization with using each of the three features: normal node selection, multiselect, and time-select. Each feature was demonstrated to the participant in turn, after which they experimented with each by setting and using 2 lock patterns utilizing that feature

(3) Creating a personal lock pattern that they would use in practice: setting it and unlocking the device with it 4 times

(4) Completing a final exit questionnaire: this probed participants' reasoning for the choices they had made in selecting their pattern

(5) Returning after approximately one hour to unlock the device using the pattern they defined earlier: this phase was completed only for a subset of participants, due to participants' availability.

For the user study a Moto G smartphone running Android 5.0 was used as the test device. Tests were conducted with the participants standing, and they were instructed to hold and interact with the device as they would when typically unlocking a smartphone. The test moderator noted how the device was held and interacted with.

4.1.1. Participants. We recruited 36 participants (20 females, 16 males), having a mean age of 35 years (SD = 14). Of the participants, 6 were left-handed. The participants were randomly recruited at the university campus by personal invitation and were compensated with a gift of a small candy bar. Participants were informed about the scope of the study and consent for participation was obtained.

Regarding smartphone usage, 35/36 participants owned a smartphone. They reported to have had it for 1.8 years on

average (SD = 1.1). About half (15/36) used Android devices, whilst others used iOS (9/36), Windows Phone (8/36), Jolla (2/36), and Bada (1/36). The security mechanism currently used by most users was PIN-based authentication (12/36), followed by Android pattern lock (7/36) and fingerprint based authentication (4/36). Nine of the participants (9/36) did not use any lock mechanism, considering they did not have any secure data on their device or rely on physical security.

4.1.2. Reasons for Using Current Lock Mechanism. Inquiring the reason for selecting their current lock mechanism, 9/36 mentioned that it was the default mechanism on their smartphone, and they had not considered alternatives. Speed and ease of use were the main drivers, being mentioned by 5/36 and 6/36 participants, respectively. Related to this 2/36 participants commented that they were simply too lazy to use a lock mechanism. One participant noted ease of one-handed use as a particular requirement.

Ease of memorizing was mentioned by 2/36, here the participants noting that they used the same 4-digit PIN also for other systems. Four participants praised the biometric fingerprint lock mechanism they were using for its lack of need to remember anything. The security of the used mechanism was only mentioned as a consideration by 2/36 participants. In this respect one participant commented that he relied on the physical security of his device and thus did not see the need for additional security via an on-device lock mechanism.

5. Results

5.1. Personalized Lock Pattern. The lock patterns created by the test participants are presented in Table 1. Examining the user-defined patterns created by the participants, the multiselect feature was utilized by 75% of participants and time-select by 56%. All participants chose to use at least one of the extension features and 31% included both multi- and time-select. Interestingly, 31% of participants chose to include a duplicated time-selected node at the end of their patterns.

Of the participants utilizing multiselect, approximately half (48%) used it several times (between 2 and 6 times) in their patterns. This suggests that the penalty for repeated use is relatively low. The repeated use of time-select was less frequent, with 35% of the patterns that utilized it doing so more than once. In this case the 600 ms time penalty is clearly a deterrent from repeated use. Accordingly, the highest usage of time-select in a single pattern was 3 times.

Lock pattern lengths of between 5 and 15 nodes were used with a mean length of 7.5 nodes (SD = 2.6). Patterns of 7 nodes in length were most popular; see Figure 5. Extremely long patterns of more than 11 nodes in length were rare, with only 3 participants creating patterns including 15 nodes.

When describing the reason for creating their lock code, the main drivers mentioned were the following: ease of remembering (42%), security (22%), easy to enter (19%), and fast to enter (17%). Related to the memorability, 22% of participants mentioned that they had based their pattern on a letter or shape, for example, "My middle initial twice" (#16) and "A heart shape, starting from the middle node" (#9).

