Evaluating Re-authentication Strategies for Smartphones

by

Lalit Agarwal

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Mathematics
in
Computer Science

Waterloo, Ontario, Canada, 2016

This thesis consists of material all of which I authored or co-authored: see Statement of Contributions included in the thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Statement of Contributions

The content in Chapter 1 and Chapter 5 was co-authored with Hassan Khan. The statistical results provided in Chapter 5 were produced by Hassan Khan. All the other chapters in this thesis contain original work authored under the supervision of Urs Hengartner.

Abstract

Re-authenticating users may be necessary for smartphone authentication schemes that leverage user behavior, device context, or task sensitivity. However, due to the unpredictable nature of re-authentication, users may get annoyed when they have to use the default, non-transparent authentication prompt for re-authentication. We address this concern by proposing a few configurations with varying levels of screen transparency and time delays when displaying the authentication prompt. We conduct user studies with 30 participants to evaluate the usability and security of these configurations. We also study whether the user preferences of the configurations vary depending on the application the participants are using on their device or their surrounding environment. We find that the participants generally prefer the authentication configuration with a non-transparent background for sensitive applications, such as banking and photo apps. Our findings also indicate that the user preferences are inclined towards convenient, usable configurations while participants are using their devices at home. Though we did not observe any significant differences in the task completion overhead and context switch overhead among our proposed configurations, we find that participants utilize the time delay just before the authentication prompt is going to appear to complete their current task. We also provide implementation details of our Android lock library, FireLock, which developers can use to re-authenticate users while they are using their app. We conclude with suggestions to improve the design of the proposed configurations as well as a discussion of other mechanisms to notify the users in case of re-authentication.

Acknowledgements

I would like to thank my supervisor, Prof. Urs Hengartner, for his guidance and motivation throughout my Masters program. I want to thank Hassan Khan as well for his continuous guidance and support. I would also like to thank my readers Prof. Ian Goldberg and Prof. Daniel Vogel for their invaluable insights and comments.

Finally, I would like to thank my parents and brother for their love and support throughout my academic career. Also this thesis would not have been possible without the constant support of my friends in Waterloo.

Dedication

This is dedicated to my parents.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Overview

The growing usage of smartphones for both personal and corporate data has increased the need to authenticate users at multiple levels. Explicit Authentication (EA) mechanisms such as a PIN/pattern lock, text-based passwords or fingerprint recognition require the users to explicitly unlock their device by entering a pass-code in order to access the apps and other content stored on their devices. While a device-level authentication scheme is required to protect access to the device, an app-level authentication may provide further protection from accessing social networking, banking or enterprise apps. However, existing studies have shown that the short and frequent nature of smartphone sessions creates usability issues for device-level authentication schemes [HDA^{+}13] whereas constrained keyboards on smartphones are a bottleneck when users are authenticating using text-based passwords [SDW12]. Also recent studies [EJP^{+}14, HVZF^{+}14] have found that more than 40% of the participants do not use any authentication mechanism on their devices because they find it inconvenient to unlock their device every time they want to access the content on their device.

To mitigate these usability issues, researchers have proposed several techniques that reduce the authentication burden by leveraging user behavior [LZX13, SNJC10, XZL14], device context [HDA^{+}13, MJB^{+}15, MHK^{+}14] or the sensitivity of launched apps [HRS^{+}12]. While these schemes reduce the authentication burden on the user, they may require mid-task re-authentication. Schemes that leverage user behavior need re-authentication in case of a behavior mismatch against the current phone user. Similarly, device context-based schemes may need to establish a user's identity in case a contextual source (e.g., ambient noise) changes. Taking the sensitivity of launched apps into account for authentication may also require mid-task re-authentication. For instance, some users have indicated that for a messenger app, only opening old messages should trigger re-authentication [HRS$^+$12].

1.2 Re-authentication Scenarios

In this section, we describe some of the authentication techniques discussed above and their potential need to re-authenticate users.

- **Implicit authentication (IA):** Mobile devices typically have many built-in sensors which are capable of collecting rich sets of information such as the current location of the user, their touch input data or their gait pattern. IA uses the information collected by these sensors to conveniently authenticate users without requiring their explicit input. Various IA schemes have been proposed that authenticate users through their touch input behavior [FBM$^+$13, LZX13, XZL14], keystroke behavior [DZZ13, FZCS13, GMCB14], gait behavior [FMP10, MM13] or device usage behavior [SNJC10, SYJ$^+$11]. Several IA proposals have been shown to provide over 95% accuracy [FBM$^+$13, LZX13, XZL14] and researchers have proposed to use them as a primary authentication mechanism for users who do not lock their device or as a secondary authentication mechanism to complement the existing primary authentication schemes.

  There are scenarios when an IA scheme is unsure about the identity of the user. This uncertainty may be caused by an adversary using the device or it could be the result of a false reject. False rejects occur when legitimate users are misclassified as adversaries. When an IA scheme is unsure about the identity of the user, it uses an explicit authentication mechanism to re-authenticate the user. Furthermore, if an IA scheme relies on the input behavior of the user, the false rejects can occur mid-task and re-authentication requires interrupting the current task of the user [KHV15].

- **Context-aware authentication:** Several schemes have been proposed that leverage device context to reduce authentication overhead [HDA$^+$13, MJB$^+$15, MHK$^+$14, RQSL12]. These schemes rely on a variety of contextual sources, including location, proximity to WiFi and Bluetooth devices, and ambient light and noise. An evaluation of CASA [HDA$^+$13] shows that it can reduce explicit authentications by 68% and a lab study of the scheme proposed by Riva et al. [RQSL12] indicates that it can reduce the number of explicit authentications by 42%.

Context-aware schemes can be deployed to sense and assist in authentication only when users begin their interaction with the device. However, to preclude attacks from informed attackers (such as friends and coworkers), a continuous authentication scenario is more suitable. For instance, a continuous proximity sensing scheme will not allow an informed malicious coworker to unlock the device at the workplace and then move to a secluded place to access personal data on the device. Since such scenarios may arise with the legitimate user of the device (e.g., the device owner moves out of the proximity range while using the device, or an ambient noise sensor may switch off), the device owner may be subjected to mid-task re-authentication.

- **App-specific authentication:** Hayashi et al. [HRS+12] show that all-or-nothing access to smartphones does not align with user preferences. They find that while the majority of the users prefer to be authenticated for selected apps only, for a subset of apps the users want some functionality to be available always and some functionality to be available after authentication. For instance, browsing existing entries (such as contacts) in an app should always be available while modifying or deleting entries should require authentication. Similarly, looking at recent messages should not require authentication while browsing old messages should require user authentication. These scenarios require mid-task re-authentication of the user.

1.3 Need for Better Re-authentication Schemes

Preliminary evaluations show that users like the convenience offered by the schemes discussed in the previous section [CR14, HDA+13, HRS+12, KHV15, MJB+15]; however, a field study of behavior-based authentication shows that re-authentications are a potential issue [KHV15]. More specifically, the evaluated scheme used a (simulated) behavior-based authentication scheme that focused on the user's touch input behavior. Whenever re-authentication was required, the user's current task was interrupted and a re-authentication prompt with dark background, similar to the standard Android authentication prompt, appeared immediately. Non-surprisingly the unpredictability of a re-authentication and the context switch due to the task interruption were annoying to some users.

While re-authentication is unavoidable to preclude misuse of a device or an app, the unpredictability of re-authentication can be reduced by delaying the transition between the current task and the re-authentication prompt through a fade-in effect. During the fade-in, the user is allowed to continue interacting with their current task on the device. In addition to the fade-in effect, the re-authentication prompt can be configured to have varying levels of screen transparency to provide a visual of the user's current task in the background. The fade-in effect should reduce the unpredictability of the re-authentication and a visual of the current task of the user should reduce the context switch overhead due to re-authentications. Together these controls have the potential to provide increased usability at the cost of reduced security.

Thesis Statement: Re-authentication may be necessary for certain scenarios and the widely deployed authentication mechanisms can be modified to make them more usable for re-authentication without significantly comprising on security.

1.4 Contributions

In the thesis, we evaluate different configurations of explicit authentication schemes (such as PINs or pattern locks) when used for re-authentication. Our focus is on the time delay and the transparency of the re-authentication prompt. We choose behavior-based authentication as a target use case to evaluate the different configurations; however, our findings can be generalized to other authentication proposals that result in re-authentications. In addition to the re-authentication configuration employed in the previous work [KHV15], which has a non-transparent background, we select three configurations of explicit authentication schemes for re-authentication:

i) The authentication prompt appears immediately (no time delay) and the background of the authentication prompt is transparent to provide a visual of the user's current task in the background;

ii) The authentication prompt appears immediately and the background of the authentication prompt gradually transitions from transparent to opaque for improved security; and

iii) The authentication prompt appears after a four-second time delay and the background of the authentication prompt gradually transitions from transparent to opaque.

We perform lab experiments using synthetic tasks to evaluate the security perception, ease of use, obstructiveness and annoyance of PIN and pattern-lock-based re-authentication based on the default configuration from the earlier study [KHV15] (as a baseline) and the modified configurations. In addition to these qualitative usability metrics, we collect quantitative data on the task efficiency and the task error rate for a multifaceted evaluation of these configurations. Finally, we conduct interviews to gather participants' perceptions on the sensitivity of different kinds of apps and of participants' preferred configuration of the re-authentication prompt for different apps and different environments.

Our study was completed by 30 participants. Though our findings indicate no differences for the user performance (in terms of task efficiency, task error rate, and context switch overhead) against these configurations, participants found all three modified configurations to be less annoying and less obstructive as compared to the default configuration. The modified configurations were also at least as easy to use as the default configuration. As expected, the perceived security level of the modified configurations was quite low when compared to the default configuration. While the low perceived level of protection was a bottleneck in the adoption of the modified configurations in high-risk environments and for sensitive content, a significant number of participants preferred the proposed configurations over the default configuration for less sensitive content and for low-risk environments.

Based on the suggestions given by the participants, we develop FireLock, an open-source Android lock library allowing the app developers to re-authenticate the users while they are using their app. The lock library is compatible with most of the recent Android smartphones and can be easily used by the developers by adding only three lines in their code. We also provide a demo of our lock library for a behavior-based authentication scheme and integrate it with an open-source messaging app.

In the next chapter, we present some of the related work in this area. In Chapter 3, we describe in detail the proposed lock configurations. In Chapter 4, we describe our user study setup and the participant sample. We report the findings of the study in Chapter 5 followed by the details of the implementation of our lock library in Chapter 6. We also communicate the suggestions by the participants on how to improve the design of our proposed configurations and discuss guidelines for future implementations of re-authentication schemes in Chapter 7.

Chapter 2

Related Work

In this chapter, we present the related work in this area. In Section 2.1, we discuss the existing work on understanding usability issues with existing authentication schemes and present some of the alternate schemes that have been proposed to conveniently authenticate users. Section 2.2 focuses on the existing work in the areas of authentication scenarios that require re-authentication.

2.1 Authentication Schemes

Several authentication schemes have been proposed in the past ranging from alphanumeric password schemes to graphical and gesture-based password schemes. Most of these authentication schemes are susceptible to memorability issues, shoulder-surfing attacks and smudge attacks. However, as we discuss later in this section, apart from these security issues, usability issues with these schemes, such as ease of unlocking, the time taken to unlock, influence the users' preferences of these schemes. More recently biometric-based schemes, which include fingerprint recognition, face recognition and behavior-based authentication, have been introduced. Though biometric approaches provide advantages over the traditional authentication methods, these authentication techniques have still not become very popular due to a fear among the users of a potential risk to their biometric data and high authentication times compared to other schemes [ZKM15].

2.1.1 Usability Issues with Existing Authentication Schemes

Researchers have extensively investigated the usability issues with primary authentication schemes [DLHVZH15, HVZF$^+$14, TSK$^+$12, VZDDL13, ZKM15] and have shown that these issues prevent users from using these schemes [EJP$^+$14, HVZF$^+$14].

PIN/Pattern Lock

Harbach et al. [HVZF$^+$14] recently performed a real world study on smartphone unlocking and found that users spend a significant amount of the overall device usage time (2.9% on average) on entering their PINs or unlock patterns. They also found that locking the device is not necessary for all the apps installed on the device as sensitive data is rarely accessed. They observed that the participants who did not lock their phone were satisfied with their choice and indicated very few situations where they would rather have locked their screens. They concluded that higher authentication frequency reduces the usability of the authentication scheme and recommended that researchers should focus on decreasing the number of unlocks by deploying context-dependent locking mechanisms. We take this recommendation into consideration for our study.

A research study was conducted by Von Zezschwitz et al. [VZDDL13] which aimed at understanding the performance of a PIN and a pattern lock on mobile devices. Though quantitative results of the study indicated that the pattern lock is more prone to errors, it was rated favorably in terms of ease-of-use and efficiency. Therefore, while designing the lock configurations, we give a lot of importance to enhance the usability experience of the users during re-authentication.

Biometric Authentication

De Luca et al. [DLHVZH15] conducted an on-line study using MTurk to understand the reasons behind using and not using biometric authentication on smartphones. They focused their study on the two most prevalent biometrics systems, Apple's Touch ID and Android's Face Unlock, and found that the users preferred using Touch ID as they found it to be faster and more convenient compared to Face Unlock. Many participants reported that they stopped using Face Unlock because they found it to be slow in cases when they needed quick access to their device. Touch ID on the other hand did not impose much overhead and was therefore preferred by the users.

While usability is a major concern, users are worried about the security and privacy of their biometric data as well. Zirjawi et al. [ZKM15] conducted an on-line survey with 139 participants to understand the user requirements and preferences for biometric authentication. They reported in their study that users were concerned about how their biometric data was used and where it was stored. Among the biometric data, fingerprint information was the most critical information for most of the participants. The authors gave design recommendations and suggested that building secure authentication could help in building trust among the users towards the system.

2.1.2 Alternate Authentication Schemes

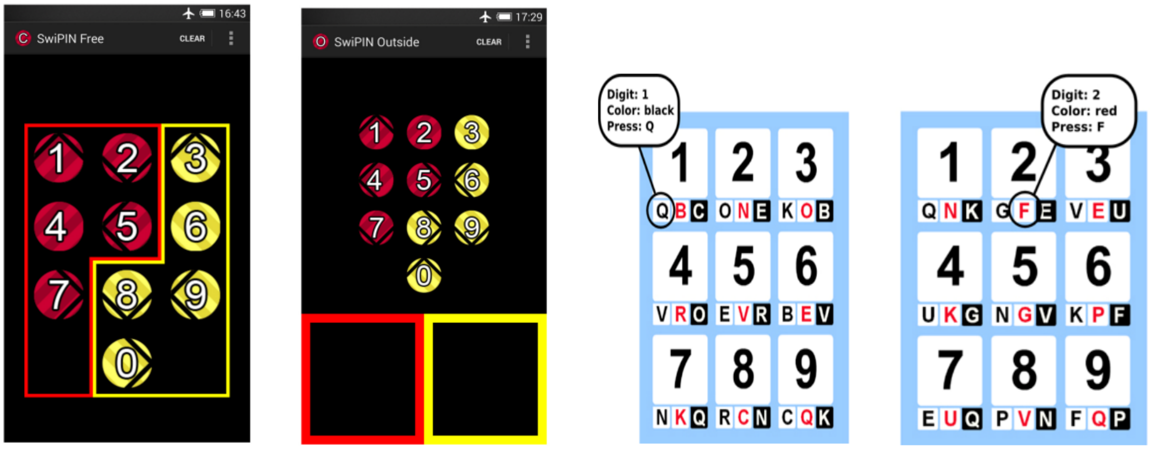
Another line of research has focused on addressing the usability issues with existing primary authentication schemes by proposing alternate mechanisms, including new PIN-based [DLHH10, VZDLBH15] schemes, graphical passwords [KN14, SWKW13, WWB+05] or gesture-based authentication [AK14, SLS13] schemes.

PIN-Based Password Schemes

New PIN based authentication schemes have been proposed to counter the shoulder surfing attacks and improve the usability issues as mentioned in the previous section. In 2015, Von Zezschwitz et al. [VZDLBH15] proposed SwiPIN, which uses touch gestures in combination with PIN to protect against shoulder surfing attacks. This system is based on a random assignment of simple touch gestures to specific digits. In order to enter a digit, users simply perform the gesture associated with that digit at a location different than the respective button (Figure 2.1a). They concluded that SwiPIN is easy to use, resistant against shoulder surfing attacks and is very useful in sensitive environments such as in an office or around strangers.

De Luca et al. [DLHH10] proposed ColorPIN by representing each digit with a color. The keypad displays all the ten digits along with three colors (black, red and white) at the bottom of each digit. Each color displays a random alphabet and thus to enter a four-digit PIN, the user is required to enter a four-letter word from the keyboard corresponding to the colors of each digit of the PIN (Figure 2.1b). The authors reported that even though it was more secure and the error rate of ColorPIN was low compared to the standard PIN-based scheme, it was almost ten times slower.

