An Implementation and Evaluation of Progressive Authentication Using Multiple Level Pattern Locks

William Aiken, Hyoungshick Kim
Department of Computer Science and Engineering
Sungkyunkwan University
Suwon, South Korea
billzo@skku.edu, hyoung@skku.edu

Jungwoo Ryoo, Mary Beth Rosson
College of Information Sciences and Technology
Pennsylvania State University
State College, Pennsylvania
jryoo@psu.edu, mrosson@ist.psu.edu

***Abstract*—This paper presents a possible implementation of progressive authentication using the Android pattern lock. Our key idea is to use one pattern for two access levels to the device; an abridged pattern is used to access generic applications and a second, extended and higher-complexity pattern is used less frequently to access more sensitive applications. We conducted a user study of 89 participants and a consecutive user survey on those participants to investigate the usability of such a pattern scheme. Data from our prototype showed that for unlocking low-security applications the median unlock times for users of the multiple pattern scheme and conventional pattern scheme were 2824 ms and 5589 ms respectively, and the distributions in the two groups differed significantly (Mann-Whitney U test, p-value less than 0.05, two-tailed). From our user survey, we did not find statistically significant differences between the two groups for their qualitative responses regarding usability and security (t-test, p-value greater than 0.05, two-tailed), but the groups did not differ by more than one satisfaction rating at 90% confidence.**

***Index Terms*—Usable security, user authentication, progressive authentication, pattern lock, graphical passwords**

I. Introduction

In recent years, the smartphone has come to store sensitive digital assets like credit cards, identity cards, vouchers, and mobile banking tokens. As a result, protecting sensitive data stored on smartphones via robust authentication methods is becoming increasingly important. Among many authentication mechanisms available, Personal Identification Numbers (PINs), graphical passwords, and biometric information are dominantly used. While other methods are being adopted, such as facial recognition and retina scanning, the robustness of their security in the long term continues to be questioned. This uncertainty may result in pattern locks and PINs remaining prominent and dependable authentication methods well into the future.

However, a user's security requirements vary depending on the type of applications installed. Online banking apps need more protection than a world clock app. Some high-sensitivity apps may require a password specific to only that application, but if users want to customize which apps require high protection, they are currently left with few options. Therefore, the all-or-nothing approach conventionally used in locking a mobile device deserves more scrutiny.

978-1-5386-7493-2/18/$31.00

In this paper, we propose a multiple pattern lock authentication scheme, which we refer to as the multi-level pattern lock. Via the multi-level pattern lock, a user is able to unlock low-security apps with an abridged version of the pattern while using the full, more complex pattern to unlock high sensitivity apps. Our research focuses solely on the derivation of the low level pattern as a shorter version of the full pattern to determine the feasibility of our approach in its minimalist form and the implications of its implementation on usability and security.

Following a user study aimed at measuring the quantitative aspects of our proposal, we conducted a survey on 75 out of the 89 participants in which we measured the participants' perceptions of the two methods. We also analyzed the entire theoretical password space for the extended pattern of this scheme and found that while still weak against long-term brute force attacks, our method offers some protection against powerful spying attacks like shoulder surfing and smudge reading because the users do not reveal their full patterns. The contributions of our paper are as follows:

1) We propose a novel authentication method that makes use of two unlock patterns of progressive length and complexity in order to access applications of different security levels.

2) We implemented a prototype of our proposed progressive authentication method and examined the usage data recorded when the participants were using it. This user study revealed statistically significant results when comparing an abridged pattern with a single level pattern as well as results that demonstrate that the multi-level pattern scheme does not significantly hinder usability.

3) From a user survey, we collected supplemental data about participants' feelings and experience with the prototype. The analysis of the data indicates they do not feel any more positive or negative about the security and usability of pattern locks when using the multi-level pattern lock scheme.

In the following section, we describe in more detail the related work that has already been done regarding pattern lock usability and concepts based on multiple security levels. In Section III, we explicitly describe which aspects of usability we evaluate, and we propose our novel multi-level pattern lock authentication scheme as well as our hypotheses regarding it. Section IV describes our user study in detail, and Section V discusses the results of that study as they apply to our hypotheses. We describe the results from our user survey in Section VI. In Section VII, we acknowledge the limitations of our work, and we conclude in Section VIII.

