PIANOTAP: IMPROVING TAP AUTHENTICATION ON MOBILE DEVICES

By

ANKIT SHAH

A thesis submitted to the
Graduate School—New Brunswick
Rutgers, The State University of New Jersey
in partial fulfillment of the requirements
for the degree of
Master of Science
Graduate Program in Electrical and Computer Engineering

written under the direction of
Dr. Janne Lindqvist
and approved by

New Brunswick, New Jersey
May, 2016

ABSTRACT OF THE THESIS

PIANOTAP: IMPROVING TAP AUTHENTICATION ON MOBILE DEVICES

By ANKIT SHAH

Thesis Director:

Dr. Janne Lindqvist

In this thesis, we study authentication on mobile devices by performing simple taps on the touchscreen. First, we have replicated a previously proposed system PassChords - a tap based authentication system. Based on the lessons learned, we present an improved system called Pianotaps, which theoretically provides an orders of magnitude larger password space. We conducted a preliminary informal user study of PassChords, Pianotaps and PINs towards understanding the benefits and drawbacks of these approaches. Our results indicate that Pianotaps provides enhanced security over shoulder surfing attacks while also being faster to authenticate with.

Acknowledgments

Firstly, I would like to thank my Master's Thesis advisor Professor Janne Lindqvist, for his guidance and motivation throughout this research project. This is my first research based project and it has not only helped me in learning relevant academic concepts, but has also helped me in developing my scientific thinking and research oriented aptitude. I am especially thankful to my advisor Dr. Lindqvist for his guidance and mentoring which has given me the confidence to finish my research and write this thesis.

I would like to thank all my fellow students in our lab. I have bumped into many technical problems and it would not have been possible to make it through without their valuable advice for my research. Talking to them has helped me make improvements to my research. I would like to thank Rutgers University for providing me this opportunity as a graduate student in which I have learned a great deal from experienced and knowledgeable faculty and staff. Over the past two years at this institution, I have many fond memories which will be cherished for life.

Last, but not the least, I would like to thank my parents, younger brother, friends, relatives and everyone for supporting me throughout writing this thesis and my life in general.

Dedication

For my Dad and Mom

Table of Contents

List of Tables

List of Figures

Chapter 1
INTRODUCTION

1.1 Motivation

Smartphones have become ubiquitous today and are embedded with the most advanced operating systems which help in bringing desktop like functionalities into the palm of any user. These advanced operating systems now come with more accessibility options than before. Innovative technologies have enabled people with visual impairment, hearing disabilities and inadequate physical or motor skills to use a smartphone with relative ease. Screen reading systems like VoiceOver in iPhone and Talkback in Android have revolutionized the way in which a visually impaired user can interact with a smartphone.

These phones are loaded with abundant personal data. There are many researchers who are trying to develop an authentication system with a global solution. The major hurdle faced by these researchers is the context of use of these authentication mechanisms. For example, a mobile banking application would rank security of the system higher than the ease of use. Vice versa is true while using an authentication system for any smartphone [1]. The process of mobile device authentication can be stressful for many users due to the challenges and annoyances of memorizing longer, difficult to memorize passwords. These complicated passwords are necessary however, in order to reduce the security venerability of the device being breached [6].

Firstly, I would like to discuss the various authentication systems currently available for smartphones in detail and the feasibility of using such systems as a secure mechanism for inconspicuous authentication.