The proposed PIN-based systems discussed above are able to successfully address the existing security issues related to PIN-based schemes. However, on the downside, almost all of these authentication techniques still follow an all-or-nothing approach: users have to authenticate or they cannot access anything. Also the improved security is provided at the cost of incurring an increase in the authentication times thereby negatively affecting the usability of these schemes.



(a) SwiPIN [VZDLBH15]

(b) ColorPIN [DLHH10]

Figure 2.1: Two recently proposed PIN-based authentication schemes.

Graphical Password Schemes

Recently, Locimetric systems have been proposed where the users have to identify specific regions in an image to authenticate themselves. A popular example of such a system is Passpoints by Wiedenbeck et al. [WWB+05] where users can select an arbitrary image and then define a password through click points within the image. An important feature of the Passpoints system is that the underlying images for a password can be selected by the users and can even include complex real-world images. Locimetric authentication systems were recently adopted by Microsoft in their Windows 8 operating system where the user is required to draw a combinations of shapes on a picture selected by them. The usability evaluation conducted by the authors and the results of the evaluation are promising with respect to memorability of the graphical passwords. However, as reported by Thorpe et al. [TvO07], these password schemes are not entirely attack-resistant and popular points or "hot-spots" exist for many images, which can be exploited by an attacker. They found that their attack strategy was quite successful and they were able to guess 36% of user passwords within 2^{31} guesses in one instance, and 20% of passwords within 2^{33} guesses in another.

Kwon and Na proposed TinyLock [KN14], which utilizes simple gestures to provide protection against shoulder surfing and smudge attacks. They addressed the problem by using a small 3x3 grid thereby confining the smudge spots to a smaller region (Figure 2.2a). Since smudges are confined to a small spot and are masked during this process, it became difficult for an adversary to observe the passcode.

Users have reported positive experiences during preliminary evaluations of these schemes [KN14, WWB+05]. However, as reported in these studies, since these systems are relatively new, it took some time for the first time users in understanding and using these password systems effectively. Another drawback of these systems is that the authentication times for these schemes are relatively high compared to the authentication times of existing schemes.

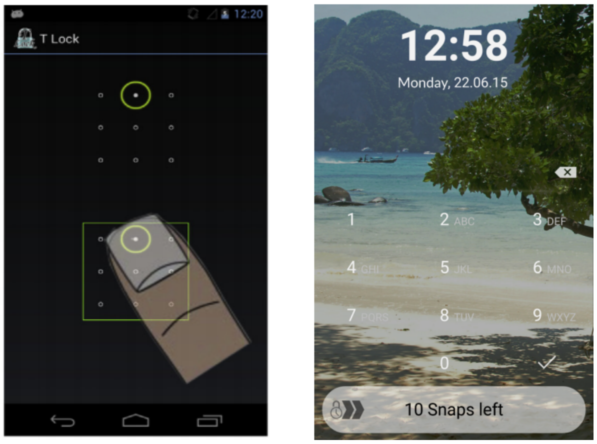
Gesture-Based Password Schemes

AirAuth presented by Aumi et al. [AK14] is a biometric authentication technique that uses in-air gesture input to authenticate users. They tracked the hand gesture input by the user using a short-range depth camera and used the features collected to decode the user's authentication secret. They were able to achieve an average accuracy of 99.6% using a predefined gesture set in an equal-error rate (EER) based study. Similar to our study, they analyzed the usability of their proposed system in three different scenarios — private (e.g., home), office, and public (e.g., around strangers) environments. Their proposed system received a higher usage acceptability rating in all the three scenarios compared to the pattern lock authentication technique found on Android devices.

Another related work is SnapApp [BHVZ+16], which is a primary authentication mechanism that provides a trade-off between security and usability. It presents a user with two unlock methods on the device screen — a PIN for secure access to the entire device and a simple slide gesture for fast yet temporary access (30 seconds or less) to the device (Figure 2.2b). Furthermore, a blacklist can be configured by the user to exclude apps that can be launched during the fast access to the device. Similar to our work, SnapApp favors usability at the cost of security; however, we feel that evaluating this scheme for re-authentication may introduce confounding factors as it has not been adopted widely. To the best of our knowledge, we perform the first ever evaluation of modified primary authentication schemes for re-authentication scenarios.

2.2 Re-Authentication Scenarios

To mitigate the usability issues with the traditional authentication schemes, several research proposals have been put forth that reduce the authentication overhead of the users



(a) TinyLock [KN14] (b) SnapApp [BHVZ+16]

Figure 2.2: Two recently proposed gesture-based authentication schemes.

by leveraging user behavior [LZX13, SNJC10, XZL14], device context [HDA+13, MJB+15, MHK+14] or the sensitivity of launched apps [HRS+12]. We provided a brief overview of these schemes in Section 1.2. We describe some of the related work in these areas below.

2.2.1 Implicit Authentication

Implicit Authentication (IA) makes use of the information collected by numerous sensors present on smartphones to continuously authenticate the device user in the background. IA creates a behavior-based profile of users and then monitors real-time device usage to detect any anomalous user behavior. Various IA schemes have been proposed that authenticate users through their touch input behavior [FBM+13, LZX13, XZL14], keystroke behavior [DZZ13, FZCS13, GMCB14], gait behavior [FMP10, MM13] and device usage behavior [SNJC10, SYJ+11].

Li et al. [LZX13] proposed a continuous and unobservable biometric re-authentication scheme using touch input of the user. They used the sliding right, left, up, down and tap gesture to classify the behavior of the user as they found these to be the most common gestures (88%) of all the gestures. They conducted their study on a rooted device and the SVM-based authentication used by them was able to attain a classification accuracy of 95%. Similar to Li et al., Touchalytics proposed by Frank et al. [FBM+13] exploits scrolling as a biometric for classifying the user behavior. They used k-nearest neighbors and SVM-based classification to study and classify the correlation between 22 analytic features from touch traces. They were able to achieve positive results with equal error rate (EER) between 0% and 4% using a window size of 13 swipes. Recent studies have also highlighted the high classification accuracy of keystroke-based classifiers, which uses keystroke behavior, such as inter-key delay or key-holding time, to authenticate the users. Giuffrida et al. [GMCB14] used motion measurements from sensors such as accelerometer and gyroscope along with keystroke behavior to authenticate users. They characterized the typing behavior of users via unique sensor features and relied on standard machine learning techniques to perform user authentication. They evaluated their proposed system with 20 participants and found that sensor-enhanced authentication mechanism improved the accuracy of keystroke dynamics (EER is around 7%).

However, in contrast to knowledge-based password techniques such as PINs or pattern based schemes, which are deterministic, the classifications performed during behavioral biometrics are based on heuristics and thus can have high false rejection rates. Incorrectly denying access is extremely annoying and can quickly lead to user opt-out. Our proposed lock configurations aim to reduce this annoyance by giving users some time before displaying an authentication prompt instead of abruptly locking them out. During the usability evaluation of a behavior-based scheme, Khan et al. [KHV15] observed the usability issues arising from re-authentications due to false rejects. They listed some suggestions given by their participants to mitigate the negative usability effects of re-authentications. One suggestion was to not interrupt the user and instead send an email alert or take a picture of the perpetrator. Another, more secure suggestion that inspired this work was to authenticate the user in a smaller portion of the screen in parallel and to offer the user a grace period before the device locks out.

2.2.2 Context-Aware Authentication

Context-aware authentication may integrate many factors in authenticating users including their device usage behavior, surrounding environment or information from various contextual sources such as current location, proximity to Bluetooth, WiFi devices. For example, Shi et al. [SNJC10] proposed an authentication scheme that calculates an authentication score based on the user's recent activity such as the logs of messages sent, calls made and the location of the person using the device. They classified the events as good or bad and updated the authentication score accordingly. A good event could be a call made to a known number or the presence of the user at a known location. A bad event, on the other hand, could be a call made to an unknown number. The method of evaluating the authentication score was based on increasing the score when positive/good events occur and decreasing the score in case of bad events. Re-authentication of the users maybe required for this scheme in case many bad events take place leading to a low authentication score.

Egelman et al. [EJP+14] conducted a study to understand participants' perceptions regarding the sensitivity of the data stored on their smartphones. They found that 25% of the participants locked their devices to protect it from being accessed by friends and family members. Hayashi et al. [HRS+12] showed that all-or-nothing access to smartphones does not align with user preferences. The participants who locked their devices wanted about half of the applications to be unlocked and always accessible. In safe, known locations or around family members, the authors proposed to either disable authentication or to make it simpler to increase usability. We consider their recommendation to take into account the surrounding environment and the application being used to decide whether to lock the screen or not.

2.3 Summary

Our research focus is to investigate different configurations of a subset of the different types of authentication schemes (PIN and pattern-lock) for re-authentication purposes and not to address previously uncovered usability issues (e.g., time consuming, considered unnecessary for some cases [HVZF+14]) with these schemes. Though the recently proposed systems provide a viable alternative for authenticating users securely, we use popular authentication techniques for re-authentication in our study. We believe that the usability perceptions of the participants might get biased due to their missing experience with these new schemes. We aim to make sure that participants evaluate different configurations of an authentication scheme that they are already familiar with in our study. We take into consideration both the usability and security requirements of the users while designing the configurations. Our aim is to allow a convenient re-authentication experience for the users and at the same time make sure their personal data on the device including their passcodes remain secure. Considering the design recommendations given by earlier work [EJP+14, HRS+12], we deploy a context-dependent locking approach and evaluate the configurations for different scenarios and mobile applications.

Chapter 3

Lock Configuration

The main objective of this study is to investigate whether existing authentication schemes can be modified to make them more usable for re-authentication scenarios without significantly compromising on security. In order to make it convenient for users to re-authenticate, we modify the default lock configuration by varying the screen transparency and the delay of the appearance of the re-authentication prompt. To this end, we evaluate the effect of introducing two configuration parameters for existing authentication prompts: *time delay* and *screen transparency*. In this chapter, we outline the security and usability trade-offs introduced by these parameters, our constructions of re-authentication prompts with different configurations of these parameters and the usability expectations from our constructions.

3.1 Configuration Parameters

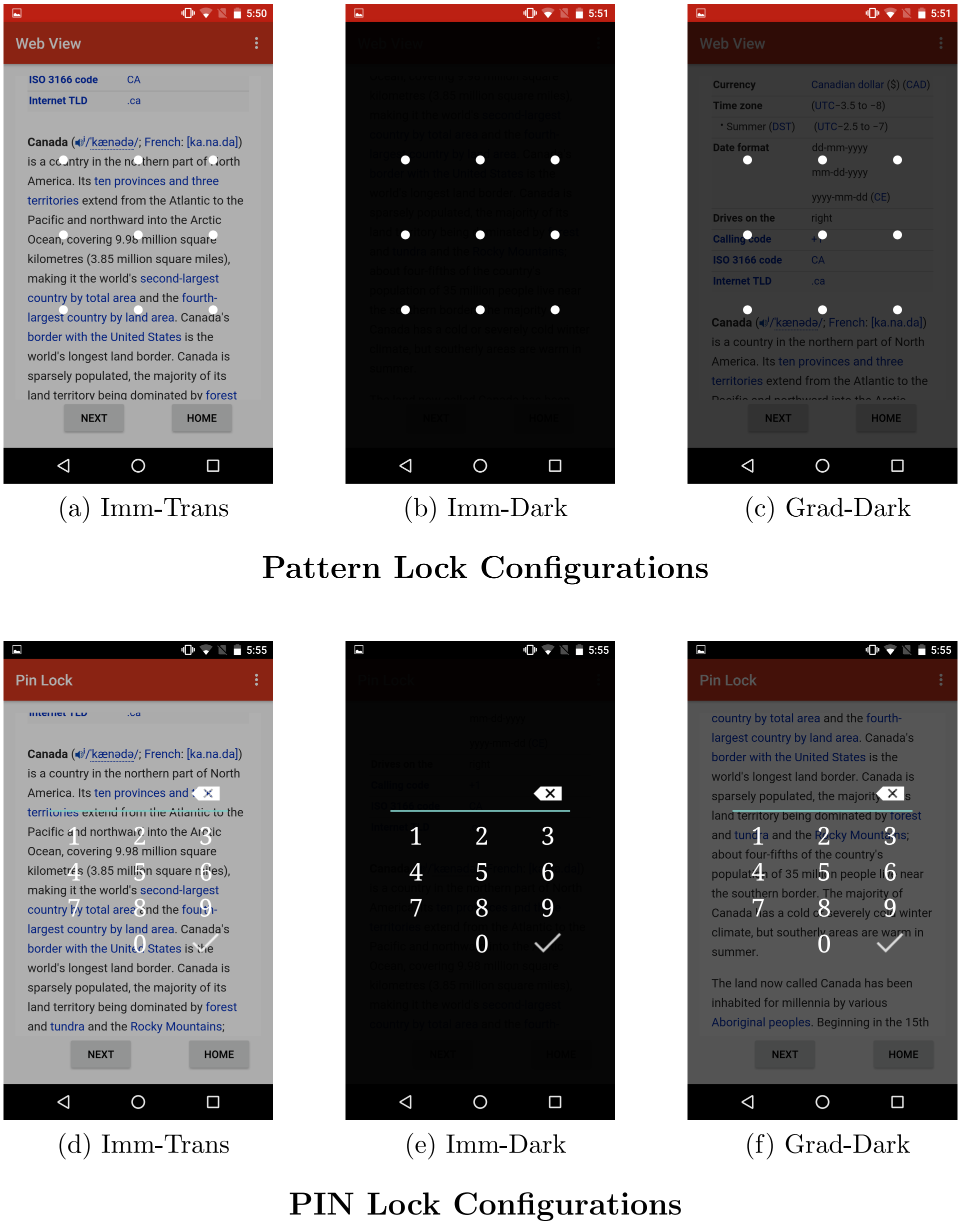
The *time delay* represents the delay in the appearance of the re-authentication-prompt after a configuration has been activated. The re-authentication-prompt refers to a numeric keypad in case of PIN lock and a 3-by-3 grid in case of a pattern lock which can be used to enter a passcode. The *time delay* variable supports two possible values: immediate lock (Imm-Lock) and gradual lock (Grad-Lock). In the Imm-Lock case, the re-authentication prompt appears immediately (without any delay) whereas for the Grad-Lock case, the re-authentication prompt appears after a predefined interval with a fade-in effect. During this fade-in, the user can continue to interact with the current task. The two possible values provide different usability and security trade-offs: the secure Imm-Lock bars the user immediately from interacting with the current task, while the less secure Grad-Lock is not abrupt and provides the user with an opportunity to interact with the current task during the grace period thereby potentially allowing the user to reduce the effect of interruption. For example, the user can finish reading a sentence or complete writing his thoughts down while writing an email before getting locked out.

For our experiments, we chose a four-second time delay. Our selection was based on the results from previous studies and our experiments with both shorter and longer delays. Ferreira et al.'s [FGK$^+$14] study on understanding micro-usage patterns for various smartphone apps revealed that 40% of the application usage lasts less than 15 seconds and is sufficient for a user to read or reply to a message. In a study conducted by Yan et al. [YCG$^+$12], they find that 50% of the smartphone interactions last fewer than 30 seconds. With such brief periods of interactions, it is therefore necessary to lock the device quickly to prevent any misuse. For the grace period, we considered and tested delays between two to seven seconds. During our empirical tests with four participants, we found that the four-second delay period allowed the participants to prepare for re-authentication prompts. The shorter delay values did not provide the users with enough time to prepare for the re-authentication prompt, whereas the longer delay values made the users anxious in anticipation of the re authentication prompt.

The *screen transparency* variable controls the visibility of the current task by configuring the background of the re-authentication prompt to remain transparent (Imm-Trans, see Figure 3.1a and 3.1d), be instantaneously dark (Imm-Dark, see Figure 3.1b and 3.1e) or gradually fade from transparent to dark (Grad-Dark, Figure 3.1c and 3.1f). Similar to the *time delay* variable, the three possible states of *screen transparency* provide varying degrees of security and usability. The Imm-Dark state is the most secure one because it hides sensitive data being displayed in the current task; however, the context-switch overhead should be the most in this case since the user's task is not visible anymore. The Imm-Trans state covers the other extreme where sensitive data displayed in the current task remains visible behind the re-authentication prompt; however, the context-switch overhead should be the least since the user's task remains visible while the user is interacting with the re-authentication prompt. The Grad-Dark state provides a grace period (default period length is 10 seconds) during which the user can authenticate to resume the task at hand; however, if the user fails to do so in a configurable amount of time, the background of the re-authentication prompt becomes dark thereby hiding the user's current task.