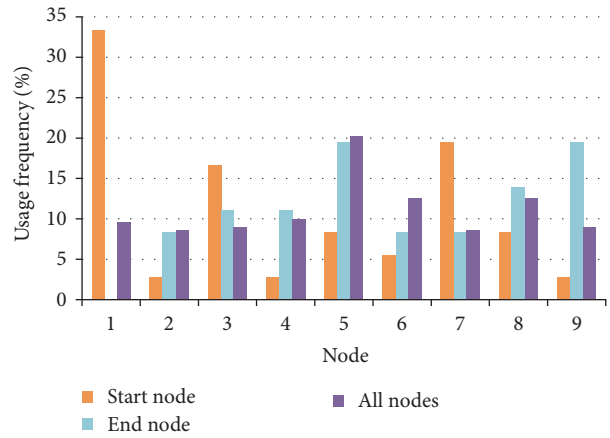

Figure 6: The usage frequency of nodes in participants' selected patterns for the enhanced lock mechanism.

Many participants (36%) commented that the pattern they had selected was, for them, a balance between speed and security: "An easy to remember, simple pattern that is however difficult to hack" (#30) and "The pattern is such that it is not easy to guess, but is quite fast to make" (#14). Interestingly, one participant commented that the speed of entry was one contributor to the resilience to shoulder surfing attacks, "... fast to enter, making it tricky for others to see it." (#22).

Some users (6%) used patterns that were extensions of the current lock pattern they were using, commenting, for example, "The same code I am using in my current device, however I added 2 repeated nodes." (#4) and "As a basis I used the code I use on my own phone, but I took advantage of the extra features in the test phone." (#2).

5.2. Resilience to Guessing Attack. The frequency of node usage in selected lock patterns is one factor affecting the ease with which an attacker can guess lock patterns. For example, with the standard pattern lock mechanism [22] identified that 52% of patterns begin on node 1.

Figure 6 shows the frequency of node usage as the start node and end node and as any node in the lock pattern. It can be noted that the frequency of node 1 as a start node has been markedly reduced compared to the standard lock mechanism case reported in [22], 33% compared to 52%.

To examine if our enhanced mechanism had introduced more variation in the nodes used in the code we followed the approach of Andriotis et al. [10] and calculated Shannon entropy for the codes created by the participants. Table 2 presents the analysis for start nodes, end nodes, and all nodes.

5.3. Unlocking Performance

5.3.1. Initial Unlock. After deciding on a lock pattern and setting it the participants then proceeded to unlock the device 4 times. The mean time for participants to unlock the device

Table 1: Lock codes chosen by test participants and mean unlock times. Node numbering refers to the position in the 3 × 3 grid; left to right, top row: 1, 2, 3, middle row: 4, 5, 6, and bottom row 7, 8, 9.

Participant	Lock code (node number)	Number of nodes	Multiselect	Time-select	Mean unlock time (ms)	Unlock time after one hour (ms)
1	5789512355	10	x	x	2458	2422
2	862112686	9	x	x	2193	
3	75357	5	x		617	
4	965369	6	x		1019	
5	3214566987	10		x	2772	
6	7856322	7		x	2522	
7	74123698755	11	x	x	3079	
8	2547856	7	x		2590	2706
9	5236987455	10	x	x	3090	3488
10	14745258563	11	x		2148	2373
11	86248654	8	x		1414	1546
12	775321599	9	x	x	3992	
13	3545853	7	x		1420	599*
14	753258	6	x		831	
15	6655884	7		x	3362	
16	1123357899	10		x	2225	
17	1144778	7		x	3385	
18	125695	6	x		1739	1488
19	11256988	8		x	3063	
20	123558	6		x	1933	**
21	156248963214789	15	x		2587	2466
22	15987532	8	x		1262	1298
23	744145636965852	15	x	x	4259	3472
24	541258965	9	x		1596	
25	147456369	9	x		1374	
26	321456987789654	15	x	x	4171	
27	369955	6		x	2534	
28	8523654	7	x		1526	
29	36989	5	x		708	743
30	6588966	7	x	x	2405	
31	4585655	7	x	x	1657	
32	368414863	9	x		2316	2493*
33	1245788	7		x	2399	
34	753695147	9	x		1261	
35	114123	6	x	x	2475	
36	14785589	8	x	x	2407	

*At second attempt. **Failed to unlock after 5 attempts.

was 2.2 seconds (SD = 0.9 seconds). To examine if there was any dependency of unlock time on the pattern length, a Pearson correlation coefficient was calculated, returning a value of $R = -0.1144$. This indicates that there is no notable correlation between the number of nodes in the pattern and the time taken to unlock the device.