II. Related work

Regarding multi-level access controls in smartphones, Seifert et al. [11] developed "TreasurePhone" which allows the device owner to separate applications and data into "spheres" such as "family," "home", "friends," etc. Their study focused primarily on the usability of the level assignment processes like "Editing Access Rights" and "Navigating through Spheres." On the other hand, Riva et al. [10] devised a system that determines whether an app requires authentication based on a user's confidence level. This scheme is unique because it categorizes apps according to their sensitivity levels and allows access to the apps in a certain sensitivity category only when the user demonstrates an appropriate confidence level based on machine learning models.

Psychology and sociology also play an important role in authentication and security. For example, Muslukhov et al. [9] demonstrated that users often regard insiders (family, friends, etc.) as comparable threats to strangers. In other words, users reported that they implemented security features on their phones to protect private information not only from strangers but from their friends and family as well. Marques et al. [8] found that about 20% of U.S. adults had successfully committed snooping attacks (looking through someone's phone without permission). Despite this, Cherapau et al. [4] found that over 55% of their participants had shared their iPhone PINs with insiders for various reasons ranging from emergency use to just trusting the other person. Therefore, we can assume that insider attacks are not an imaginary security concern. We believe that multi-level authentication schemes could help alleviate this problem.

Other research has tried to definitively improve usability in Android authentication. The work done by Uellenbeck et al. [13] provides insight into some potential solutions to the security shortfalls of the default Android pattern scheme, but their conclusions regarding usability stated that their proposed arrangements were not superior to the standard Android pattern layout. Huh et al. [7] examined the role of "chunking" PINs (Personal Identification Numbers) into memorable segments. While the results did not prove a statistical difference in improving memorability for randomly-generated PINs, the numbers did imply a relationship that warranted further study. Additionally, focusing on pattern lock usability and security, Cho et al. [5] examined the impact of pattern policies that mandate one or two points be used in pattern creation. The work resulted in users creating passwords of equal memorability as regular Android patterns while being more robust against guessing attacks.

Other recent work has shown that pattern locks are unable to protect against shoulder surfing attacks, shown especially in Aviv et al.'s research [1] as well as against Cha et al.'s smudge attacks [3]. Additionally, algorithm-based shoulder surfing attacks have essentially rendered the pattern lock as providing no security at all in public settings. Ye et al. [15] demonstrated in their computer vision attack that Android pattern locks can be cracked in less than 5 attempts for over 95% of patterns and that more complex patterns were unable to provide better protection against the attack and that longer patterns were *easier* to crack. In many ways, this defeats previous attempts to encourage users to create more complex patterns.

III. Proposed authentication scheme

Depending on the input pattern, a user's privilege can be legitimately escalated to a specific level to give access to apps belonging to a particular security sensitivity level. At the same time, this allows users to access applications in public that do not require high security protections, which is true for nearly 75% of interactions with a smartphone [6], without revealing their extended pattern to any attackers that may be watching. However, we must emphasize that users of this scheme always want some level of required authorization. It is indeed possible to remove authentication entirely for the lowest sensitivity applications as some researchers and operating systems support. Not all users are in favor of this system for fear of theft, use of their device without permission, etc. Additionally, to decrease the burden on users, we investigate using only one pattern progressively rather than two separate patterns. In this section we will first describe our multiple level authentication system and then outline our hypotheses for it.

A. Multi-level access control using pattern locks

Unlike the conventional authentication systems where a single pattern is used for a single user, multiple patterns are required to provide a multi-level access control system. When setting up user patterns for multi-level access control, a user is required to have an unlock pattern consisting of at least k line segments (here, k is a system parameter which is used to guarantee a relatively high level of security). A line segment s is a part of a pattern that is bounded by two distinct end points.