3.2 Proposed Configurations

The four configurations of re-authentication prompts that we construct using the different meaningful combinations of the two configuration parameters discussed above are as



(a) Imm-Trans (b) Imm-Dark (c) Grad-Dark

Pattern Lock Configurations

(d) Imm-Trans (e) Imm-Dark (f) Grad-Dark

PIN Lock Configurations

Figure 3.1: The proposed configurations with varying values for *screen transparency*. Figures (a), (b) and (c) show the three possible values when a pattern-lock based re-authentication prompt is used. Figures (d), (e) and (f) show the three possible values when a PIN-lock based re-authentication prompt is used. For the Grad-Dark configuration, the background of the re-authentication prompt gradually turns from transparent to dark.

follows:

1. **Immediate Dark, Immediate Lock (Imm-Dark-Imm-Lock):** We evaluate the default lock scheme on most Android smartphones to establish a baseline for when it is used for re-authentication. In this configuration, the re-authentication prompt appears immediately with a dark background, which completely hides the content of the current task, and the user can no longer view and interact with the current task. The re-authentication prompt asks the user to enter a PIN or pattern-lock and the user is able to access the current task again only after entering a correct passcode. This configuration was also used in the earlier work by Khan et al. [KHV15], as discussed in Section 1.3.

2. **Immediate Transparent, Immediate Lock (Imm-Trans-Imm-Lock):** The re-authentication prompt appears immediately in this configuration and the user can no longer interact with the current task once this configuration is activated. However, the background of the re-authentication prompt remains transparent, which allows users to observe the content of the current task.

3. **Gradual Dark, Immediate Lock (Grad-Dark-Imm-Lock):** In this configuration, the re-authentication prompt appears immediately and the user can no longer interact with the current task. Furthermore, the background of the re-authentication prompt is initially transparent and the contents of the current task are visible. The background of the re-authentication prompt gradually fades into a dark screen and hides the contents of the current task from the user. If the user manages to authenticate before the screen has darkened completely, this configuration keeps the user's current task visible in the background.

4. **Gradual Dark, Gradual Lock (Grad-Dark-Grad-Lock):** In terms of task visibility, this configuration is similar to the Grad-Dark-Imm-Lock configuration described above. That is, the background of the re-authentication prompt is initially transparent and then turns dark gradually. However, this configuration also allows the user to continue interacting with the current task for a grace-period of four seconds before the re-authentication prompt appears. During the four-second grace period, the brightness of the current task is reduced to indicate the forthcoming re-authentication prompt to the user. After the re-authentication prompt appears, the users can no longer interact with their task.

3.2.1 Other Possible Configurations

There are other possible combinations of the two configuration parameters, which we did not evaluate for re-authentication purposes. We discuss these configurations and our reasons for dismissing them below:

1. **Immediate Transparent, Gradual Lock (Imm-Trans-Grad-Lock):** The background of the re-authentication prompt turns and stays transparent allowing a visual clue of the task to the users. The re-authentication prompt appears after a delay giving them some time to interact with the task. This configuration is relatively insecure compared to the other configurations discussed in the previous section. An adversary can use the grace period to his advantage by browsing to a particular place in the task and then read through the displayed text even after the app locks him out.

2. **Immediate Dark, Gradual Lock (Imm-Dark-Grad-Lock):** The background of the re-authentication prompt immediately turns dark obscuring the content of the task. Also the re-authentication prompt appears after a delay as in the previous case. However, the grace period in this case is useless as it does not assist users in completing their tasks because the content of the background task is completely hidden.

3.3 Alternate Lock Configurations

While designing the re-authentication configurations, we considered a few other possible ways to re-authenticate the users securely and conveniently. However, due to limited time, we were not able to evaluate the other proposed configurations. Below, we list some of the other configurations we considered to re-authenticate the users.

3.3.1 Split Screen

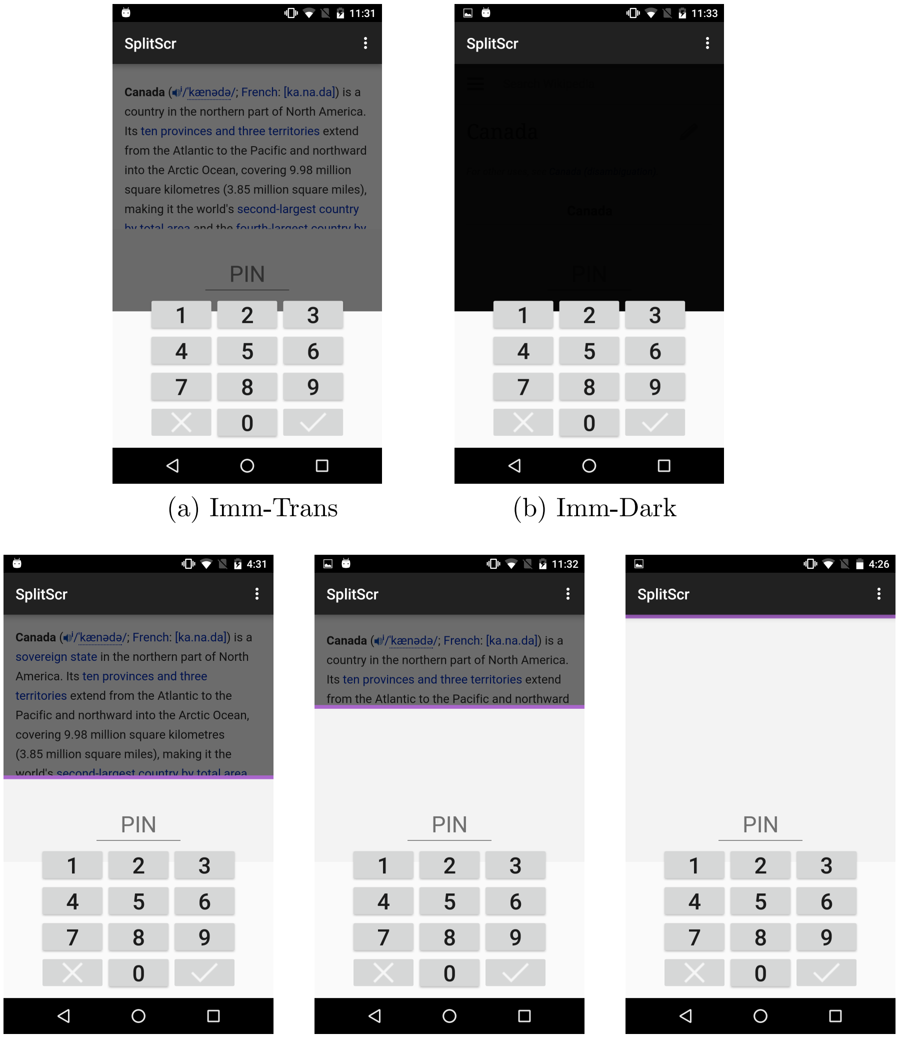
We designed and modified the lock configurations discussed in Section 3.2 by positioning the re-authentication prompt at the bottom of the screen instead of placing it in the center of the screen. The motivation behind this placement of the prompt was to assist the users enter the passcode comfortably because during this arrangement, the keys would be closer to the fingers while holding the device. Also in the split screen configuration, since the authentication prompt is placed at the bottom, a larger portion of the screen is available to the users to read/write any content during the grace period.

Similar to the re-authentication configurations mentioned in Section 3.2, we designed and implemented different split screen configurations with varying values of the *time delay* and *screen transparency* parameters as shown in Figure 3.2. While the *screen transparency* parameter for both Imm-Trans (Figure 3.2a) and Imm-Dark (Figure 3.2b) cases were similar to the originally proposed lock configurations, we modified the Grad-Dark (Figure 3.2c) configuration such that instead of gradually turning the screen dark, we used a vertical slider to gradually hide the content displayed on the top half of the screen.

The split screen configurations provides an alternative re-authentication configuration by placing the authentication prompt at the bottom of the screen. These configurations can be used along with our proposed configurations as well to give the smartphone users an option to select their preferred re-authentication prompt placement.

3.3.2 Notifying the User

We also looked into ways to notify the users when their device is being used by someone else and the re-authentication prompt is activated. Some of the ways to inform the users could be to send an email or an SMS to an alternate email address or to an alternate phone number pre-registered by the device owner. Since the device owner does not have the device, it is unlikely that he will be able to get the email/SMS if it is sent to their original email address or phone number. We can provide basic information to the user through the email or SMS such as the time and the reason for activating the re-authentication prompt. However, as reported in Section 5.3.2, we found that some of the study participants wanted to get detailed information about the device user instead, including the number of unlock attempts made, any identifying characteristic behavioral information of the user that can be used to identify him such as the touch input information or the browsing activity of the user around the time the re-authentication prompt was activated. Some participants even wanted the device to take a picture of the user using the front camera and save it on the device so that it can be checked by the device owner later. These notification techniques can be used in addition to using the lock configurations thereby providing additional information to the users apart from simply locking the device.



Canada (/ˈkænədə/; French: [ka.na.da]) is a country in the northern part of North America. Its ten provinces and three territories extend from the Atlantic to the Pacific and northward into the Arctic Ocean, covering 9.98 million square kilometres (3.85 million square miles), making it the world's second-largest country by total area and the fourth-largest country by

(a) Imm-Trans

(b) Imm-Dark

Canada (/ˈkænədə/; French: [ka.na.da]) is a sovereign state in the northern part of North America. Its ten provinces and three territories extend from the Atlantic to the Pacific and northward into the Arctic Ocean, covering 9.98 million square kilometres (3.85 million square miles), making it the world's second-largest country by total area

Canada (/ˈkænədə/; French: [ka.na.da]) is a country in the northern part of North America. Its ten provinces and three territories extend from the Atlantic to the Pacific and northward

(c) Grad-Dark

Figure 3.2: The proposed Split Screen configurations with varying values for *screen transparency*

3.3.3 Alternative Re-authentication Techniques

Previous studies [DLVZH09, DLVZN+13] have proposed alternate ways to authenticate users apart from the most commonly used authentication methods which ask users to enter a PIN or a pattern to authenticate themselves. Many proposed alternatives use the additional hardware capabilities of the device. For example, in VibraPass [DLVZH09], the user has to enter an incorrect digit or character whenever the device vibrates thus providing protection from a shoulder-surfing attack. Back of Device (BoD) Authentication proposed by De Luca et al. [DLVZN+13] is one of the other ways of implementing security by enforcing secrecy and confidentiality of the pass-code. Their proposed scheme allow users to enter their pre-selected gesture using the back of the device in order to authenticate themselves. Though the above techniques have a few limitations such as passcode memorability issues or hardware limitations, the above authentication techniques provide possible alternative ways to users to re-authenticate themselves.

3.3.4 Context-Based Configurations

Many systems use individuals' behavioral information to authenticate them. Over time, a system can collect information about the way a given user interacts with the device such as their keystroke dynamics, swipe inputs or motion patterns. In case the system notices any suspicious behavioral activity, it may lock the device to re-authenticate the user. Instead of locking the device right away, it can collect additional input from the user and re-classify their behavioral information to re-authenticate them. For example, for touch-based and keystroke-based behavioral biometrics, it can give additional tasks to the user to collect more touch and keystroke input from the users to classify their behavior. Such types of re-authentication tasks will be really helpful in cases where the device user is not the actual owner of the device but is close to the owner such as a family member or a close friend and therefore may already know the passcode. However, implementing these tasks so that it is convenient and least annoying for the users will be a challenge and presents a good opportunity for future work in this area.

Chapter 4

User Study

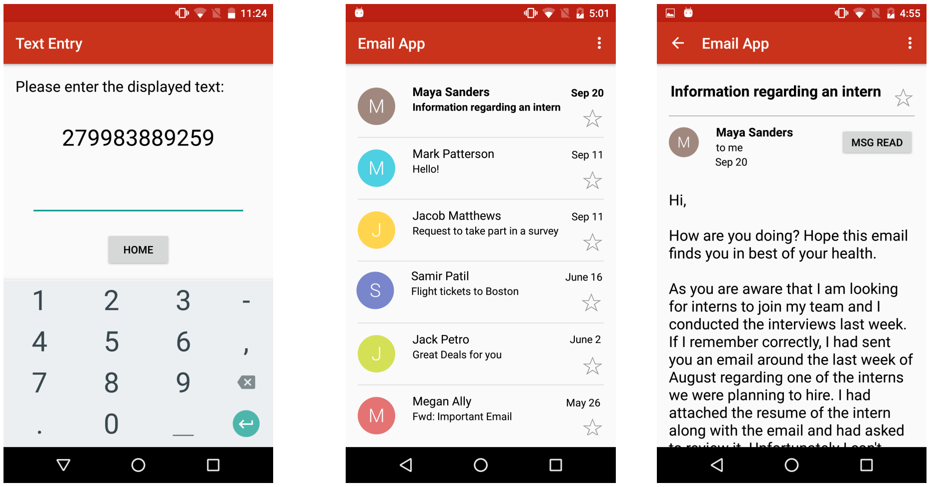
In this chapter, we outline our design of a user study to evaluate the four re-authentication prompt configurations. To measure the properties of each configuration, we perform a lab-based evaluation where participants are invited to experience each configuration by performing predefined synthetic tasks. After the users experience these configurations, they are asked to rate and provide qualitative feedback in terms of usability, security perception and their willingness to use these configurations. In addition to the user feedback, we measure the task efficiency, context switch overhead, and task error rate against each configuration. Our evaluation and feedback setup are designed to elicit the efficacy of these configurations for re-authentication in different scenarios. Our study was reviewed and has received approval from the Office of Research Ethics (ORE) of our university. We now provide details of our study design in terms of experimental setup and our methodology.

4.1 Apparatus

While several use cases exist for re-authentication (see Section 1.2), we choose implicit authentication (IA) as the representative use case in this work. Our choice of IA is motivated by the prior work of Khan et al. [KHV15] in the IA domain that highlights the issues with re-authentications in case of false rejects. To ensure that each participant experiences a certain number of false rejects, we use a simulated IA scheme, as was also done by Khan et al. In particular, our scheme simulates IA schemes based on a user's touch input or keystroke behavior.

For our experiments, we select two widely used authentication mechanisms on Android: a four-digit PIN and the Android pattern-lock (with the same constraints on possible patterns as in Android). The user interface of both schemes is similar to the Android lock screens (see Figure 3.1). We use Hai Bison's pattern lock library[1] to develop a pattern lock for each of the lock schemes and use a user interface similar to Android for developing the PIN lock.

The four re-authentication prompt configurations introduced in Section 3.2 are evaluated using two synthetic activities — a text entry activity and an email activity (see Figure 4.1). We choose these activities since they represent common smartphone activities (i.e., reading and composing emails and text messages or interacting with social media apps).



11:24

Text Entry

Please enter the displayed text:

279983889259

HOME

5:01

Email App

M	**Maya Sanders** **Information regarding an intern**	**Sep 20** ☆
M	Mark Patterson Hello!	Sep 11 ☆
J	Jacob Matthews Request to take part in a survey	Sep 11 ☆
S	Samir Patil Flight tickets to Boston	June 16 ☆
J	Jack Petro Great Deals for you	June 2 ☆
M	Megan Ally Fwd: Important Email	May 26 ☆

4:55

← Email App

Information regarding an intern ☆

M **Maya Sanders**
to me
Sep 20

MSG READ

Hi,

How are you doing? Hope this email finds you in best of your health.

As you are aware that I am looking for interns to join my team and I conducted the interviews last week. If I remember correctly, I had sent you an email around the last week of August regarding one of the interns we were planning to hire. I had attached the resume of the intern along with the email and had asked

(a) Text Entry Activity

(b) Email Activity

Figure 4.1: The activities performed by the participants during the user study. Figure (a) shows the text entry activity containing a twelve-digit number and Figure (b) shows the email activity.

- **Text entry activity:** This activity displays a twelve-digit number to the participants. It also contains a text box and the users are asked to enter the displayed number in the text box using the numeric keyboard of the device.
- **Email activity:** In the email activity, users are asked to read an email in an email app. The user interface for the email app developed for this activity looks similar to the Android Gmail app. Once a participant has read the email, they are asked

[1] https://bitbucket.org/haibison/android-lockpattern

to answer a multiple choice question related to the email on a laptop. The emails composed for this activity contained sensitive data, which emphasized the need to protect the emails from adversaries (see Figure 4.1b for an example).

The design of the text entry activity ensures that the interaction of the users with the app can be measured, which enables us to compute several metrics in terms of context-switch overhead and errors made by the users. For the email activity, since the emails contain sensitive material, the users performing the email activity should consider the security implications of a re-authentication prompt configuration in addition to its usability aspects.