Participants that used the time-select feature in their lock pattern had somewhat longer lock times (M = 2.8 s, SD = 0.7 s) than participants that did not utilize the time-select feature (M = 1.5 s, SD = 0.6 s). This was expected due to the 600 ms delay required to insert a node in the pattern with the time-select feature.

5.3.2. Unlock after One Hour. Of the participants 13 returned after approximately one hour to unlock the device using the code they had previously set. At this point 10/13 unlocked the device at first attempt, 2/13 unlocked the device at second

Table 2: Shannon entropy for nodes used in the lock pattern. A value of 3.00 indicates that there is no bias towards particular nodes and that all nodes are used equally. ∗ = exact values not provided.

	Standard pattern lock (Andriotis et al. [10])	Enhanced lock mechanism
Start nodes	2.35	2.68
End nodes	3*	2.92
All nodes	3*	2.97

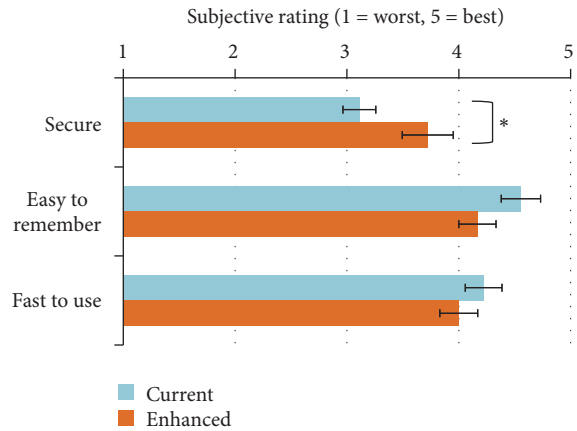

Figure 7: Mean subjective ratings for the user's current and the enhanced lock mechanism. Error bars indicate standard error of mean. ∗ = significant difference $p < .05$.

attempt, and one user was unable to unlock the device within an allowed maximum of 5 attempts.

5.4. Subjective Ratings and Qualitative Comments. The participants' responses to the subjecting rating questions are shown in Figure 7. A Wilcoxon signed rank test indicated that there was a significant difference between the perceived security of the current lock mechanism used by participants and the enhanced lock mechanism ($Z = -2.149$, $p = .032$). There was no significant difference in either the ease of memorability or speed of use between current and enhanced mechanisms ($Z = -1.946$, $p = .052$ and $Z = -1.134$, $p = .297$, resp.).

Considering the use of multiselect and time-select features, on a scale of 1 (would not use) to 5 (would use) participants mean rating for multiselect was 4.0 (SD = 1.1) and for time-select 3.2 (SD = 1.2). Examining the participants' qualitative comments regarding the enhanced mechanism, 11/36 participants highlighted the improved security of the multiselect feature as beneficial. The speed of operation was clearly an important issue with 7/36 participants praising the speed and naturalness of multiselect, whilst 6/36 considered the time-select as slow. Additionally, 9/36 felt the time-select required too much concentration or was too error prone.

Negative comments on the enhanced mechanism were, for example, that it added complexity (mentioned by 2/36 participants) and that biometric based mechanisms were more effective (2/36 participants). Additionally, one participant wished for clearer visualization of the multiple selected nodes.

6. Discussion

6.1. Increment to Pattern Lock. Overall our extensions to the pattern lock mechanism were well received, with all of our test participants being able to use both multiselect and time-select additions without problems. When selecting their own lock all of the participants selected to include at least one of the enhancements to the baseline pattern lock mechanism in their pattern.

Longer lock patterns are more secure against brute force attack; for our enhanced mechanism the mean pattern length used by study participants was 7.5 nodes (SD = 2.6). This compares to the mean pattern length of 6.6 nodes (SD = 1.9) reported by [10], for the standard pattern lock mechanism. Similarly, the larger variation in start node seen in our study, compared to the standard lock mechanism [10], improves resilience against dictionary type attacks. Thus overall our enhanced pattern lock mechanism is more secure than the baseline pattern lock mechanism.

Additionally, it should be noted that the general increase the overall code space and code variation introduced by the enhancements also provides an increase in reliance to attack for those users that do not choose to utilize the extension features in their lock pattern.