Let $p = s_1 s_2 \quad s_k$ be the pattern that a user chose in the pattern generation step. We assume that given the security parameter k, users will choose a pattern consisting of k line segments to minimize their burden in memorizing the pattern. In the proposed system, any prefix of p is defined as a user pattern. A prefix $\hat{p}$ of a pattern p is any leading contiguous

segments of p and denoted by $\hat{p}$ p. When a prefix $\hat{p}$ of a pattern p is used as an input for the proposed authentication system, it has the security level $L(\hat{p}) = jxj$, where jxj is the number of segments in a pattern $\hat{p}$. During the authentication process, if the user inputs the pattern $\hat{p}$, the apps having the security level which are less than or equal to $L(\hat{p})$ will become accessible. For example, when a photo album app is security level 5, if users log in with a pattern of length 5, they can access the app, but the app cannot be accessed if they log in with a pattern of length 4. Note that all apps belonging to a lower security level are accessible as well.

Therefore in the minimalist case, the user establishes just one additional pattern consisting of a subset of the line segments of p (i.e., $\hat{p}$), allowing access to all apps that require privilege levels lower than $L(p)$. It is important to note that in our study a subset of pattern "12345678" could include patterns such as "1234" or "123456" but **not** "345678".

B. Research questions and hypotheses

Our use case design was motivated by one main exploratory research question: Does the two-level pattern lock system impose a significant usability burden on a user when compared with the conventional unlock pattern authentication system?

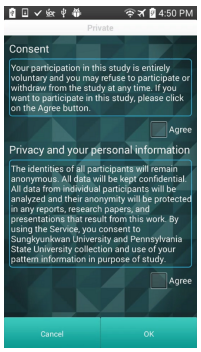
For our research purposes to answer this question, we measure and define usability by 4 quality components: **(1) Learnability**: time spent initially learning their custom generated pattern(s), **(2) Efficiency**: time required to reach a specific app when prompted with an initial log-in screen, **(3) Errors**: number of mistakes made when entering an unlock pattern, and **(4) Satisfaction**: user reports for the quality of multi-level pattern lock regarding usability and security.

Due to the time limitation of our study, we exclude a robust exploration of memorability as a factor for usability; users did not face the challenge of remembering their pattern(s) over a long period of time. Based on these research questions and our assumptions about the impact additional patterns would impose, we derived these three null hypotheses: **H1**: The time required to learn multiple levels of pattern locks is equal to the time required to learn one single level pattern lock. **H2**: The time required to input a choice of multiple levels of pattern locks is equal to the time required to input one single level pattern lock. **H3**: The number of attempts required to unlock a mobile device using a multiple level pattern lock is no different from using a single level pattern lock.

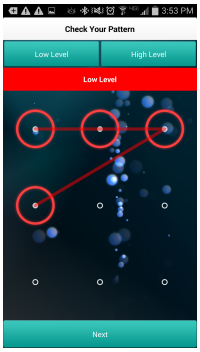
IV. User Study

A. Study design

To measure the usability of our design, we developed an Android application that emulated the process of inputting a pattern lock to open an application; however, our design also separated low-security applications from high-security applications. We explained to our participants what is an authentication scheme and how a pattern lock can be used as one. The participants were also given a cover story about how they would be using a pattern lock to unlock the phone in two different scenarios. We did not explain the goal of the user study to the participants or that we were recording their study time, input time, incorrect inputs, etc. However, we we did explain to the multi-level group that the abbreviated pattern was able to unlock low security applications but not high security ones. During this time, we explained to the participants that their participation was voluntary and that they had the right to terminate the study any time, which was reiterated in the consent form in the app as shown in Figure 1 (a). The ethical perspective of our research was validated through an institutional review board (IRB) at a university. The rest of this section will describe the steps and details of the user study and and data collection.



(a) Consent form



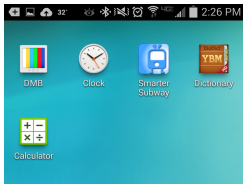
(b) Shorthand pattern

Fig. 1: Consent form and demonstration view from our multi-level pattern lock app.

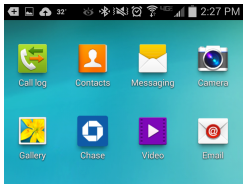
1) Group assignment: Each user was randomly assigned into one of the two groups: (1) Conventional unlock pattern-based authentication (single pattern, all-or-nothing approach), or (2) Two sensitivity levels, unlocked by the successful entry of correct unlock patterns (our proposed approach).