These activities are bundled in two separate Android apps, which allow users to perform tasks. We define a task as completing the text entry or the email activity along with a mid-task re-authentication of the user using either the PIN or the pattern-lock in one of the four configurations. For the text entry task, the users are interrupted at predefined intervals, which are triggered based on the key presses by the users. The number of key presses required to trigger re-authentication vary across different text entry tasks for each user but it stays constant across users for those tasks for results to be comparable. Similar to the text entry task, the users are interrupted with a re-authentication prompt after a predefined number of swipes for the email task. The apps are instrumented to gather the timestamps of events, including input events by the user and the display and dismissal events of the re-authentication prompts. The apps also collects the errors made by the users for the text entry activity and during the re-authentication. We also log the user interactions, including the keystrokes and screen touch events, during the grace period for the Grad-Dark-Grad-Lock configuration. The data collected by the apps is instrumental in computing the task completion rate, context switch overhead and the error rate against each re-authentication prompt configuration.

4.2 Evaluation Methodology

We evaluate the four re-authentication prompt configurations using the text entry and email tasks. Each scheme was evaluated in a round that consisted of four text entry tasks and two email tasks. Each user was subjected to five rounds and in each round a different re-authentication prompt configuration was evaluated. For the first round, the participants performed the tasks without any authentication, which allowed us to establish a baseline. The participants were allowed to take a break between each task and each round. The order of the four re-authentication prompt configurations was randomly chosen for the participants.

The participants shortlisted for this study were invited for an hour-long lab-based study. The participants were first asked to fill a demographic survey which asked about their age, gender, and current occupation. They were then asked to fill a security preferences survey. In terms of security preferences, we asked the participants about their device locking habits, their preferred authentication scheme, and the adversaries that they wanted protection against. These pre-study surveys are provided in Appendix A. After the pre-study surveys, the participants were introduced to IA and the possibility of false rejects in IA using a short 3-minute Youtube video[2]. We also gave a brief description about the tasks and apps used during the study, and the different re-authentication prompt configurations. The participants were also told that false rejects were simulated for the purpose of this study. We gave participants the option to select their preferred lock scheme (PIN or pattern-lock) and a corresponding secret for the study. We did not assign participants a specific scheme to avoid any bias due to their inexperience with it. This design decision prohibited us to counterbalance the authentication methods. The authentication times varied across participants. To cater for this, we report within-subject relative differences instead of absolute values. The participants then experienced the different configurations in multiple rounds. After the completion of each round, they were asked to rate the usability and perceived security of the configuration that they experienced and to give an overall ranking in terms of their preferences by taking both the usability and the security of the evaluated configuration into account. Participants were also asked to indicate their preferences for the evaluated configurations under different device usage scenarios and were subjected to a semi-structured interview to gain further insight into their feedback. A researcher was present to respond to any questions the participants had.

4.3 User Feedback

The evaluated schemes trade off security for usability and since different users have different security preferences for different apps and different scenarios, we sought feedback from the users against four apps for three different scenarios. A previous study has shown that users prefer a strict security setting for financial and email apps, which contain highly sensitive data, whereas they prefer a relatively relaxed security setting for contacts and other utility apps [HRS+12]. We sought feedback from the users for four apps: a banking app, an email

[2] http://youtu.be/HUR2-bxBtI8

app, a photos app, and a contacts app. These apps are commonly used and contain varying levels of sensitive data of the smartphone user. The participants were asked to consider the following device usage scenarios with the aforementioned apps available on the device.

- **Bus Scenario:** The participants had to consider a situation where they are traveling on a bus and they accidentally leave their smartphone behind. A stranger picks up their device and starts using it.
- **Office Scenario:** This scenario asks the participants to consider a work environment where one of their colleagues starts using their device when it is left unattended. For this scenario, the apps on the device may be used for a limited time by someone known by the smartphone owner.
- **Home Scenario:** In this scenario, we asked the participants to consider that their spouse accesses their device while it is left unattended or when they are asleep. The number of adversaries is limited in this scenario as compared to the others and the users may or may not want to protect their data from their spouse.

A researcher presented the scenarios to the participants and was available during the interview to answer any questions participants may have. Participants were given sufficient time to consider the presented scenarios. For each scenario, the participants were told that the re-authentication prompt would get activated in case the system notices any suspicious activity. We also reminded them of false rejects and the fact that they may be subjected to re-authentication while they are using the device. In order to inquire about the security perception of an evaluated re-authentication prompt configuration, the participants were told that for the purpose of these scenarios, they should consider that only IA is protecting their device. Since different users may have different security preferences for each configuration and each usage scenario, we initially asked the users to establish the sensitive nature of the apps and usage scenarios. Then the participants were asked to provide feedback in terms of security perception, usability and preferred re-authentication prompt configuration for each of the four apps under each of the three device usage scenarios. The feedback questionnaire is provided in Appendix B.

Finally, at the end of the study, we conducted a short semi-structured interview (provided in Appendix C) to gain insight into participants' overall impression of the configurations that they evaluated.

4.4 Hypotheses

We had some expectations from our proposed re-authentication configurations before we began the user study. Our hypotheses were as follows:

Hypothesis 1 (H1): *Imm-Dark-Imm-Lock is the most obstructive therefore it should be the most annoying for the users.*

Hypothesis 2 (H2): *Since Imm-Dark-Imm-Lock provides no visual clues of the current task to the user, the task efficiency should be reduced during this configuration as compared to other configurations.*

Hypothesis 3 (H3): *Grad-Dark-Imm-Lock has similar properties as Imm-Trans-Imm-Lock but it provides additional security by making the current task of the user invisible after a predefined time interval. Therefore, it should score similar to Imm-Trans-Imm-Lock in terms of usability with a relatively better security perception.*

Hypothesis 4 (H4): *Grad-Dark-Grad-Lock enables the user to interact with the current task during the grace period and this may increase the task efficiency of the users.*

4.5 Participant Sample

We advertised the study through the university-wide mailing list and the university's graduate student research portal. The study was advertised with the title "Evaluating authentication schemes for smartphones" and we recruited only those users for the study who had some experience using smartphones. Interested participants contacted us via email and we asked them to come to our lab at a fixed time which was decided mutually. Each user study typically lasted between 40 minutes and an hour. We audio-recorded the responses of the participants throughout the study. A researcher was available throughout the study to answer any questions that the participants had. Each participant received $10 for taking part in the study.

We recruited 30 participants for the study (see Table 4.1 for their demographics). All the participants were students from our university. Our sample included 12 male and 18 female participants. Most of our participants belonged to the age group 21–25 years (57%) and either had a bachelor's degree (37%) or a master's degree (27%) from some university.

	N=30	
Gender	60%	Females
	40%	Males
Age	33%	Under 20 years
	57%	21–25 years
	7%	26–30 years
	3%	31–35 years
Education	37%	Bachelor's Degree
	27%	Master's Degree
	20%	Some college
	13%	High School/GED
	3%	Ph.D., Law or Medical Degree
Occupation	100%	Student

Table 4.1: Demographic information of the participants.

The majority of our participants (87%) reported that they used a lock scheme on their device. The security preferences of participants regarding their device lock usage are provided in Table 4.2. Out of the participants who locked their device (26/30), the majority of them used a pattern lock (50%) followed by using a fingerprint recognition (23%) to protect their device from being accessed by someone else. Almost all participants except one wanted to protect their device from strangers. We asked the four participants who did not lock their devices for their reason to do so: two indicated that they had nothing to protect, two wanted their emergency contacts to be available and one considered authentication to be inconvenient (multiple answers were possible).

Also we report the perception of the participants regarding IA in Table 4.3. As expected, all but three participants had no clue about IA. Also the three participants claimed that they had heard about IA in the past but did not have much knowledge about it. After the participants came to know about IA through the video, they responded positively to the idea of being implicitly authenticated and most of the participants were interested in using IA on their devices.

	N=30	
Lock device?	26 (87%)	Yes
	4 (13%)	No
Authentication scheme	13/26	Pattern Lock
	5/26	PIN (4 digits)
	6/26	Fingerprint recognition
	2/26	Password
Protecting from?	25/26	Strangers
	16/26	Friends
	14/26	Room-mate
	14/26	Coworker
	3/26	Spouse, own children
Reasons for not using any authentication scheme	2/4	Don't have any data to protect
	2/4	Allow others to use the device in emergency
	1/4	Takes time to unlock the device

Table 4.2: Device lock usage pattern of the participants.

	N=30	
Know about IA?	10%	Somewhat know about it
	90%	Have no clue about IA
Want to use IA? (After seeing the video)	4/30	Willing to replace current scheme with IA
	13/30	Willing to use IA in addition to the current authentication scheme
	10/30	May use it
	2/30	Not willing to use it
	1/30	Not sure

Table 4.3: Participants' perceptions about IA.

Chapter 5

Findings

We recorded and analyzed the data collected through the user studies and the interviews. We also transcribed the audio responses of the participants. We report both the quantitative and the qualitative results from the study in this chapter. For statistical significance, we used paired t-tests when comparing continuous data for the within-subjects condition such as the inter-stroke rate for each user between grace and non-grace periods. We used one-way ANOVA when comparing continuous data for the within-subjects condition for the four authentication configurations (e.g., context-switch overhead). We used chi-squared tests when comparing participants' responses to categorical Likert-type questions.

5.1 Quantitative Results

All 30 participants performed the text entry activity and the email activity while experiencing the four configurations. Out of the 30 participants, 18 participants chose to use a pattern-lock during the study, while the remaining participants chose to use a PIN. Participants were subjected to five rounds in total. During the first round, participants were not interrupted for re-authentication. This round was used to establish a baseline and we use the term *BaseRound* to refer to it. For the remaining rounds, participants tested one of the four configurations in each round.

During each round, participants completed four text entry tasks and two email tasks. They re-authenticated once for every email and text entry task during all rounds except *BaseRound*. The high rate of re-authentication is not representative of a real-world scenario; however, our motivation was to get participants acquainted with the configurations and to collect sufficient data to evaluate the metrics used in this section. Overall, during the study each participant re-authenticated themselves 16 times during the text entry activity (four times per configuration) and eight times during the email activity (twice per configuration). In total, we logged 120 re-authentication events, 120 text entry tasks and 60 email tasks per configuration for all 30 participants.

5.1.1 Effect on Task Completion Overhead

The task completion time is the time taken by the users to complete a text entry or an email task. It also includes the time taken by the users to re-authenticate themselves while evaluating one of the configurations. The task completion overhead is the additional time taken to complete a text entry task as compared to the *BaseRound* in which a user is not interrupted to re-authenticate. For the task completion overhead, we only take into account the text entry activity since the emails used for the email activity were of a different nature and length during each round. Our goal is to find if there are any re-authentication prompt configurations that assist the users in completing their text entry tasks faster.

We found that on average users took three to four seconds longer when they had to re-authenticate during a text entry task (see Figure 5.1). We conducted a one-way between subjects ANOVA to compare the effect of the four configurations on the task completion overhead, which indicated no significant differences across the four configurations *(F(3,116)=2.31, p=0.08)*.

Discussion: Our expectation that the Imm-Dark-Imm-Lock configuration is less efficient as compared to the modified re-authentication prompt configurations turns out to be incorrect. Though we did not find any significant differences in the performance of the configurations, the participants mentioned during the study that they felt that their performance was affected during the Imm-Dark-Imm-Lock configuration:

> “It kind of freaks me out because it is too sudden, it slows down whatever I was doing.” (P4)

5.1.2 Effect on Context Switch Overhead

Context switch overhead for the text entry task is defined as the time taken by the users to resume their text entry task once they have re-authenticated. Except the *BaseRound*,

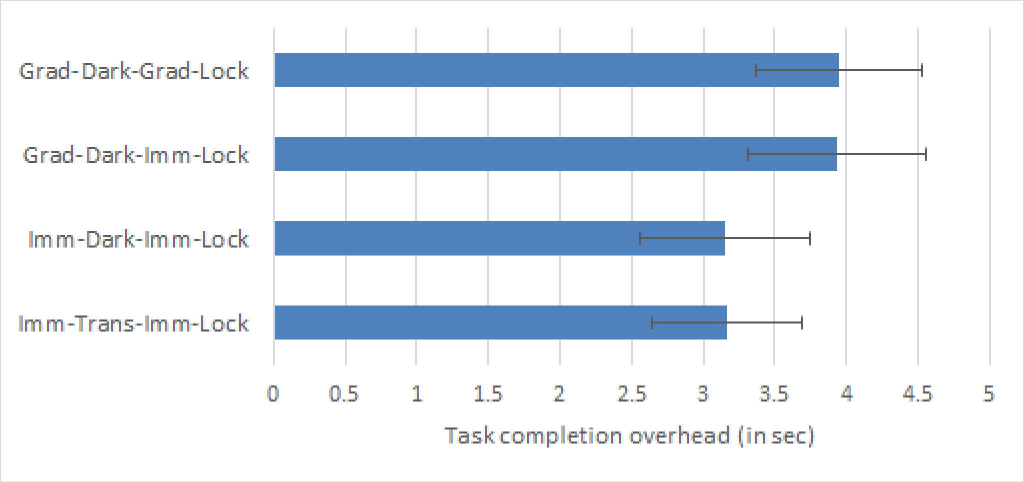


Figure 5.1: Task completion overhead time for the text entry activity relative to the *BaseRound* (Error bars represent 95% confidence intervals).

during all other rounds the participants were interrupted during the task and received a re-authentication prompt. The context switch overhead is represented by the time interval between the dismissal of the re-authentication prompt and the first key press on the text entry task once the re-authentication prompt has disappeared. It was not possible to compute this metric for the email task because after re-authenticating a user would complete reading the email text visible on the screen before interacting with the device. Despite our expectation that a visual of the user task in the background would reduce the context switch overhead, the context switch overhead across different re-authentication prompt configurations is comparable and contains no statistically significant differences *(F(3,116)=1.15, p=0.33)* as shown in Figure 5.2.

Discussion: While no statistically significant differences were observed, during the interviews, most users found the Imm-Dark-Imm-Lock configuration to be abrupt and reported that it was difficult to resume their task after re-authentication:

> “I lost my place [context] on what I was doing before [the lock appeared], so it is my least favourite. It would be too frustrating for me for everyday use, so I would rather take the risk.” (P9)
>
> “You can’t prepare for what’s going to come. It takes more time to pick up after unlock” (P10)

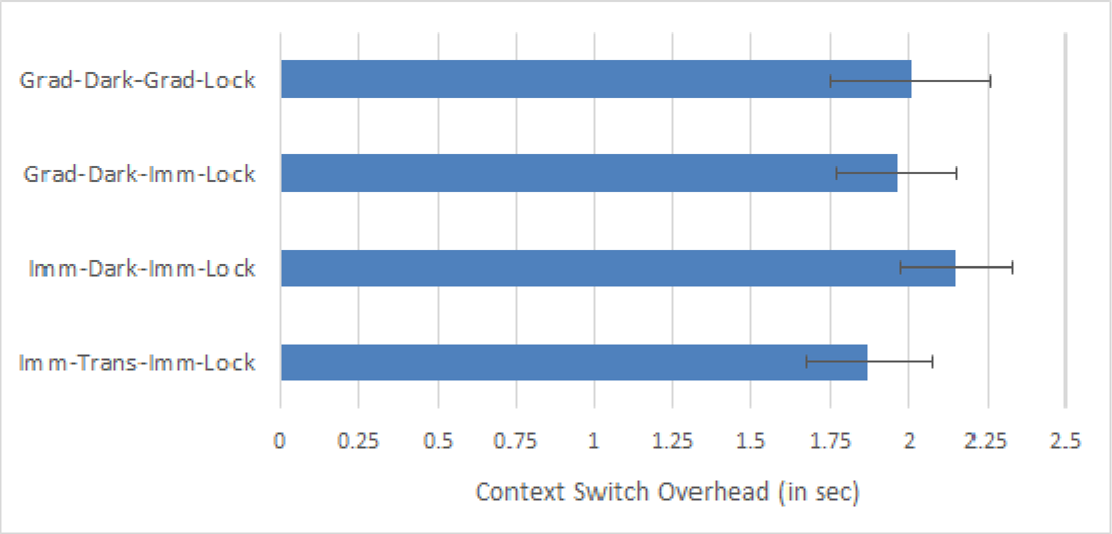


Figure 5.2: Context switch overhead time for the text entry activity (Error bars represent 95% confidence intervals).