6.2. Unlocking Performance

6.2.1. Time to Unlock. The measured mean unlocking time of 2.2 seconds (SD = 0.9) compares favorably to the unlocking times reported for the standard pattern lock mechanism for example, 1.4 seconds [19]. It should be noted that considering the use of the multiselect feature only, that is, excluding the time-select feature, the mean unlock time of feature 1.5 seconds (SD = 0.6) is directly comparable to that reported for the baseline mechanism. Further, the measured unlock times fall well below the values reported by other proposals to improve the pattern lock mechanism [20, 21]. It should also be noted that much of the prior work in the area does not include a user study and thus presents no information on actual usability of the mechanisms.

6.2.2. Memorability. Even though the codes used by many of our study participants were rather long (M = 7.5 nodes, SD = 2.6), largely participants had no difficulty memorizing them. As noted by several of the participants, the possibility to select nodes multiple times gave the possibility to draw letters and shapes, thus creating a memorable lock pattern. This can be compared to normal handwriting, where many letters such as "b," "k," and "p" require passing over the same point more than once.

6.3. Backwards Compatibility. Our results indicate that, if made available as part of the default device lock mechanism, the adoption rate of enhancements would be high. Here, the

fact that it is a backwards-compatible extension to the existing lock mechanism appears to reduce the adoption threshold for many users, for example, participants' comments on using variations of their existing lock pattern. Based on the currently widespread usage of the pattern lock mechanism and the large amount of work that has been motivated by its limitations, we feel that our work has potential to be of direct practical benefit in providing improved security to smartphone users.

6.4. Balancing Security and Usability. Usable security is concerned with providing users with "acceptable" procedures whilst trying to ensure a basic level of security. For pattern lock mechanisms, the topic of smudge attacks received great attention and numerous publications, as detailed in the related work section of this paper.

Although prior research on the area has introduced a wide variety of alternative concepts aiming to address the security limitations of the pattern lock method, these have to date also resulted in somewhat reduced usability. We also note that many of the concepts proposed are either unevaluated in a user study or have not been evaluated considering one-handed or semiblind device usage. Noting that none of the proposals of prior research have yet gained widespread adoption, we hypothesize that one of the reasons is that the proposed solutions, whilst increasing security, result in a decrease in usability compared to the current solution. We contrast this with our approach to provide a moderate increase in resilience to attack, without compromising the usability of the standard solution.

6.5. Limitations. We acknowledge that our work is limited by our small sample size and laboratory setting. However, as the general idea of our concept was immediately understood by the majority of our test participants we believe they were well able to immediately reflect on its usage in everyday in-the-wild contexts. As future work, we intend to conduct the study in a larger context to address effects on the measure variables and statistical effects.

7. Conclusion

We have created a touchscreen locking mechanism that extends the widely used pattern lock mechanism, improving its resilience to attack. In the case of smudge based attacks our approach increases the code space for a particular smudge pattern by a factor of 15 times. Evaluation of our concept in a user test ($n = 36$) revealed that users considered it more secure than their currently used lock mechanism, yet equal in its speed of use and memorability. The mean time taken to unlock the device using the enhanced mechanism was 2.2 seconds (SD = 0.9). For patterns including only the multiselect feature the mean unlock time of 1.5 seconds (SD = 0.6) is equivalent to those reported for the standard pattern lock mechanism.

Competing Interests

The authors declare that they have no competing interests.

Acknowledgments

This research has been supported by a grant from Tekes, the Finnish Funding Agency for Innovation, as part of The Naked Approach, A World without Gadgets programme.

References

[1] D. Marques, T. Guerreiro, L. Duarte, and L. Carrico, "Under the table: tap authentication for smartphones," in *Proceedings of the 27th International British Computer Society Human Computer Interaction Conference: The Internet of Things (BCS-HCI '13)*, article 33, 6 pages, British Computer Society, London, UK, September 2013.

[2] M. Harbach, E. von Zezschwitz, A. Fichtner, A. De Luca, and M. Smith, "It's a hard lock life: a field study of smartphone (un)locking behavior and risk perception," in *Proceedings of the Symposium on Usable Privacy and Security (SOUPS '14)*, pp. 213–230, USENIX Association, Menlo Park, Calif, USA, 2014, https://www.usenix.org/conference/soups2014/proceedings/presentation/harbach.

[3] M. A. Sasse, S. Brostoff, and D. Weirich, "Transforming the 'weakest link'—a human/computer interaction approach to usable and effective security," *BT Technology Journal*, vol. 19, no. 3, pp. 122–131, 2001.