2) Pattern generation: After consenting to participate in our study, the participants were randomly assigned a full pattern lock, according to the procedures in Section IV-C.

3) Pattern learning and practice After the initialization process, the participants of the multi-level pattern lock scheme were shown a shorthand version of the full pattern lock as shown in Figure 1 (b) followed by their full pattern. Conventional pattern lock participants were not shown an abbreviated pattern. Starting from when the application began drawing the unlock pattern, a hidden timer began counting the time users spent studying their pattern. The participants then touched the 'Next' button and were prompted to draw the unlock pattern(s) themselves. The participants were prompted to enter their assigned pattern(s) twice to ensure they had learned it adequately. They were able to review their pattern by pressing back. After two successful inputs, the hidden timer for study times stopped counting.



(a) Sensitivity level 1 apps



(b) Sensitivity level 2 apps

Fig. 2: Examples of post-authentication screens that correspond to different levels of sensitivity. Sensitivity level 2 applications appear on their own screen, but the user can still swipe to access the sensitivity level 1 applications.

4) Use cases: Regardless of pattern type, every user performed two different use cases of their pattern(s) to expose the participants to the typical usage scenarios of our proposed approach. **Use case 1** represents the first variation in which multi-level participants used the shorthand version of the full pattern, which led the participants to a set of unprivileged or less sensitive apps (sensitivity level 1: Figure 2 (a)) but none of the high sensitivity apps. In our user study, we simply told our participants to open a calculator app, which is a sensitivity level 1 app. **Use case 2** represents the second variation in which the participants used the full pattern. Doing so gave the participant access to a set of designated high security or sensitive apps (sensitivity level 2: Figure 2 (b)). Note that in the single level pattern group, the participants would use the *same* pattern in both use cases. For use case 2, we told participants to open a banking app, which is a sensitivity level 2 app.

5) Survey questions: At the end of the tests, the participants were asked to take an optional 15 minute online survey which included a satisfaction questionnaire combined with a demographic questionnaire. We discuss the results of the survey in Section VI.

B. Participants

During the user study period, 89 people participated in the user study, and 75 filled out the questionnaire. The random split between the two groups resulted in 40 multi-pattern users and 49 single pattern users. Of our users, 40% reported never receiving any college education, and our mean age fell between 25 and 34 years old (with a median between 18 and 24). More detailed information about the demographics form our user study is shown in Appendix A.

We also measured the participants' experience with smartphones as well as their current unlock scheme usage. The vast majority of the users (75%) had been using a smartphone for over 4 years, but the remaining non-negligible 25% were relatively new users to smartphones. We found a surprisingly fairly even split between the participants who use fingerprints (25%) and those who use no scheme at all (24%). Fifteen of the participants reported using multiple authentication methods, either on the same device or across multiple devices, and the most noteworthy aspect is that while 10 users reported using a pattern lock on at least one device, only 2 users reported that they used *only* the pattern lock.

C. Pattern generation procedures

For two reasons, we decided to randomly generate patterns instead of allowing users to create their own. First, we wanted to remove any usability burden that may be introduced if the user has difficulty in thinking of or inputting a pattern of their own creation; our study was focusing on pattern input usability, not creation usability. Second, we wanted to remove the bias that users tend to have in pattern selection (i.e., starting from the top-left point or deliberately choosing very simple patterns) and ensure more consistency with regards to complexity. Based on their inherit security preferences, some users may design easy to input patterns, and others may create what they perceive as very difficult patterns. As discussed in Section II, these kinds of biases and behaviors have already been heavily studied, and we designed our study to focus solely on the differences between the single level and multi-level pattern lock performance.

The generated shorthand pattern was never less than 4 dots, but the long pattern (used as the only pattern in the single level cases) consisted of 6 or 7 dots. We decided our bounds based on the results in [13], in which participants revealed their real Android pattern locks, and they found the average pattern length was 5.63 with a standard deviation of 1.50. Accepting that users do occasionally create patterns of 4 dots, we chose this as the minimum possible length. The additional length was generated randomly. We acknowledge that only two or three additional dots may still leave the user vulnerable to brute-force long-term attacks as mentioned in Section VII.