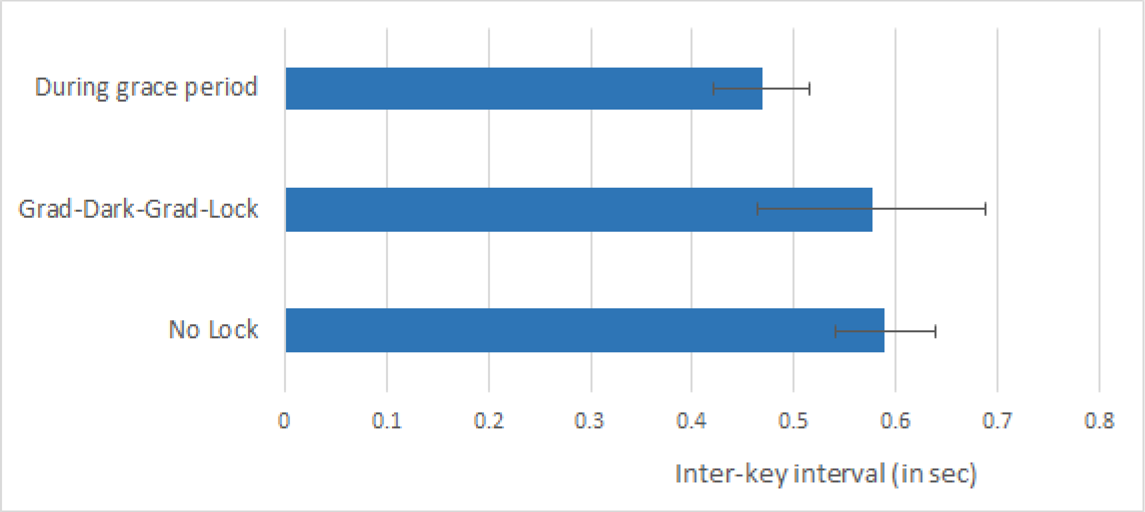


Figure 5.3: Inter-key interval for the text entry activity (Error bars represent 95% confidence intervals). The top bar represents the inter-key interval for the Grad-Dark-Grad-Lock configuration during the grace period.

5.1.3 Effect of Grace Period

We allowed a grace period of four seconds for the Grad-Dark-Grad-Lock configuration. During the grace period the participants could continue working on their task for four seconds before getting locked out. We observe that all participants took advantage of this grace period by continuing their work during the text entry activity. The average task completion time for the Grad-Dark-Grad-Lock configuration was 13 seconds and we found that on an average users entered 38% of the text during the four second grace period with some users entering up to 60% of the total text in the grace period. A similar trend was observed for the email task where 23% of the swipe events occurred during this period (average time to complete the email task for the Grad-Dark-Grad-Lock configuration was 41 seconds).

We find that the inter-key intervals (time interval between two consecutive key presses) of the users reduced significantly for the Grad-Dark-Grad-Lock configuration during the grace period. The average inter-key interval of users reduced by almost 60% during the grace period when compared to the average inter-key interval during the task (see Figure 5.3). A paired t-test was conducted to compare the inter-key interval between the grace and non-grace period for the same text entry activity for each user. The results show that inter-key intervals are significantly different between the grace and non-grace period *(t(29)=2.1, p=0.04)*.

Discussion: The participants took advantage of the grace period by attempting to quickly complete the text entry activity. They typed faster than their normal speeds during the grace period.

5.1.4 Effect on Task Error Rate

In case the input of the users mismatched the displayed text for the text entry task, we counted it as an error (with at most one error per task). Since each user completed four tasks per configuration, we compute the average number of tasks per configuration where users made one or more error(s). Overall, our results indicate that users made errors in 77 out of 600 text entry tasks presented to them. However, a one-way between subjects ANOVA for the task error rate across the four configurations and *BaseRound* indicates no significant differences *(F(4,145)=1.51, p=0.2)*. Similarly, while participants made errors in 43 out of 240 email tasks, the differences were not significant across the different configurations *(F(4,28)=0.28, p=0.84)*.

As shown in Figure 5.4, we find that the minimum number of errors were committed

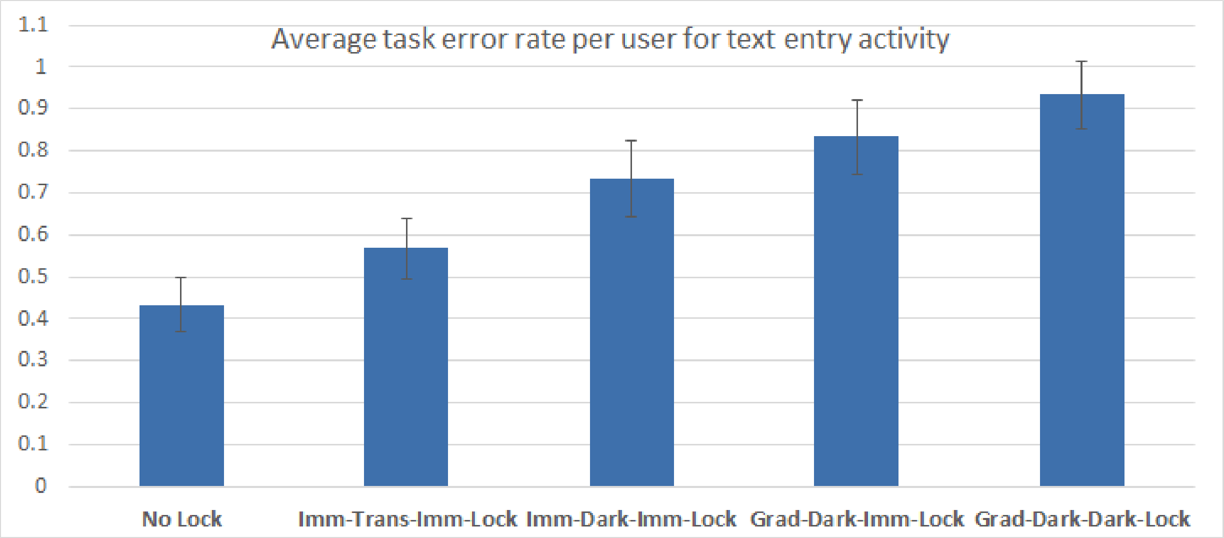


Figure 5.4: Average task error rate for the text entry activity. Vertical axis represents the task error rate per configuration. (Error bars represent 95% confidence intervals)

by the users during the *BaseRound* when they were not interrupted by an authentication prompt while entering the digits. Overall 20 out of 30 users did not make any error in any of the four tasks during the *BaseRound*. It was followed by 18/30 and 17/30 users not making an error in any of the tasks during the Imm-Trans-Imm-Lock and Imm-Dark-Imm-Lock configurations respectively. Interestingly, the Grad-Dark-Grad-Lock configuration had a very high task error rate with almost 66% users entering at-least one incorrect digit during at least one of the four tasks when this configuration was presented to them (see Figure 5.4). On an average, the task error rate during the Grad-Dark-Grad-Lock configuration was 70% higher than the task error rate during the *BaseRound* with some users even making 6-7 incorrect key presses during the same task.

Next we compute the error rate during the email task by verifying the answer selected by the users after they read an email. The users gave their responses to multiple choice questions after reading the emails. Each incorrect answer during a task contributed to the error rate for the configuration experienced during that task. Overall, a total of 43 email tasks had an error out of a total of 240 email tasks presented to them. However, similar to the text entry tasks, the differences were not significant across the different configurations.

Discussion: The task error rate among the configurations were comparable. Though the inter-key interval of the users during the grace period reduced significantly, it may have affected the task error rate compared to the other configurations.

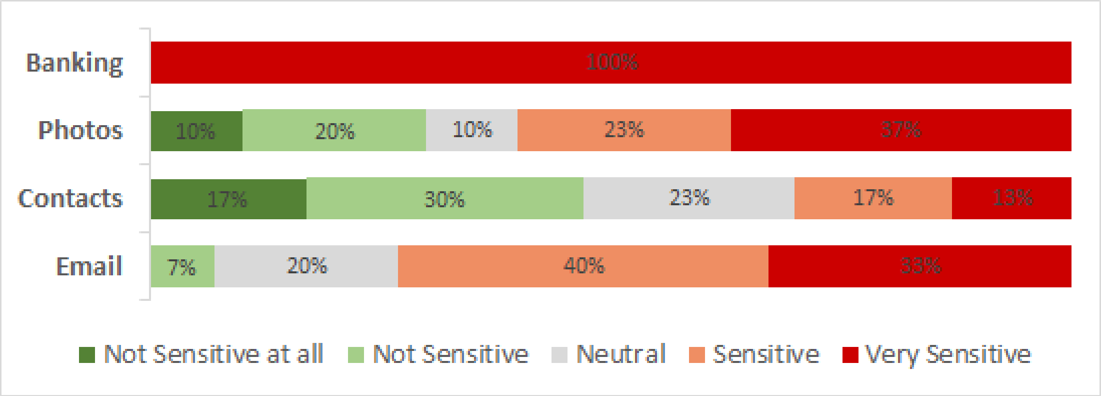


Figure 5.5: App sensitivity rating given by the participants.

5.2 Qualitative Results

For the apps evaluated in this work, 100%, 73%, 60% and 30% of the participants considered the banking, email, photo and contacts app to be sensitive, respectively. As shown in Figure 5.5, while the banking app was considered to be very sensitive by all the users, the sensitivity preference for other apps varied across users. The responses to the pre-study question regarding the adversaries that the participants (who used device lock) wanted protection against indicate that different scenarios require different levels of protection. As outlined in Section 4.5, almost all users wanted protection against strangers, which corresponds to the bus scenario. Corresponding to the office scenario, 54% of participants wanted protection against co-workers. On the other hand only 11% of participants considered that they needed protection against family members, which corresponds to the home scenario.

> "Contacts are not a big deal for me because most likely they [strangers] will not know who they [contacts] are." (P5)
>
> "Emails might be more sensitive especially about my jobs, so I don't want my co-worker to know anything about that." (P6)

We now present the findings from the feedback of the participants regarding the usability and security perceptions of the four re-authentication prompt configurations for each app in the different usage scenarios.

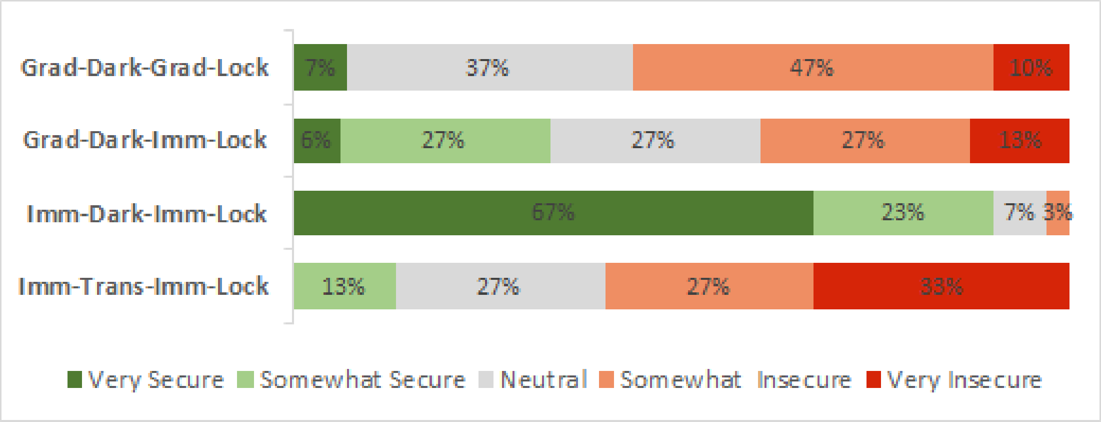


Figure 5.6: User perceptions of the security of the four re-authentication prompt configurations.

5.2.1 Security Perceptions

Figure 5.6 shows the security perceptions of the participants for each re-authentication configuration. Significantly more (57% more) participants thought that the Imm-Dark-Imm-Lock configuration was more secure than the other configurations ($\chi^2(3) = 151, p < 0.001$). Imm-Dark-Imm-Lock immediately hides the content on the screen to prevent the leakage of any sensitive information. Some participants indicated that they would take advantage of this increased security at the cost of usability for some apps:

> "If I am sending an important email, I do not want anybody else to look at it even for a second. It is annoying but it would be the most beneficial." (P13)

This was followed by the Grad-Dark-Imm-Lock configuration, which was considered to be secure by 33% of the participants. The participants considered the immediate locking of the app to be secure:

> "I liked the idea that how the lock appears at the start [during Grad-Dark-Imm-Lock], so if it is someone else, they can't enter any text message and they can't send anything compared to the last scheme [Grad-Dark-Grad-Lock] where they can do anything if they are fast enough" (P4)

The Imm-Trans-Imm-Lock and Grad-Dark-Grad-Lock scored poorly and only 13% and 7% of the participants considered them to be secure respectively. The security perceptions indicate that, as expected, the visibility of the user task in the background is perceived negatively by the users in terms of security. We now explore whether the less secure configurations were considered appropriate for some usage scenarios.

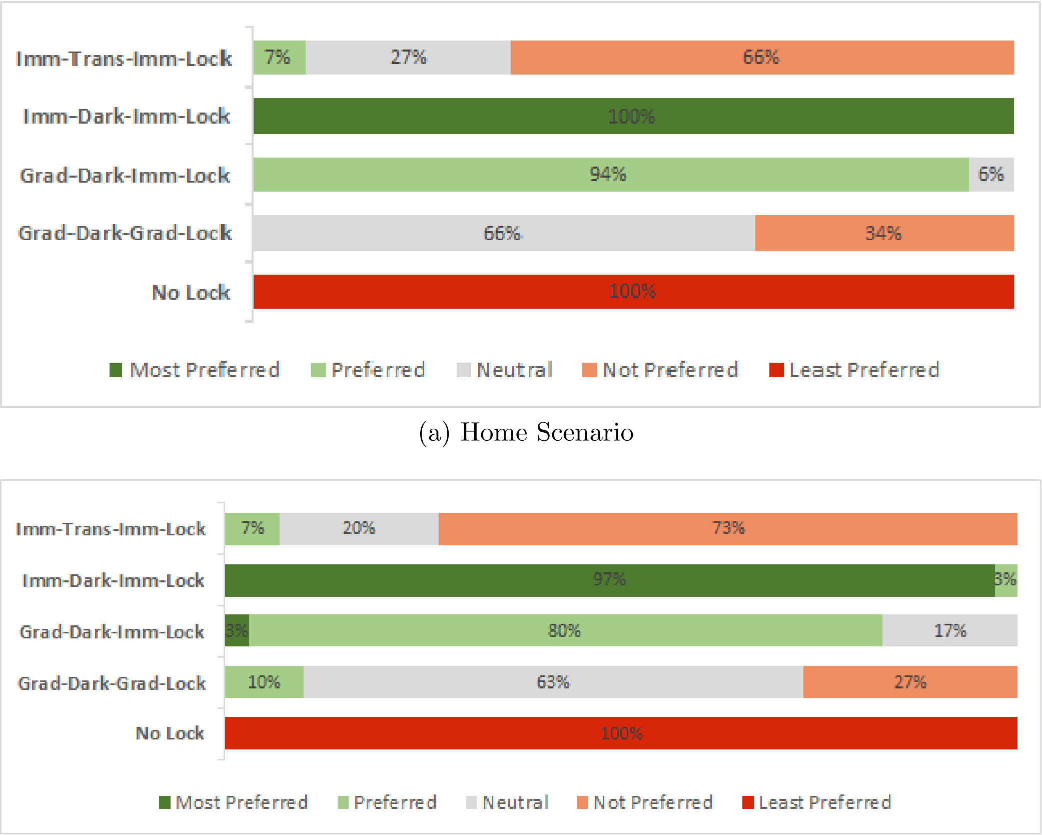
The Imm-Dark-Imm-Lock configuration was perceived most secure and all participants indicated that they would only consider using this configuration for their banking app on a bus and at the office (see Figure 5.7). On the other hand, for the home scenario, users had different preferences. 40% of the users indicated that they would still only consider using the Imm-Dark-Imm-Lock configuration for the banking app at home while 23% of the users indicated that they would prefer using the Grad-Dark-Imm-Lock configuration. Some of the user comments shed more light on the user preferences for the banking app:

> "Banking would be very sensitive, so I want it to get dark as quickly as possible." (P9)
>
> "Even with my partner, I won't feel completely secure with my banking app opened on my phone that is why I would prefer immediate dark." (P4)

The feedback from the users was inconclusive for the email app and there is no single configuration that users significantly prefer over the other for the different usage scenarios. On the other hand, for the photos app, the majority of the participants who considered the photos app to be sensitive preferred the Imm-Dark-Imm-Lock configuration at the office and for the bus scenario (Figure 5.8). For the office scenario, the participants who were very concerned about protecting their photos preferred configurations that obscured or gradually obscured the app, preventing it from being accessed by their co-workers:

> "I won't care about my photos with respect to a stranger but in office where its more professional environment with the people I know, I would increase the security of the scheme." (P12)
>
> "I have a lot of photos that are very personal and I don't want them [strangers] to see any part of them." (P6)
>
> "I might have already shared a lot of photos with my partner, so I would prefer a comfortable lock scheme." (P6)