[4] A. Beautement, M. A. Sasse, and M. Wonham, "The compliance budget: managing security behaviour in organisations," in *Proceedings of the ACM Workshop on New Security Paradigms (NSPW '08)*, pp. 47–58, Lake Tahoe, Calif, USA, September 2008.

[5] L. Koved, S. Trewin, C. Swart, K. Singh, P.-C. Cheng, and S. Chari, "Perceived security risks in mobile interaction," in *Proceedings of the Symposium on Usable Privacy and Security (SOUPS '13)*, Newcastle, UK, July 2013.

[6] E. Von Zezschwitz, P. Dunphy, and A. De Luca, "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)*, pp. 261–270, ACM, August 2013.

[7] P. Riedl, R. Mayrhofer, A. Möller et al., "Only play in your comfort zone: interaction methods for improving security awareness on mobile devices," *Personal and Ubiquitous Computing*, vol. 19, no. 5-6, pp. 941–954, 2015.

[8] A. De Luca and J. Lindqvist, "Is secure and usable smartphone authentication asking too much?" *Computer*, vol. 48, no. 5, pp. 64–68, 2015.

[9] J. Matero and A. Colley, "Identifying unintentional touches on handheld touch screen devices," in *Proceedings of the Designing Interactive Systems Conference (DIS '12)*, pp. 506–509, ACM, June 2012.

[10] P. Andriotis, T. Tryfonas, G. Oikonomou, and C. Yildiz, "A pilot study on the security of pattern screen-lock methods and soft side channel attacks," in *Proceedings of the 6th ACM Conference on Security and Privacy in Wireless and Mobile Networks (WiSec '13)*, pp. 1–6, ACM, New York, NY, USA, April 2013.

[11] R. Biddle, S. Chiasson, and P. C. Van Oorschot, "Graphical passwords: learning from the first twelve years," *ACM Computing Surveys*, vol. 44, no. 4, article 19, 2012.

[12] F. Schaub, M. Walch, B. Könings, and M. Weber, "Exploring the design space of graphical passwords on smartphones," in *Proceedings of the 9th Symposium on Usable Privacy and Security (SOUPS '13)*, ACM, July 2013.

[13] A. De Luca, E. von Zezschwitz, N. D. H. Nguyen et al., "Back-of-device authentication on smartphones," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*, pp. 2389–2398, ACM, 2013.

[14] R. Srilekha and D. Jayakumar, "A secure screen lock system for android smart phones using accelerometer sensor," *International Journal of Science Technology & Engineering*, vol. 10, pp. 96–100, 2015.

[15] L. Yang, Y. Guo, X. Ding et al., "Unlocking smart phone through handwaving biometrics," *IEEE Transactions on Mobile Computing*, vol. 14, no. 5, pp. 1044–1055, 2015.

[16] R. D. Findling, M. Muaaz, D. Hintze, and R. Mayrhofer, "ShakeUnlock: securely unlock mobile devices by shaking them together," in *Proceedings of the 12th International Conference on Advances in Mobile Computing and Multimedia (MoMM '14)*, pp. 165–174, ACM, Kaohsiung, Taiwan, December 2014.

[17] S. Chen, A. Pande, and P. Mohapatra, "Sensor-assisted facial recognition: an enhanced biometric authentication system for smartphones," in *Proceedings of the 12th Annual International Conference on Mobile Systems, Applications, and Services (MobiSys '14)*, pp. 109–122, June 2014.

[18] A. Goode, "Bring your own finger—how mobile is bringing biometrics to consumers," *Biometric Technology Today*, vol. 2014, no. 5, pp. 5–9, 2014.

[19] T. Kwon and S. Na, "TinyLock: affordable defense against smudge attacks on smartphone pattern lock systems," *Computers and Security*, vol. 42, pp. 137–150, 2014.

[20] H.-Y. Chiang and S. Chiasson, "Improving user authentication on mobile devices: a touchscreen graphical password," in *Proceedings of the 15th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '13)*, pp. 251–260, ACM, Munich, Germany, August 2013.