V. User Study Results

This section reveals the results from our user study and discusses them as they relate to our hypotheses in Section III-B. The results of Ryan-Joiner tests for the recorded times required to learn and times required to input the patterns indicated our distributions were non-normal (all p-value values $< 0{:}01$). Therefore, we conducted Mann-Whitney U non-parametric tests to compare the multi-level pattern lock with the single level pattern lock to test those hypotheses. The number of multi-level pattern lock participants (n_1) was 40, and the number of single level pattern lock participants (n_2) was 49.

H1: Time required to learn: Our original hypothesis for this research was that "the time required to learn multiple levels of pattern locks is equal to the time required to learn one single level pattern lock." More specifically, we define "time required to learn" as the time spent on the screens where the pattern lock is drawn for the user added with the time spent on the screen where the user practices inputting the patterns for the first time (Step 3 of the user study).

By the Mann-Whitney U test, we discovered that the median time required to learn both patterns combined in the multiple level pattern lock scheme was longer than the median time required to learn the single level pattern with median values of 34252 ms and 21756 ms respectively, and the distributions were significantly different at a corrected p-value of 0.0312. Also by the Mann-Whitney U test, we discover statistically significant distributions of the time required to learn between the multi-level abridged pattern and the single level pattern; the medians are 14077 ms and 21756 ms respectively, and the corrected p-value is 0.0183. However, we do not find a statistically significant difference of the time required to learn between the multi-level full pattern and the single level pattern. Therefore, **we reject the null H1 hypothesis** and conclude that there is a significant difference in the time required to learn multiple levels of pattern locks versus one single level pattern lock. Users of the multi-level pattern lock scheme required significantly less time to learn the abridged pattern but required more time overall to learn both patterns. As a result, the implications on the learnability aspect of usability are unclear.

H2: Input time: Our second hypothesis supposed that "the time required to input a choice of multiple levels of pattern locks is equal to the time required to input one single level pattern lock." We conducted a Mann-Whitney U test for the time required to open a low security application comparing the multi-level abridged pattern and the single level pattern. The abridged pattern's median was 2824 ms, and the single level pattern's median was 5589 ms, with statistically significant distributions at a corrected p-value of 0.0034. Of course, the abridged pattern was either 2 or 3 segments shorter than the single level pattern; however, this time includes the time required for the user to decide which pattern to use.

Therefore, in the case of the abridged pattern regarding the efficiency aspect of usability, **we reject the H2 null hypothesis** and conclude there is a significant difference in time required to input a choice of multiple levels of pattern locks, where users perform more quickly with the choice of an abridged pattern than with a single level pattern. We also compared the performance of the full multi-level pattern and the single level pattern and as expected found no significant difference of medians 4845 ms and 5602 ms respectively at a corrected p-value of > 0.9995.

H3: Attempts required: We also supposed that the number of attempts required to unlock a mobile device using a multiple level pattern lock is equal to using a single level pattern lock. Note that entering a low level pattern lock for a high sensitivity app was designated as an incorrect entry, and the user was notified of this mistake. Our findings indicate that there was no significant difference in the number of attempts to unlock at a p-value of 0.711 for a two-tailed t-test of difference between the two groups (with means of 1.46 and 1.52, respectively) attempts. Using our smaller sample size of 40 for the multi-level pattern lock group, we tested at an 85% confidence rate that there was no more than a number of 0.5 attempts difference between the two groups. Therefore, **we do not reject the H3 null hypothesis** and conclude the difference regarding the error aspect of usability between the two groups is no more than 0.5 attempts.