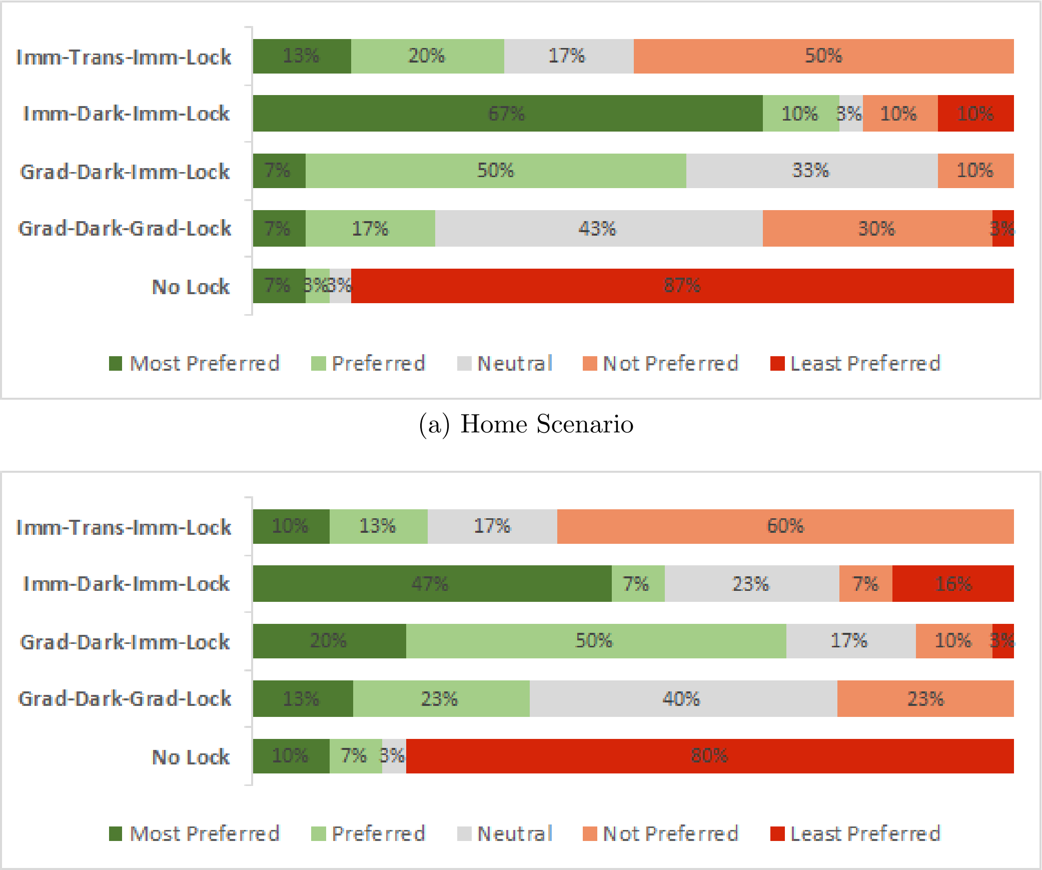
For the contacts app, the participants were willing to use configurations that provided device access for a period before locking them out. They wanted it so because this would allow a stranger to contact them in case they lost their device. The participants were less concerned about securing their contacts at home or office because they felt that they shared contacts with individuals at these locations.



(a) Home Scenario

(b) Office Scenario

Figure 5.7: User preference of the configurations for the banking app in different scenarios.



(a) Home Scenario

(b) Office Scenario

Figure 5.8: User preferences for the configurations for the photos app in different scenarios.

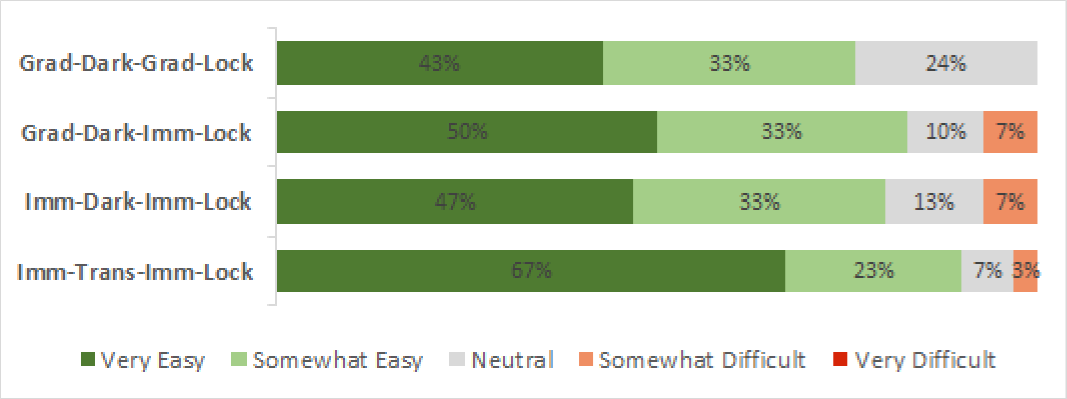


Figure 5.9: User perceptions on how easy it was to use the configurations.

> "If someone picked up my phone and they are looking at my contacts, they could try to return it to me through someone in my contacts, so I would choose something except the one that turns dark immediately." (P7)
>
> "For contacts, now there is an issue of privacy because these are people which they [office colleagues] might also know, so it is important that I protect their information but at the same time I don't want it to be very inconvenient for me when I look at the contacts." (P2)

Discussion: The participants considered the Imm-Dark-Imm-Lock configuration to be the most secure out of all four configurations. The inclination of the users while selecting the configurations are location- and app-based. While they prefer the Imm-Dark-Imm-Lock configuration to protect their banking information, they prefer to protect access to the photos app only at unknown locations. Users feel comfortable while browsing their device at home, and care less about using a more secure configuration except for the banking app.

5.2.2 Usability Perceptions

Our main goal while designing these configurations was to reduce the usability issues with re-authentication reported by Khan et al. [KHV15]. To this end, our configurations provided the users a visual of their tasks or a grace period to continue their work without disruption. We now present the perceived usability by the participants of these configurations.

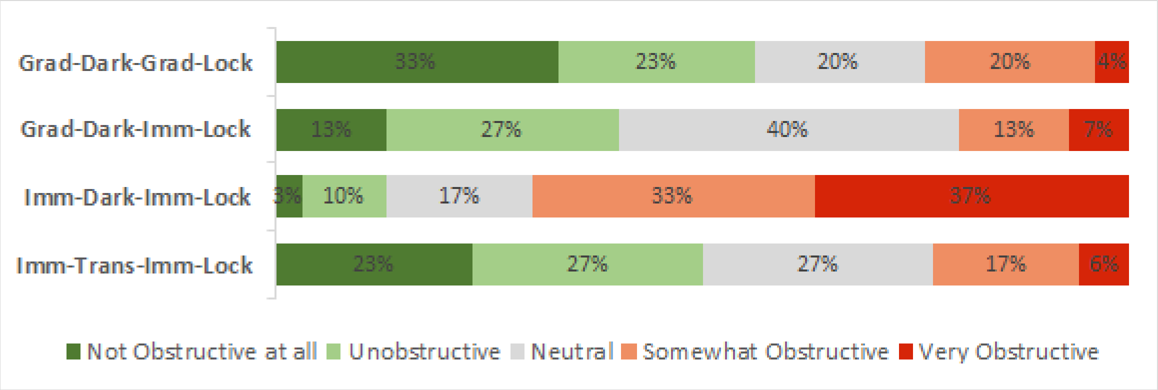


Figure 5.10: User perceptions regarding obstructiveness of the configurations.

We asked the users to rate the configurations in terms of ease of use. Figure 5.9 summarizes the responses of the users. We found that all configurations received a high rating in terms of ease of use and there were no statistically significant differences among the four configurations.In addition to a positive reception of the fade-in effect in Grad-Dark-Grad-Lock, users utilized the grace period to input data. Some of the users' comments include:

> "It helps you to continue typing and get your thoughts out. It didn't allow you to access the app though [after sometime] so it is a good balance between usability and security." (P16)

> "If I was in a rush to send an email to a client or my boss, I wouldn't want it to immediately get dark, I would want that buffer time to carry on my thoughts." (P4)

We also asked users how obstructive and annoying they thought each configuration was. Their responses (see Figure 5.10) indicate that significantly more participants considered the Imm-Dark-Imm-Lock configuration as more obstructive ($\chi^2(3) = 96, p < 0.01$). Similarly, Figure 5.10 shows that significantly more participants considered the Imm-Dark-Imm-Lock configuration was more annoying ($\chi^2(3) = 71, p < 0.01$). In terms of obstructiveness, 70% participants rated the Imm-Dark-Imm-Lock configuration as somewhat or very obstructive and a 67% of participants rated it as somewhat or very annoying (Figure 5.11). This explains why Imm-Dark-Imm-Lock was least preferred configuration for

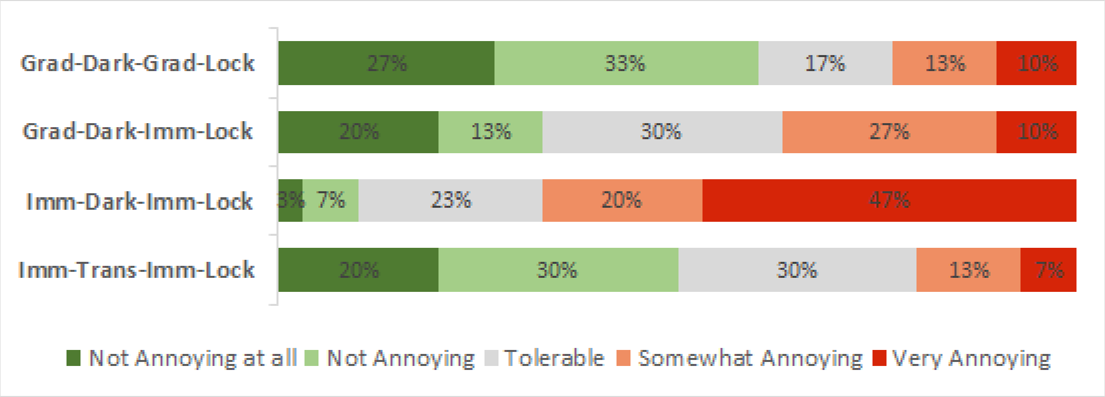


Figure 5.11: User perceptions regarding annoyance of the configurations.

the users for email (47%), photos (52%) and contacts (47%) apps for the home scenario. On the other hand, users positively perceived the gradual fading of the screen transparency and the delay of the authentication prompt. User comments that reflect these findings are:

> "I lost my place what I was doing before [the Imm-Dark-Imm-Lock configuration appeared], so it is my least favorite. It would be too frustrating for me for everyday use, so I would rather take the risk." (P9)
>
> "I found it [Imm-Dark] very annoying because it was really an abrupt interruption to me, others were not abrupt." (P8)
>
> "When you were explaining to me, I thought it would be difficult to wait for the lock [during Grad-Dark-Grad-Lock] but I guess it was nice to not lock right away, so you can continue what you are doing and wait for it to come up." (P12)

Discussion: While the Imm-Dark-Imm-Lock was considered most secure and was preferred for sensitive apps and risky scenarios, it annoyed the users. On the other hand, the less secure configurations were perceived to be more usable and users preferred those for less sensitive apps and for medium- and low-risk scenarios.

5.3 Overall Perceptions

We find that the participants not only easily adapt to using the modified configurations, but they respond positively as well to the idea of using our proposed modifications. The

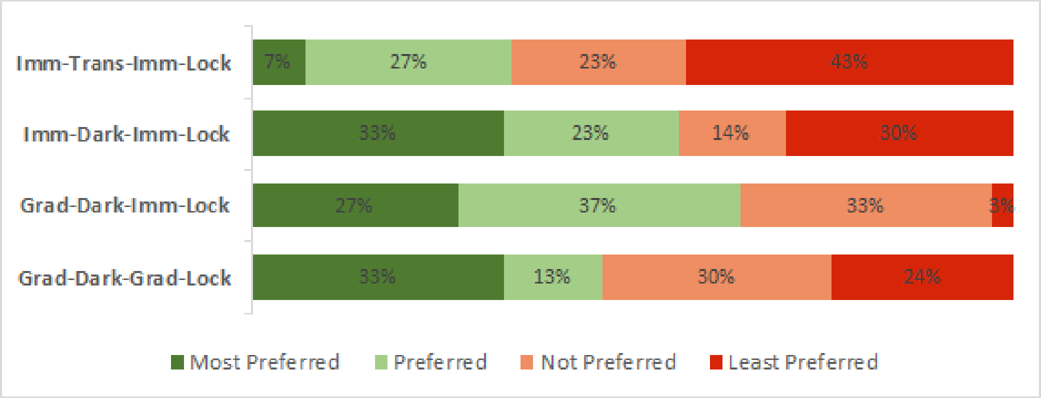


Figure 5.12: Ranking of the configuration given by the users.

participants liked the idea of getting a fade-in effect during the Grad-Dark-Grad-Lock configuration rather than being abruptly interrupted by an opaque re-authentication prompt. Also, they utilize the grace period during the fade-in effect to complete their current task. However, we found no significant difference when users were asked to rank the four configurations in the order of their preference while considering both the security and the usability of the configurations. Our results suggest that the users generally find it hard to select a particular configuration as their most preferred configuration (see Figure 5.12) and their choices are influenced largely by their perceived levels of the sensitivity of the apps they are using and their perceived security of the surrounding environment.

Location-based preferences: Users demand a high level of protection of the data on their device in high-risk environments, such as unknown-locations or around strangers/unwanted individuals. They consider the default configuration (Imm-Dark-Imm-Lock) to be the most useful in protecting their content in such scenarios. However, their preferences of the configurations changes considerably while they are using their device in a relaxed environment, such as around their family members or at home. They dislike using the default configuration for these cases and prefer using a more convenient configuration where they can access the content of the app for a while before being asked to re-authenticate.

App-based preferences: Almost all participants in our study were strongly motivated to use a very secure configuration to protect the information on their banking app. All of them reported that they were satisfied with the security provided by the default configuration and prefer using it for the banking app at all times. However, their preferred choices of the configurations for the email and the photos app varied depending on their individual perception of the sensitivity of these apps. For the email app, many participants liked the idea of getting a grace period while composing the emails as it allowed them some time to complete writing their thoughts down before finally getting a lock. Most of the participants did not seem to care about their data on the contacts app. Most of them were least bothered about using a secure configuration for the contacts app as they felt that the individual accessing their device will most likely not know any of their contacts.

We present the configuration preferences in terms of the percentage of users willing or not willing to use a particular configuration for various scenarios in Table 5.1. The table provides an overview of the participants' configuration preferences for the email, contacts and photos app in the bus, office and home scenarios. As mentioned in Section 5.2.1, while all participants preferred Imm-Dark-Imm-Lock for the banking app, for other scenarios participants had varied choices. For each scenario, we mention the percentages of users who are willing to use or are not willing to use a particular configuration. During the study, users who gave a rating of 1 or 2 on a 5-point Likert scale were considered to be willing to use the configuration while the users who gave a rating of 4 or 5 were considered not to be willing to use that configuration.

5.3.1 User Perception Regarding IA

At the end of the study, we again asked users if they were willing to use IA as a primary or a secondary authentication technique on their device. We find that overall, users had a positive attitude towards IA and were willing to use it (see Figure 5.13). They felt that IA would be able to provide protection from their family members or close friends who may already know the passcode of their device. Even some users who did not use any authentication on their device because they found it time-consuming, preferred being authenticated implicitly.