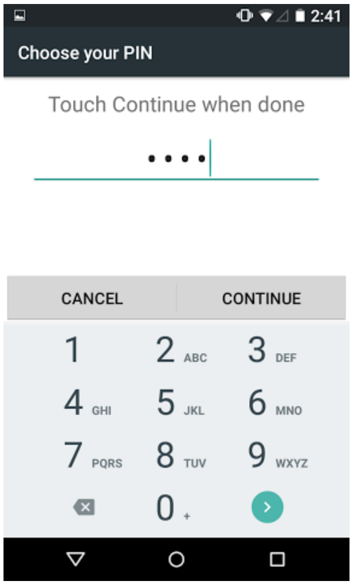


Figure 1.1: Choose PIN

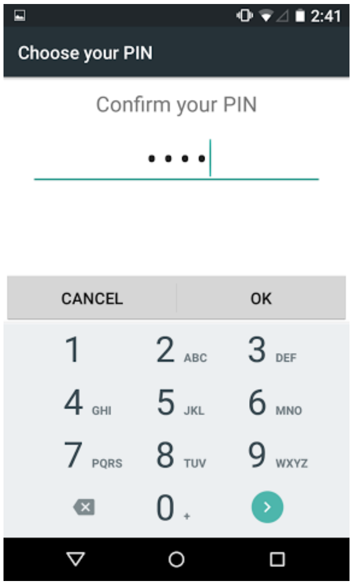


Figure 1.2: Confirm PIN

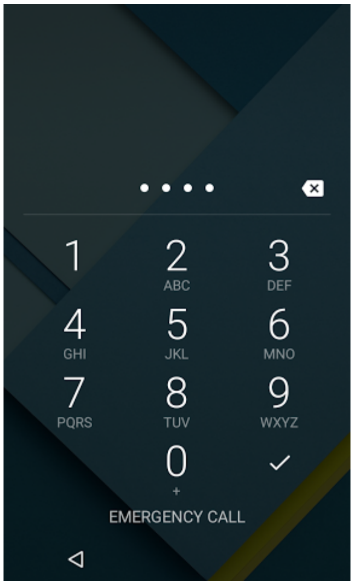


Figure 1.3: Enter PIN

1.2 Background

1.2.1 Authentication systems currently used in smartphones

An authentication system can be classified into three broad categories: 1. Something the user knows 2. Something the user is 3. Something the user has [19].

The PIN, password and Android pattern lock come under the category of something the user knows type of authentication. Biometric authentication systems like fingerprint, face recognition, voice recognition etc. come under the category of something the user is style of authentication system. The something the user has style of authentication includes smartcard systems and some auxiliary type of devices such as a smartwatch for authentication in the phone. Lets dive deep into each system and study them in detail.

Personal Identication Numbers (PIN) or Password Authentication

The PIN originated with the introduction of Automatic Teller Machines (ATM) in 1967, as an efficient way for banks to dispense cash to their customers. The passwords were in use even before the introduction of computing systems [1]. Thus, these methods have been in use for more than four decades. The currently used PIN system generally requires a minimum of four digits and a maximum of six digits (latest iteration of iPhones) which works out to ten thousand (four digits) and a million (six digits) combinations respectively.

Current screen readers help the visually impaired in their daily smartphone usage by allowing accessibility to different applications, making a call and sending text messages. As far as using PIN or password authentication by those who are who want to authenticate inconspicuously, the screen readers are too slow and vulnerable to many security attacks. A screen reader will read out the keys as and when the user presses them. The user will double tap the keys which are a part of the PIN or password. An attacker at a close distance can eavesdrop the PIN which is read by the screen reading software and use it to reduce the passcode combination search space [7]. Azenkot et al. [3] conducted a study which found that even with visually impaired users who were accustomed to using the PIN for authentication it took at least 8 seconds to unlock their phones. Even if the person uses headphones to avoid aural eavesdropping this technique is vulnerable to visual eavesdropping by people standing around when the user is unlocking the phone. The onlookers would be able to see what the user is entering with the exception some screen occlusion techniques such as those featured in Apples iOS. Visual eavesdropping is an obstacle for users who are not acquainted with the surroundings and the bystanders around them. Trying to prevent aural eavesdropping using headphones can also become a problem because the person would not be aware of the events taking place in the surrounding they are currently located.

The PIN authentication used in the study and additional analysis on the security of the PIN system will be discussed in the following chapters.