The impact of complexity on performance: It is possible that some of the participants received easier patterns than others and that this introduced bias into our measurements. To combat this, we took a measure of the complexity of the pattern using the algorithm developed in [12] during the study. Due to the non-normal distribution of our data, we performed Spearman's correlation tests to discover what role the complexity of the pattern had on user performance. We found statistically that across all case study groups: (1) complexity and time required to learn had weakly correlated with Spearman Rank-Order Correlation (r_s) of 0.161 at a p-value of 0.032, and (2) complexity and time required to input had weakly correlated with $r_s = 0.225$ at a p-value of 0.002, but we did not find any statistically significant correlation between complexity and number of attempts ($r_s = 0.019$ at p-value of 0.802). In our study, we were unable to remove the influence of the complexity of the pattern on performance entirely, but during our user study we did ensure that neither group received full patterns that were statistically significantly more complex than the other on average.

VI. User survey and results

Our survey aimed to evaluate the perceived usability (i.e., satisfaction) of the system by the users of the user study; if users believed the multi-pattern was a burden, they may be inclined to disable the multi-level pattern scheme regardless of its actual usability or security benefits. The participants' subjective satisfaction was evaluated using an ordinal rating scale in order to measure the overall perception of and ability to measure security and in order to compare any differences between the two groups regarding the security and usability of our method. We chose not to implement a more sophisticated survey framework like System Usability Scale (SUS) because we need not compare the multi-level pattern lock method to other authentication methods in general; other studies have done this for the pattern lock as discussed previously. We need only compare our two study groups. Ordinal and consistent survey questions provide enough information for our baseline comparison.

One of the key benefits of using a multi-level pattern lock system is the ability to separate apps into two distinct categories based on their security level. However, if users are unable to make such a distinction correctly, the secondary pattern provides no practical security at all. As shown in Figure 3, survey results indicate that while over 70% of all participants place at least some value on information security, half of participants claim that they would be neutral or worse at deciding an apps' security level. This strongly indicates that users may need assistance in assigning an app's security level.

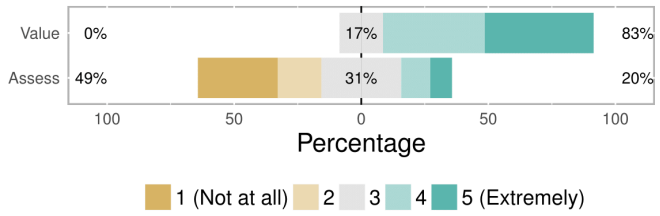


Fig. 3: Comparison of how users reported how much they value information security vs how well they can assess it.

As for the subjective feelings about the multi-level pattern lock scheme, results were mostly positive. We asked the 75 participants who took the survey to rank on a scale from *Not at all (1)* to *Extremely (5)* how well they believe our proposed scheme provides security in one question and how well they believe our scheme provides usability in the following question. The majority of the participants responded with better than neutral responses for both security and usability. The mean for multi-level users responded that the application provided a 4.2 out of 5 for security and a 3.9 for usability; the mean for single-level users responded that the application provided a 4.0 for security and a 3.8 for usability. Statistically, there was no significant difference between the two groups' responses for either security or usability via two-tailed t-tests (at p-values of 0.272 and 0.548, respectively). The user study confidence in this regard is 90% that the groups did not differ significantly by more than 0.75 points by the ranking scale. We can take this as a slight indication that participants' satisfaction aspect of usability of pattern lock usability may not be greatly impacted when using a multi-level pattern lock scheme. In other words, after learning two patterns, users still reported that they felt pattern locks provided usability quite well.

Moreover, we mitigate the limitations imposed by uncontrolled complexity on our survey results with an additional analysis. Based on a Spearman Rho correlation analysis, we found the correlations between the complexity of the patterns and the qualitative feedback on both usability and security are neither large nor statistically significant (r_s = 0:036 at p-value of 0.762; r_s = 0:015 at p-value of 0.900, respectively).

VII. Limitations and Future Work

One main limitation in our study may be in the representativeness of pattern lock users. Many participants were students, but because of the number of faculty members who also participated, our average age group (25-34) is still comparable to similar research conducted in this field like in Zezschwitz et al.'s work [14] where the average pattern lock user age was 26. However, as discussed in Section IV, we received a very unusual ratio of users who reported using a password for their device to users who reported using a pattern lock for their device. We believe that respondents may be responding in this way because they either use a password on a tablet device that is running a desktop operating system, or because they are applying this question to the application level. Despite this discrepancy, when also totaling the percentage of users that reported using the pattern lock along with some other scheme, we find that 16% of our respondents were pattern locks users in some way, which is much closer to the percentage of approximately 25% found in similar studies [2].