> "Yes, I would definitely want to use it because it is definitely more secure than my current scheme which is using passwords. There's a higher chance for someone to know my password than be able to imitate my swiping pattern." (P4)
>
> "For a person like me, I would like to use IA because I don't like to lock my phone because I am very lazy." (P8)

However, as anticipated because of the frequent authentication interruptions during the study, a few users were annoyed and raised some concerns regarding using IA. Though we told the users at the beginning of the study that the interruptions are being simulated, nevertheless the false rejects slightly impacted their attitude negatively towards IA. They

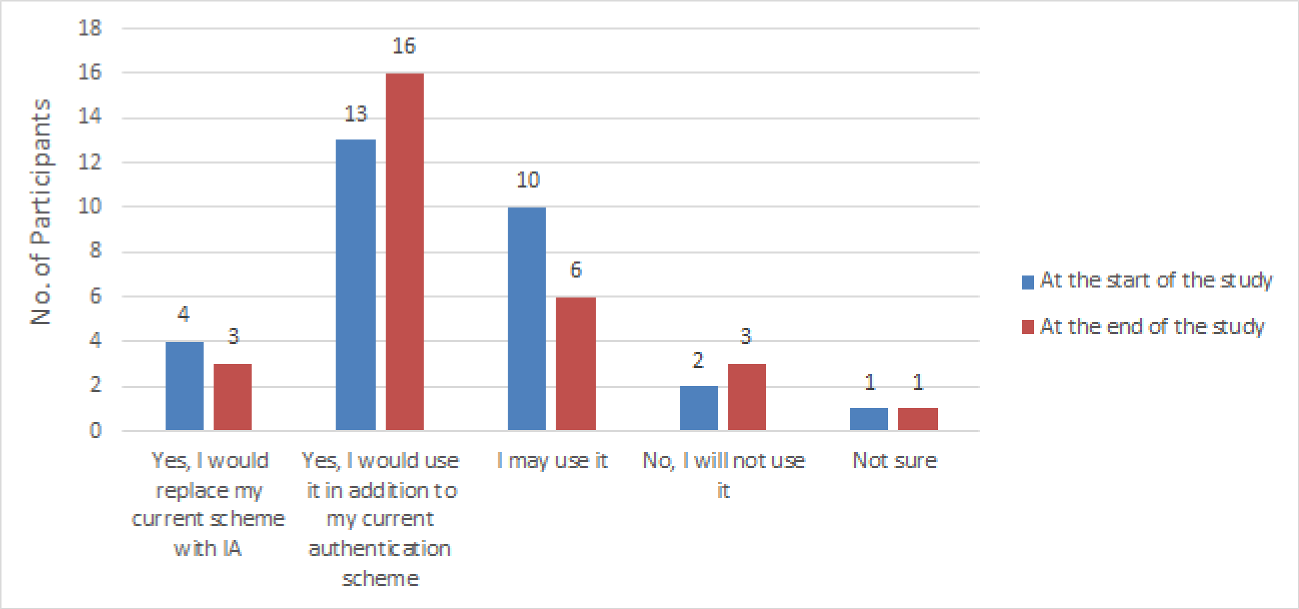


Figure 5.13: User preference regarding using IA before and after the study

wanted to use it only for sensitive apps and wanted the accuracy of the system to be improved. Some users even had concerns regarding the correct classification of their behavior in case of a finger injury or in case their swiping pattern changes after a few weeks or months.

> “I won’t use it because after going through the study, I found it really annoying. I use emails a lot, so [using IA] would be very annoying.” (P2)
>
> “My speed of swiping could change depending on how tired I am, if I have a finger injury, so false rejects are possible which is not very convenient.” (P4)

5.3.2 Alternate Re-authentication Techniques

We also asked the participants if there were alternative re-authentication techniques they would prefer using in addition to locking the device using the above configurations. The majority of the participants said they would like to get notified of the time and the reason for re-authentication through an email or an SMS. Some of them insisted on getting detailed information such as the browsing activity of the user or identifying behavioral characteristic information which can be used to identify the device user later.

> “Send a notification to the phone and preferably record the time whenever it gets locked out so at least afterwards whenever I pick up the phone I can try to guess who might be using it.” (P2)

> “It could send an email to the owner as soon as it locks it. If you leave it [device] in a place and a stranger finds it, it could set up an alternate contact information, so that if the phone is lost, it can call that number or send an email to contact.” (P6)

5.4 Summary

Overall, some of the findings from the study correspond to our expectations of these configurations. We find that our hypothesis H1 is correct. As expected, the participants consider the default configuration, Imm-Dark-Imm-Lock, to be the most annoying configuration. They do not prefer using the default configuration for less sensitive apps and surroundings. However, contrary to our expectations, we do not find a significant reduction in the task efficiency while the users complete the tasks using the default configuration and our hypothesis H2 is not supported. The task completion time for all the four configurations are comparable. We also find that Grad-Dark-Imm-Lock is perceived to be more secure than Imm-Trans-Imm-Lock, which aligns with H3. The participants respond positively to the idea of obscuring the content on their device in terms of security. Although the grace period during Grad-Dark-Grad-Lock does not increase the task efficiency of the users as we had predicted in H4, users find it really helpful and utilize this period to complete their tasks.

		Bus		**Office**		**Home**	
		Would like to use?	Would not like to use?	Would like to use?	Would not like to use?	Would like to use?	Would not like to use?
Emails	Imm-Trans-Imm-Lock	27%	**53%**	27%	40%	47%	26%
	Imm-Dark-Imm-Lock	**70%**	13%	**50%**	37%	10%	**70%**
	Grad-Dark-Imm-Lock	**60%**	7%	**67%**	13%	37%	40%
	Grad-Dark-Grad-Lock	37%	33%	**50%**	23%	**63%**	13%
	No Lock	7%	**93%**	7%	**86%**	43%	**50%**
Contacts	Imm-Trans-Imm-Lock	37%	47%	37%	20%	**50%**	33%
	Imm-Dark-Imm-Lock	40%	47%	23%	**64%**	7%	**80%**
	Grad-Dark-Imm-Lock	43%	17%	**53%**	24%	27%	36%
	Grad-Dark-Grad-Lock	**57%**	20%	**70%**	10%	**57%**	13%
	No Lock	23%	**70%**	17%	**83%**	**60%**	40%
Photos	Imm-Trans-Imm-Lock	33%	**50%**	23%	**60%**	44%	33%
	Imm-Dark-Imm-Lock	**77%**	20%	**54%**	23%	17%	**80%**
	Grad-Dark-Imm-Lock	**57%**	10%	**70%**	13%	34%	23%
	Grad-Dark-Grad-Lock	23%	33%	37%	23%	**57%**	16%
	No Lock	10%	**87%**	17%	**80%**	**50%**	47%

Table 5.1: Configuration preferences of the participants for different apps and scenarios. Values above 50% are in bold.

Chapter 6

Lock Library

In this chapter, we present the design and implementation of FireLock, our Android lock library, which can be used by developers to re-authenticate users while they are using their Android app. The library allows the developers to select the lock type out of PIN and pattern locks and the lock configuration out of the four configuration discussed in Section 3.2. We used Hai Bison's pattern lock library[1] for creating the pattern lock configurations and used a user interface similar to Android for developing the PIN lock configurations. Below, we discuss some of the design features provided by FireLock followed by the implementation of the library. We also provide an evaluation of the GPU performance of the lock library on a smartphone.

6.1 Design Features

- **Flexibility:** The lock library can be easily used along with any Android app by simply adding three lines to the code where the user needs to be re-authenticated. The developers can also select the type of configuration they want to use for various applications and scenarios.

- **Customizability:** The *time delay* and the *screen transparency* parameters described in Section 3.1 can be customized depending on the requirements. For example, for an email app, the value of the *time delay* parameter can be increased to leverage more time to the user in order to allow them to finish writing their thoughts down. On

[1] https://bitbucket.org/haibison/android-lockpattern

the other hand, since a banking app contains sensitive data, the *screen transparency* could have a small value to hide the data quickly.

- **Compatibility:** FireLock has been tested on smartphones with different screen sizes including Nexus 5 and Nexus 6. The configurations are compatible with both landscape and portrait orientation of the device screen.
- **Pass-code Security:** The lock library allows secure storage of the pattern and PIN secrets selected by the users. We provide retention of the secret even after the app is uninstalled or the cache is cleared by using cloud storage to backup the data and the user settings.

6.2 Implementation

6.2.1 Requirements

In order to use FireLock, the app developers will be required to download the library from the Github repository[2] and import it as a module in their Android app. The library can be used in any application running on Android 2.1+ (API level 7+).

6.2.2 First-time Usage

When the users install an app that uses FireLock for the first time, they will be prompted to select a lock type out of a PIN or a pattern lock for that app. Once they select one of the lock types, they will be asked to select a pass-code, which they will have to enter whenever re-authentication is required while they are using that particular app. The lock type and the pass-code can be later changed by the users using the navigation menu.

6.2.3 Pass-code Persistence

The library securely stores the pass-code (used for re-authentication) to protect it from being accessed by an adversary. For the pattern lock, we used the in-built encryption algorithm provided by Hai Bison's pattern lock library to securely save the user's choice of pattern. In order to securely store the PIN, we use PBKDF2WithHmacSHA1, the

[2] https://github.com/lalitagarwal/FireLock

native encryption algorithm provided by Android. A randomly generated pass-phrase and a randomly generated salt is fed into the PBKDF2 algorithm to generate the encryption key. The salt along with the pass-phrase are securely stored in an internal storage using SharedPreferences[3]. SharedPreferences securely stores the passcode and the user settings of any app in the form of a key-value pair and prevents other apps from accessing this data. However, the SharedPreferences data is lost once the app is uninstalled or the app-specific cache is cleared. In order to address this issue, we use BackupAgent[4] to store a copy of the app data to a remote cloud storage. This allows the restoration of the pass-code and the user settings in case the user re-installs the app or performs a factory reset. Therefore, an adversary will not be able to access the content of a locked app by re-installing it or by clearing the cache.

6.2.4 Re-authenticating Users

The lock library contains the `PinLock` and the `PatternLock` activity that contain the appropriate visual components using which the users can enter their pass-code. While the `PinLock` activity provides a numeric keypad to enter the 4-digit PIN, the `PatternLock` activity displays a 3-by-3 grid for entering the pattern. To activate one of the re-authentication configurations, the app developers need to use Intents[5] to call the `PinLock` or the `PatternLock` activity from the appropriate location in the code. For example, if a user needs to be re-authenticated every time they send a text message to a specific person, the following code can be inserted inside the `onClickListener`[6] method of the send button:

```
1 Intent pinLock= new Intent(this, PinLock.class);
2 pinLock.putExtra("category", <category>);
3 pinLock.putExtra("scrTransp", <seconds>); //Optional
4 pinLock.putExtra("timeDelay", <seconds>); //Optional
5 startActivity(pinLock);
```

The app developer needs to specify the lock configuration out of the four configurations mentioned in Section 3.2. Lines 3 and 4 are optional and can be used to change the default

[3] https://developer.android.com/reference/android/content/SharedPreferences.html

[4] https://developer.android.com/reference/android/app/backup/BackupAgent.html

[5] https://developer.android.com/reference/android/content/Intent.html

[6] https://developer.android.com/reference/android/view/View.OnClickListener.html

values of the *time delay* and the *screen transparency* parameters. While the default value for the `scrTransp` variable is ten seconds, the default value for the `timeDelay` variable is four seconds. Therefore, by simply including three lines of code in their app, developers can ask users to re-authenticate themselves.

6.3 Evaluation

We analyzed the performance of FireLock in terms of the resources being used. We measured the battery consumption using Android's Battery Historian tool[7] and found the library to be consuming less than 2% of the battery for a continuous one-hour usage.

We also analyzed the GPU resources being used by FireLock by measuring the number of frames rendered by each configuration. We activate each configuration ten times on a Nexus 5 device and use Android's GPU profiling tool[8] to record the number of frames rendered by the configuration during each activation. As shown in Figure 6.1, we find that among the four configurations, the maximum number of frames are rendered by the Grad-Dark-Grad-Lock and by the Grad-Dark-Imm-Lock configuration. Not surprisingly, the other two configurations, which did not use any animation feature rendered the least number of frames. We also observed the frames rendered by Android's Webview displaying a single Wikipedia page and a Webview displaying popular websites which included Facebook, Twitter, New York Times and IMDB and found it to be much higher than the frames rendered by each of the four configurations.

6.3.1 Use Case

We provide an implementation of FireLock for a behavior-based authentication scheme and test it on a messaging app. We use our library along with Itus[9], which is an Android library authenticating users using touch-based and keystroke-based classifiers. We integrate both FireLock and Itus with an open-source messaging application called Signal[10]. We provide separate implementations of touch-based and keystroke-based authentication in our Github repository[11]. The demo re-authenticates the users based on the classification

[7] https://developer.android.com/studio/profile/battery-historian.html

[8] https://developer.android.com/training/testing/performance.html

[9] https://crysp.uwaterloo.ca/software/itus/

[10] https://github.com/WhisperSystems/Signal-Android

[11] https://github.com/lalitagarwal/FireLock/tree/master/demo

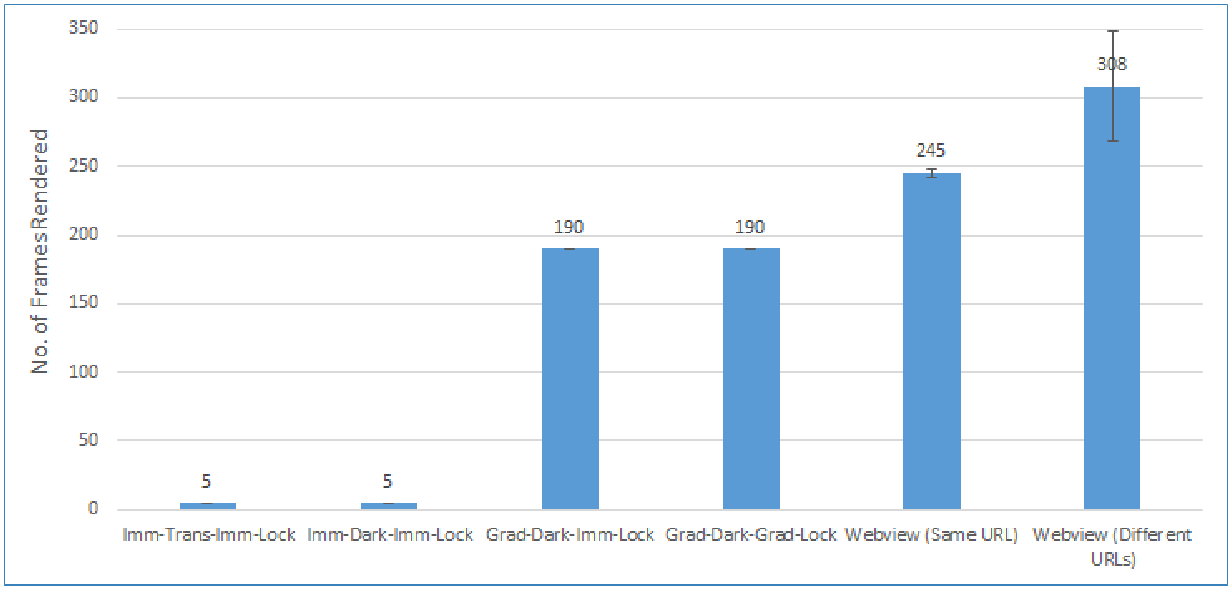


Figure 6.1: The number of frames rendered by the GPU for each of the four configurations and a Webview displaying a Wikipedia page. (Error bars represent 95% confidence intervals)

score as provided by Itus. In case of a low score, the re-authentication prompt is activated whenever the user presses the send button to send a text message. The user is unable to send the message unless they re-authenticate themselves by entering the correct pass-code.

Chapter 7

Conclusion and Future Work

We proposed two modifications to the default authentication prompts of two primary authentication schemes (PIN and pass-lock) to make them more suitable for re-authentication scenarios: a transparent authentication prompt and a time delay before the authentication prompt appears. In terms of task performance, the proposed configurations perform as well as the default configuration. However, the proposed configurations were perceived to be more convenient and less annoying by the users. We observe that user preferences of the configurations are largely context-based and there is no particular configuration that users want to use at all times. In terms of preference, while users want to use the default configuration (which obscures the app content) for highly sensitive applications, their choices for medium and less sensitive apps are influenced by their perception of the security of the surrounding environment and users preferred the proposed configurations for most of the less risky scenarios. We provide implementation details of our lock library, FireLock, and a real-world usage scenario for a behavior-based authentication scheme. The lock library provides a convenient way for developers to allow re-authentication in their apps by adding only three lines in their code. We believe that our findings will help in the design of re-authentication schemes that satisfy users' competing security and usability requirements.

The feedback from the user study gave us insights into the preferences of the participants, which will help in improving the design of the configurations and the implementation of the lock library. We present some of the suggestions given by the participants in the sections below.

7.1 Design

While the majority of users responded positively to the modified re-authentication prompt configurations, six participants found the fade-in effect to be annoying and recommended alternate ways to notify them before locking them out. During the interviews, these participants indicated that the cause of this annoyance was the wait for the authentication prompt to appear:

> "I would rather deal with the lock as quickly as I can so I can get back to using the phone." (P9)

One participant suggested that the source of annoyance was its resemblance to the interruption on the web for subscription-based content:

> "I don't like it at all because it reminded of those websites, where you are scrolling and it stops letting you read the content and that kind of is obstructed and annoying." (P7)

We sought suggestions from the participants during the semi-structured interview on how the re-authentication should be performed or improved. They proposed displaying a small timer at the top of the screen to indicate the time left before the users would be re-authenticated. Their comments were:

> "Maybe it can prompt you to type out a pattern on your phone without the visual obstruction, maybe like a small notification. It will warn you that it is going to lock and you can dismiss it by providing the secret." (P9)
>
> "Maybe instead of gradual fading, you can have a small timer up there on the screen near the status bar so that I should be expecting to get a lock screen." (P15)

Similar to the findings of previous research efforts on primary authentication schemes [EJP+14, HRS+12], our findings indicate that future experiments on user re-authentication should leverage app sensitivity and location information to ease the re-authentication burden. Whereas most participants of our study had similar security preferences in terms of the three scenarios (bus, office, home) evaluated in this study, there was a disagreement regarding the security preferences for the four apps. Therefore, re-authentication schemes need to provide users with a control to define these security preferences. A comment by a participant demonstrates the need for this:

> “You can have three different levels of security [depending on security preferences] and group your apps into those levels depending on the security you want for each app.” (P9)

7.2 Implementation

During the user study, we received some comments regarding the design and display of the re-authentication prompt suggesting that the delay before the appearance of the re-authentication prompt and the color of the screen during the fade-in effect should be customizable. Our current implementation only allows the app developers to change the values of the *time delay* and the *screen transparency* parameters. Future work in this area can focus on allowing even users to select their preferred configuration and change the parameter values according to their needs through the app menu.