[21] A. De Luca, M. Harbach, E. Von Zezschwitz et al., "Now you see me, now you don't: protecting smartphone authentication from shoulder surfers," in *Proceedings of the 32nd Annual ACM Conference on Human Factors in Computing Systems (CHI '14)*, pp. 2937–2946, ACM, Toronto, Canada, May 2014.

[22] P. Andriotis, T. Tryfonas, and G. Oikonomou, "Complexity metrics and user strength perceptions of the pattern-lock graphical authentication method," in *Human Aspects of Information Security, Privacy, and Trust: Second International Conference, HAS 2014, Held as Part of HCI International 2014, Heraklion, Crete, Greece, June 22–27, 2014. Proceedings*, vol. 8533 of *Lecture Notes in Computer Science*, pp. 115–126, Springer, Berlin, Germany, 2014.

[23] M.-K. Lee, "Security notions and advanced method for human shoulder-surfing resistant PIN-entry," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 4, pp. 695–708, 2014.

[24] A. J. Aviv, K. Gibson, E. Mossop, M. Blaze, and J. M. Smith, "Smudge attacks on smartphone touch screens," in *Proceedings of the 4th USENIX Conference on Offensive Technologies (WOOT '10)*, pp. 1–7, 2010.

[25] S. Schneegass, F. Steimle, A. Bulling, F. Alt, and A. Schmidt, "SmudgeSafe: geometric image transformations for smudge-resistant user authentication," in *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp '14)*, pp. 775–786, ACM, Seattle, Wash, USA, September 2014.

[26] E. von Zezschwitz, A. Koslow, A. De Luca, and H. Hussmann, "Making graphic-based authentication secure against smudge attacks," in *Proceedings of the 18th International Conference on Intelligent User Interfaces (IUI '13)*, pp. 277–286, ACM, Santa Monica, Calif, USA, March 2013.

[27] A. J. Aviv and D. Fichter, "Understanding visual perceptions of usability and security of androids graphical password pattern," in *Proceedings of the 30th Annual Computer Security Applications Conference (ACSAC '14)*, pp. 286–295, ACM, New Orleans, La, USA, December 2014.

[28] S. Uellenbeck, M. Dürmuth, C. Wolf, and T. Holz, "Quantifying the security of graphical passwords: the case of android unlock patterns," in *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (CCS '13)*, pp. 161–172, ACM, Berlin, Germany, November 2013.

[29] C. Sun, Y. Wang, and J. Zheng, "Dissecting pattern unlock: the effect of pattern strength meter on pattern selection," *Journal of Information Security and Applications*, vol. 19, no. 4-5, pp. 308–320, 2014.

[30] K. I. Shin, J. S. Park, J. Y. Lee, and J. H. Park, "Design and implementation of improved authentication system for Android smartphone users," in *Proceedings of the 26th IEEE International Conference on Advanced Information Networking and Applications Workshops (WAINA '12)*, pp. 704–707, March 2012.

[31] H.-W. Kim, J.-H. Kim, J. H. Park, and Y.-S. Jeong, "Time pattern locking scheme for secure multimedia contents in human-centric device," *The Scientific World Journal*, vol. 2014, Article ID 796515, 9 pages, 2014.

[32] J. D. Lee, Y.-S. Jeong, and J. H. Park, "A rhythm-based authentication scheme for smart media devices," *Scientific World Journal*, vol. 2014, Article ID 781014, 9 pages, 2014.

[33] P. Riedl, P. Koller, R. Mayrhofer, A. Möller, M. Koelle, and M. Kranz, "Visualizations and switching mechanisms for security zones," in *Proceedings of the ACM International Conference on Advances in Mobile Computing & Multimedia (MoMM '13)*, 278 pages, Vienna, Austria, December 2013.

[34] B. J. Casey, J. M. Logan, E. M. Cressall, and S. H. Cotterill, Gesture entry techniques, US Patent App. 13/651,100, April 2014, https://www.google.com/patents/US20140109018.

[35] B. J. Casey, J. M. Logan, E. M. Cressall, and S. H. Cotterill, Gesture entry techniques, US Patent App. 13/651,118, April 2014, https://www.google.com/patents/US20140109010.

[36] A. De Luca, A. Hang, F. Brudy, C. Lindner, and H. Hussmann, "Touch me once and i know it's you!: implicit authentication based on touch screen patterns," in *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*, pp. 987–996, New York, NY, USA, 2012.