Because of Android's pattern lock rules, two or three additional dots only provide a very limited number of additional patterns in the minimalist implementation, and the number of combinations depends on the structure of the abridged pattern. For example, when analyzing pattern 1243, 1 more dot has 3 possible legal combinations, 2 more dots have 15 combinations, and 3 more dots have 47 combinations. If the attacker does not know the length of the full pattern (be it 5, 6, or 7 dots), there is a total of 47 combinations that the user could implement, assuming the user does not use an 8 or 9 dot pattern, as tested in this research. If the user has the ability to implement a full pattern of 8 or 9 dots in length, the total combination count reaches 107 and 167 respectively for pattern 1243. Visual examples of the available extended pattern space is included in Figure 4 in Appendix B. For a comparison of the available password space, the maximum number of combinations for any extended pattern is 325 while a regular 4-point Android pattern lock has 1,624 combinations.

Finally, the multi-level pattern lock system could easily consist of two separate patterns, but we limited our research to two progressive patterns. This research aimed at understanding a minimalist implementation of progressive authentication of the pattern lock, and unless explicitly prohibited, some users may implement such a method for their own patterns. Future work will build upon our findings to evaluate how this scheme performs with two separate patterns. Moreover, progressive authentication could also be applied to PINs or passwords in a similar fashion. For example, a 4-digit abridged PIN could unlock low security apps, and a 6-digit PIN could unlock high security apps. Implementation in other authentication schemes is beyond the scope of this research, but our minimalist implementation in the pattern lock serves as a motivation to consider their feasibility and as a baseline for future comparisons.

VIII. Conclusion

In our research, we augmented the conventional pattern-based authentication method to provide progressive authentication where the abridged pattern was derived from the high security pattern. We introduced a minimalist implementation of such a system using pattern locks similar to those deployed in popular mobile environments. Results from the user study showed users were able to unlock their device for low security applications more quickly in the multi-level scheme, but they also revealed that users required more time to learn both patterns. Performance regarding the number of attempts does not seem to be significantly affected by either method. Additionally, survey responses indicate that users may not feel any more positive or negative about the security and usability of pattern

locks when using our proposed method, which suggests its implementation would not discourage users from adopting our method. Our contributions aimed at bridging the gap between security and usability of multi-level authentication for the pattern lock scheme.

REFERENCES

[1] A. J. Aviv, J. T. Davin, F. Wolf, and R. Kuber, "Towards baselines for shoulder surfing on mobile authentication," *CoRR*, vol. abs/1709.04959, 2017. [Online]. Available: http://arxiv.org/abs/1709.04959

[2] E. Bursztein, "Survey: most people don't lock their android phones - but should," https://elie.net/blog/survey-most-people-dont-lock-their-android-phones-but-should, [Online; accessed 03-May-2018].

[3] S. Cha, S. Kwag, H. Kim, and J. H. Huh, "Boosting the guessing attack performance on Android lock patterns with smudge attacks," in *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security*, ser. ASIACCS '17. New York, NY, USA: ACM, 2017, pp. 313–326. [Online]. Available: http://doi.acm.org/10.1145/3052973.3052989

[4] I. Cherapau, I. Muslukhov, N. Asanka, and K. Beznosov, "On the impact of Touch ID on iPhone passcodes," in *Proceedings of Symposium On Usable Privacy and Security (SOUPS)*, Jun 2015.

[5] G. Cho, J. H. Huh, J. Cho, S. Oh, Y. Song, and H. Kim, "SysPal: System-guided pattern locks for Android," in *2017 IEEE Symposium on Security and Privacy (SP)*, May 2017, pp. 338–356.

[6] M. Harbach, E. 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," *Proc. SOUPS '14*, pp. 213–230, 01 2014.