Use Cases

We provided a demo of the lock library for a behavior-based authentication scheme. Deploying the lock library along with contextual-based authentication schemes, which authenticate users based on their location or based on their proximity to Bluetooth or WiFi devices can be an interesting work for the future.

APPENDICES

Appendix A

Pre-Study Survey

Before the study, participants were asked about their security preferences. In addition, we collected demographic information from the participants including their name, age group, gender, highest level of education and their current occupation.

A.1 Device Lock Usage

1. Do you currently use a lock mechanism for your device (smartphone/tablet)?

   (a) Yes; (b) No

2. **If they use a lock mechanism:** Which lock mechanism do you use to lock your device?

   (a) PIN Lock (4 digit or more); (b) Password Lock (characters and numbers); (c) Pattern Lock; (d) Fingerprint Recognition; (e) Face Recognition

3. **If they use a lock mechanism:** Who all do you want to protect your smartphone access from? (Select all that applies)

   (a) Coworker; (b) Friends; (c) Spouse; (d) Own Children; (e) Room-mate; (f) Other unwanted individual/stranger

4. **If they do not use a lock mechanism:** Why do you not use any authentication on your phone? (Select all that applies)

(a) It takes time to unlock the phone; (b) I don't have any data on my phone which needs to be protected; (c) No one would care about what's on my phone; (d) In an emergency, others can use my phone; (e) I have never thought about it

Appendix B

Study Questionnaire

B.1 User Perception of Individual Configurations

After the participants completed both the text entry and the email activity using one of the four configurations, we asked them to give a feedback on their experience with the evaluated configuration using the following questionnaire.

- Evaluate each of the following configurations that you will observe while doing the experiment. For each category, rate each configuration on a 5-point-Likert scale.

1. Immediate Dark Immediate Lock: Screen turns dark right away and PIN/Pattern appears
2. Immediate Transparent, Immediate Lock: Screen turns and stays transparent and PIN/Pattern appears right away
3. Gradual Dark, Immediate Lock: Screen slowly turns dark and PIN/Pattern appears right away
4. Gradual Dark, Gradual Lock: Screen slowly turns dark and PIN/Pattern appears after a while

(For each configuration, the participants had to give a rating to the following questions on a 5-point Likert-type scale.)

1. Assume someone picks up your smartphone and starts reading your emails. How secure do you find the configuration to protect your data in this scenario?

   (5- Very Secure, 1- Very Insecure)

2. How easy was it to use the configuration?

   (5- Very Easy, 1- Very Difficult)

3. How obstructive was the configuration?

   (5- Not Obstructive at all, 1- Very Obstructive)

4. How annoying was the configuration?

   (5- Not Annoying at all, 1- Very Annoying)

(Once the participant evaluated and rated all four configurations, we asked them to rank them in the order of their preference.)

- Rank the configurations in the order of your preference. Please take both the configuration's security and its usability into account.

  (1- Most Preferred configuration, 4- Least Preferred configuration)

B.2 Context-based Feedback of the Configurations

B.2.1 Sensitivity Ratings

Please provide a sensitivity rating of the following apps given how you use your mobile device and how sensitive you think each app is:

1. Email App
2. Contacts App
3. Photos App
4. Banking App

(5- Very sensitive, 1- Not very sensitive)

B.2.2 Scenarios

Now imagine the following scenarios and select which lock configuration you would prefer in each case. The lock configurations will get activated in case the system notices any suspicious activity. Please remember that since the system does not have 100% accuracy, it may assume you to be an adversary and you could encounter one of the lock configurations while you are using the device yourself. Assume that all of the apps below are protected only with implicit authentication and no other protection mechanism.

Bus Scenario

Imagine you are riding a bus and you accidentally leave your smartphone on the bus. A stranger picks up the device and starts using it, which gets detected by the implicit authentication protection mechanism. The stranger may launch different apps on your smartphone. For each app, the implicit protection mechanism could take a different action when detecting misuse. For each of the apps listed below, rank the order of preference of the lock configuration you would prefer with 1 being your most preferred lock configuration and 5 being your least preferred lock configuration.

Please remember that even you could encounter these configurations while you are using your phone on the bus.

(For each of the above scenarios the participants had to rank the configurations according to their order of preference for the following apps)

1. Views the emails in your inbox
2. Looks at the contacts on your smartphone
3. Views the photos stored on your smartphone
4. Accesses the banking app on your smartphone

Office Scenario

Imagine you are in your office and your boss urgently calls you for a meeting. You leave your phone on your desk and one of your office colleagues starts using your phone, which gets detected by the implicit authentication protection mechanism. Your colleague may launch different apps on your smartphone. For each app, the implicit protection mechanism could take a different action when detecting misuse. For each of the apps listed below, rank the order of preference of the lock configuration you would prefer with 1 being your most preferred configuration and 5 being your least preferred configuration.

Please remember that even you could encounter these configurations while you are using your phone in your office.

Home Scenario

Imagine you are watching television at home with your partner and you unknowingly doze off to sleep. Your partner realises that you are asleep and starts using your smartphone, which gets detected by the implicit authentication protection mechanism. Your partner may launch different apps on your smartphone. For each app, the implicit protection mechanism could take a different action when detecting misuse. For each of the apps listed below, rank the order of preference of the lock configuration you would prefer with 1 being your most preferred configuration and 5 being your least preferred configuration.

Appendix C

Semi-Structured Interview

We asked the following questions to the participants during the semi-structured interviews:

1. What was your overall impression of the configurations we showed to you? Did you find the configurations to be annoying?

2. Would you change anything about these configurations to improve their usability or security?

3. Did you like a particular configuration more than the other?

4. Did you dislike a particular configuration more than the other?

5. Will you be willing to use any configuration on your device for daily usage? Why? Why not?

6. Are there any particular scenarios where you think that these configurations will be useful to you?

References

[AK14] Md Tanvir Islam Aumi and Sven Kratz. AirAuth: Evaluating In-air Hand Gestures for Authentication. In *International Conference on Human-Computer Interaction with Mobile Devices & Services*. ACM, 2014.

[BFMN14] Joseph Bonneau, Edward W Felten, Prateek Mittal, and Arvind Narayanan. Privacy Concerns of Implicit Secondary Factors for Web Authentication. In *SOUPS Workshop on "Who are you"*. ACM, 2014.

[BHVZ$^+$16] Daniel Buschek, Fabian Hartmann, Emanuel Von Zezschwitz, Alexander De Luca, and Florian Alt. SnapApp: Reducing Authentication Overhead with a Time-constrained Fast Unlock Option. In *SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2016.

[CR14] Heather Crawford and Karen Renaud. Understanding User Perceptions of Transparent Authentication on a Mobile Device. *Journal of Trust Management*, 1(1), 2014.

[DLHH10] Alexander De Luca, Katja Hertzschuch, and Heinrich Hussmann. ColorPIN: Securing PIN Entry Through Indirect Input. In *SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2010.

[DLHVZH15] Alexander De Luca, Alina Hang, Emanuel Von Zezschwitz, and Heinrich Hussmann. I Feel Like I'm Taking Selfies All Day!: Towards Understanding Biometric Authentication on Smartphones. In *ACM Conference on Human Factors in Computing Systems*. ACM, 2015.

[DLVZH09] Alexander De Luca, Emanuel Von Zezschwitz, and Heinrich Hußmann. Vibrapass: Secure Authentication Based on Shared Lies. In *SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2009.

[DLVZN+13] Alexander De Luca, Emanuel Von Zezschwitz, Ngo Dieu Huong Nguyen, Max-Emanuel Maurer, Elisa Rubegni, Marcello Paolo Scipioni, and Marc Langheinrich. Back-of-device Authentication on Smartphones. In *SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2013.

[DZZ13] Benjamin Draffin, Jiang Zhu, and Joy Zhang. Keysens: Passive User Authentication through Micro-behavior Modeling of Soft Keyboard Interaction. In *Mobile Computing, Applications, and Services*. Springer, 2013.

[EJP+14] Serge Egelman, Sakshi Jain, Rebecca S Portnoff, Kerwell Liao, Sunny Consolvo, and David Wagner. Are You Ready to Lock? In *SIGSAC Conference on Computer and Communications Security*. ACM, 2014.

[FBM+13] Michael Frank, Ralf Biedert, En-Di Ma, Ivan Martinovic, and Dong Song. Touchalytics: On the Applicability of Touchscreen Input as a Behavioral Biometric for Continuous Authentication. *IEEE Transactions on Information Forensics and Security*, 8(1), 2013.

[FGK+14] Denzil Ferreira, Jorge Goncalves, Vassilis Kostakos, Louise Barkhuus, and Anind K Dey. Contextual Experience Sampling of Mobile Application Micro-usage. In *International Conference on Human-computer Interaction with Mobile Devices & Services*. ACM, 2014.

[FMP10] Jordan Frank, Shie Mannor, and Doina Precup. Activity and Gait Recognition with Time-Delay Embeddings. In *Association for the Advancement of Artificial Intelligence*, 2010.

[FZCS13] Tao Feng, Xi Zhao, Bogdan Carbunar, and Weidong Shi. Continuous Mobile Authentication using Virtual Key Typing Biometrics. In *IEEE International Conference on Trust, Security and Privacy in Computing and Communications*. IEEE, 2013.

[GMCB14] Cristiano Giuffrida, Kamil Majdanik, Mauro Conti, and Herbert Bos. I Sensed it was you: Authenticating Mobile Users with Sensor-Enhanced Keystroke Dynamics. In *Detection of Intrusions and Malware, and Vulnerability Assessment*. Springer, 2014.

[HDA+13] Eiji Hayashi, Sauvik Das, Shahriyar Amini, Jason Hong, and Ian Oakley. CASA: Context-aware Scalable Authentication. In *Symposium on Usable Privacy and Security*. ACM, 2013.

[HRS$^+$12] Eiji Hayashi, Oriana Riva, Karin Strauss, AJ Brush, and Stuart Schechter. Goldilocks and the Two Mobile Devices: Going Beyond All-or-nothing Access to a Device's Applications. In *Symposium on Usable Privacy and Security.* ACM, 2012.

[HVZF$^+$14] Marian Harbach, Emanuel Von Zezschwitz, Andreas Fichtner, Alexander De Luca, and Matthew Smith. It's a Hard Lock Life: A Field Study of Smartphone (Un)Locking Behavior and Risk Perception. In *Symposium On Usable Privacy and Security.* ACM, 2014.

[KAH14] Hassan Khan, Aaron Atwater, and Urs Hengartner. A Comparative Evaluation of Implicit Authentication Schemes. In *Research in Attacks, Intrusions and Defenses.* Springer, 2014.

[KHV15] Hassan Khan, Urs Hengartner, and Daniel Vogel. Usability and Security Perceptions of Implicit Authentication: Convenient, Secure, Sometimes Annoying. In *Symposium On Usable Privacy and Security.* ACM, 2015.

[KN14] Taekyoung Kwon and Sarang Na. TinyLock: Affordable Defense Against Smudge Attacks on Smartphone Pattern Lock systems. *Computers & Security*, 42, 2014.

[LZX13] Lingjun Li, Xinxin Zhao, and Guoliang Xue. Unobservable Re-authentication for Smartphones. In *Network and Distributed System Security Symposium*, 2013.

[MHK$^+$14] Markus Miettinen, Stephan Heuser, Wiebke Kronz, Ahmad-Reza Sadeghi, and N Asokan. Conxsense: Automated Context Classification for Context-aware Access Control. In *ACM symposium on Information, Computer and Communications Security.* ACM, 2014.

[MJB$^+$15] Nicholas Micallef, Mike Just, Lynne Baillie, Martin Halvey, and Hilmi Güneş Kayacik. Why Aren't Users Using Protection? Investigating the Usability of Smartphone Locking. In *International Conference on Human-Computer Interaction with Mobile Devices and Services.* ACM, 2015.

[MM13] Muhammad Muaaz and René Mayrhofer. An Analysis of Different Approaches to Gait Recognition using Cell Phone Based Accelerometers. In *International Conference on Advances in Mobile Computing & Multimedia.* ACM, 2013.

[MMC$^+$14] Shrirang Mare, Andrés Molina Markham, Cory Cornelius, Ronald Peterson, and David Kotz. ZEBRA: Zero-effort Bilateral Recurring Authentication. In *IEEE Symposium on Security and Privacy*. IEEE, 2014.

[RQSL12] Oriana Riva, Chuan Qin, Karin Strauss, and Dimitrios Lymberopoulos. Progressive Authentication: Deciding When to Authenticate on Mobile Phones. In *USENIX Security Symposium*, 2012.

[SDW12] Florian Schaub, Ruben Deyhle, and Michael Weber. Password Entry Usability and Shoulder Surfing Susceptibility on Different Smartphone Platforms. In *International Conference on Mobile and Ubiquitous Multimedia*. ACM, 2012.

[SLS13] Muhammad Shahzad, Alex X Liu, and Arjmand Samuel. Secure Unlocking of Mobile Touch Screen Devices by Simple Gestures: You Can See it But You Can Not Do it. In *International Conference on Mobile Computing & networking*. ACM, 2013.

[SNJC10] Elaine Shi, Yuan Niu, Markus Jakobsson, and Richard Chow. Implicit Authentication through Learning User Behavior. In *Information Security*. Springer, 2010.

[SPW13] Abdul Serwadda, Vir V Phoha, and Zhen Wang. Which Verifiers Work?: A Benchmark Evaluation of Touch-based Authentication Algorithms. In *IEEE International Conference on Biometrics: Theory, Applications and Systems*. IEEE, 2013.

[SWKW13] Florian Schaub, Marcel Walch, Bastian Könings, and Michael Weber. Exploring the Design Space of Graphical Passwords on Smartphones. In *Symposium on Usable Privacy and Security*. ACM, 2013.

[SYJ$^+$11] Weidong Shi, Feng Yang, Yifei Jiang, Feng Yang, and Yingen Xiong. Senguard: Passive User Identification on Smartphones using Multiple Sensors. In *IEEE International Conference on Wireless and Mobile Computing, Networking and Communications*. IEEE, 2011.

[TSK$^+$12] Shari Trewin, Cal Swart, Larry Koved, Jacquelyn Martino, Kapil Singh, and Shay Ben-David. Biometric Authentication on a Mobile Device: A Study of User Effort, Error and Task Disruption. In *Annual Computer Security Applications Conference*. ACM, 2012.

[TvO07] Julie Thorpe and Paul C van Oorschot. Human-Seeded Attacks and Exploiting Hot-Spots in Graphical Passwords. In *USENIX Security*, 2007.

[VZDDL13] Emanuel Von Zezschwitz, Paul Dunphy, and Alexander De Luca. Patterns in the Wild: A Field Study of the Usability of Pattern and Pin-based Authentication on Mobile Devices. In *International Conference on Human-Computer Interaction with Mobile Devices and Services*. ACM, 2013.

[VZDLBH15] Emanuel Von Zezschwitz, Alexander De Luca, Bruno Brunkow, and Heinrich Hussmann. SwiPIN: Fast and secure pin-entry on smartphones. In *Annual ACM Conference on Human Factors in Computing Systems*. ACM, 2015.

[WWB$^+$05] Susan Wiedenbeck, Jim Waters, Jean-Camille Birget, Alex Brodskiy, and Nasir Memon. PassPoints: Design and Longitudinal Evaluation of a Graphical Password System. *International Journal of Human-Computer Studies*, 63(1), 2005.

[XZL14] Hui Xu, Yangfan Zhou, and Michael R Lyu. Towards Continuous and Passive Authentication via Touch Biometrics: An Experimental Study on Smartphones. In *Symposium On Usable Privacy and Security*. ACM, 2014.

[YCG$^+$12] Tingxin Yan, David Chu, Deepak Ganesan, Aman Kansal, and Jie Liu. Fast App Launching for Mobile Devices using Predictive User Context. In *International Conference on Mobile Systems, Applications, and Services*. ACM, 2012.

[ZKM15] Nedaa Zirjawi, Zijad Kurtanovic, and Walid Maalej. A Survey about User Requirements for Biometric Authentication on Smartphones. In *IEEE Workshop on Evolving Security and Privacy Requirements Engineering*. IEEE, 2015.