[7] J. H. Huh, H. Kim, R. B. Bobba, M. N. Bashir, and K. Beznosov, "On the memorability of system-generated pins: Can chunking help?" in *Eleventh Symposium On Usable Privacy and Security (SOUPS 2015)*. Ottawa: USENIX Association, 2015, pp. 197–209. [Online]. Available: https://www.usenix.org/conference/soups2015/proceedings/presentation/huh

[8] D. Marques, I. Muslukhov, T. Guerreiro, K. Beznosov, and L. Carrico, "Snooping on mobile phones: Prevalence and trends," in *Proceedings of Symposium On Usable Privacy and Security (SOUPS)*, Jun 2016.

[9] I. Muslukhov, Y. Boshmaf, C. Kuo, J. Lester, and K. Beznosov, "Know your enemy: The risk of unauthorized access in smartphones by insiders," in *Proceedings of the 15th International Conference on Human-computer Interaction with Mobile Devices and Services (Mobile HCI)*, Jun 2013.

[10] O. Riva, C. Qin, K. Strauss, and D. Lymberopoulos, "Progressive authentication: Deciding when to authenticate on mobile phones," in *Proceedings of the 21st USENIX Conference on Security Symposium*, ser. Security'12, 2012.

[11] J. Seifert, A. De Luca, B. Conradi, and H. Hussmann, "TreasurePhone: Context-sensitive user data protection on mobile phones," in *Proceedings of the 8th International Conference on Pervasive Computing*, ser. Pervasive'10. Berlin, Heidelberg: Springer-Verlag, 2010, pp. 130–137. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-12654-3_8

[12] Y. Song, G. Cho, S. Oh, H. Kim, and J. H. Huh, "On the effectiveness of pattern lock strength meters: Measuring the strength of real world pattern locks," in *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, ser. CHI '15. New York, NY, USA: ACM, 2015, pp. 2343–2352. [Online]. Available: http://doi.acm.org/10.1145/2702123.2702365

[13] 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 2013 ACM SIGSAC Conference on Computer & Communications Security*, ser. CCS '13. New York, NY, USA: ACM, 2013, pp. 161–172. [Online]. Available: http://doi.acm.org/10.1145/2508859.2516700

[14] 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*, ser. MobileHCI '13. New York, NY, USA: ACM, 2013, pp. 261–270. [Online]. Available: http://doi.acm.org/10.1145/2493190.2493231

[15] G. Ye, Z. Tang, D. Fang, X. Chen, K. Kim, B. Taylor, and Z. Wang, "Cracking Android pattern lock in five attempts," in *Proceedings 2017 Network and Distributed System Security Symposium 2017 (NDSS'17)*. Internet Society, 2017.

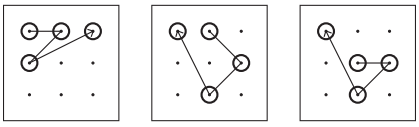
APPENDIX A
DEMOGRAPHICS

TABLE I: The demographics of 75 of the 89 participants in the user study*

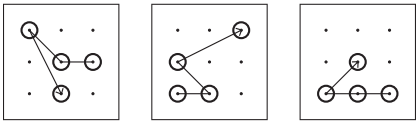
Age group	
18–24	64% (48)
25–34	8% (6)
35–44	9% (7)
45–54	12% (9)
55–64	5% (4)
65+	1% (1)
Gender	
Male	59% (44)
Female	41% (31)
Highest level of education completed	
High school	40% (30)
Some college	9% (7)
Associate degree	5% (4)
Bachelors degree	36% (27)
Masters degree	6% (5)
Doctorate degree	4% (3)

**The remaining 14 participants declined to take part in the survey*

APPENDIX B
PATTERN LOCK COMBINATION ANALYSIS



up to additional dot:	1243	2681	5681
1-dot combinations:	3	3	3
2-dot combinations:	15	14	15
3-dot combinations:	47	45	51
4-dot combinations:	107	105	123
5-dot combinations:	167	165	195



	6518	7843	9875
1-dot combinations:	5	3	5
2-dot combinations:	25	15	23
3-dot combinations:	85	47	73
4-dot combinations:	205	107	169
5-dot combinations:	325	167	265

Fig. 4: Examples of low-level patterns that have different numbers of high-level combinations; count is the number of possible combinations above the abridged 4-dot pattern