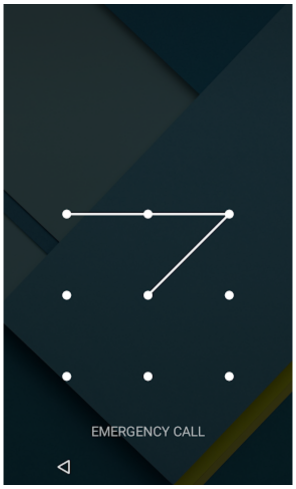


Figure 1.4: Android Pattern Lock

Android Pattern Lock Authentication

In 2008, Android introduced a unique authentication system to unlock a smartphone which required the user to connect the dots displayed on the touchscreen by swiping across a user selected sequence of dots in one finger stroke. This system is proven to be slower and more error prone than the PIN authentication, but there are millions of users still using this authentication method because it is easy to use and the users can make numerous patterns using their creativeness. This system is easy to crack using visual eavesdropping techniques like shoulder surfing and smudge attacks [1]. The system can be made less vulnerable by selecting the option of making the pattern invisible. By doing so, the user will not be able to see the lines connecting the dots nor will they see highlighted dots while drawing the pattern on the touchscreen. The drawback of enabling this option is that it is more prone to error because the user needs to be careful while connecting the dots since no form of line guidance is provided. These lines help the user know whether the dots are actually connected. A person will not be able to use this system inconspicuously, which is its biggest drawback. Thus, a lot is left to be desired from this authentication system.

We have not included this system as part of this study because it is not possible to use this system without looking at the screen i.e. inconspicuously.

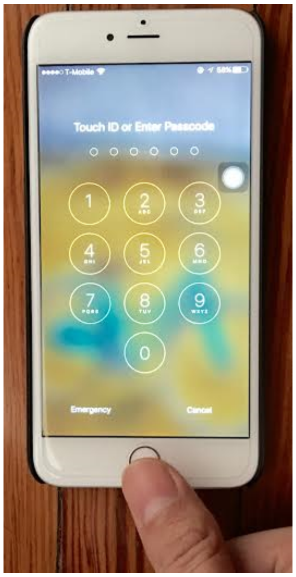


Figure 1.5: Biometric Authentication (Apple iPhone Touch ID)

Biometric Authentication

The latest smartphones come with advanced sensors to build state-of-the-art biometric authentication systems like face, fingerprint, voice and iris recognition and detection. These biometric systems are used in consumer electronic devices like laptops, electronic money and jewelry vaults, automobiles and security systems. Apples Touch ID technology offers a feature to unlock the iPhone and iPad using finger prints. It has inspired other technology firms like Samsung to embed a finger print recognition system for authentication in their smartphones and other devices as well. The finger print recognition API can be used with the other apps in smartphones like Apple and Android Pay, mobile banking apps and can be used as a substitute to any app requiring a username and password for login.

These biometric systems are easy to use, but they have many shortcomings which question the usefulness of these systems. De Luca and Lindqvist [1] have listed the drawbacks of using fingerprint authentication systems. The article mentions, injuries to the fingers like cuts and burns as well as environmental factors like moisture, sunscreen, ink, and dust can interfere with the fingerprint sensors. In addition, the fingerprints cannot be changed if they become compromised. Fingerprints are highly vulnerable to theft because government agencies and many other companies around the world store them. Fingerprint recognition can also be a safety hazard for a person using it because a malicious attacker may remove the users finger to access any information needed desperately [1]. There are videos on the internet demonstrating procedures in order to break into a system by smudge attacks and creating a fake finger.

Face detection systems are the most gullible authentication systems. The attacker can use a photograph of the user to authenticate without much hassle. Similarly, for voice recognition systems, the attacker can play the recorded voice command password of the user. This can be avoided by using advanced machine learning and artificial intelligence for face and voice recognition. This system would prove to be too expensive for use in any consumer electronic device like a smartphone or a laptop. Anthony et al. [8] has extensively conducted research on the security issues of biometric systems and mentions that if someone guesses your password, it is usually trivial to change that password. However, if someone gets hold of your fingerprint scan, you can only change this identifier nine more times, by switching fingers. In cases of retinal and palm scans, you can change this information once, and in the cases of DNA and facial scans once someone obtains your identifier, you need to change the authentication system to keep the malicious intruders out.

1.2.2 Tap authentication for smartphones

The authentication techniques discussed above are inaccessible to many disabled people and are susceptible to aural and visual eavesdropping. Thus, we need a system which is secure, user friendly and robust. Azenkot et al. [3] designed a tap based authentication system for smartphones known as PassChords. Their research study on the authentication system has proved that the tap authentication is better than the traditional PIN and password lock. The system implemented by them required the user to enter three similar or different combinations out of a total of fifteen possible combinations of finger taps to unlock the smartphone. We have implemented the PassChord authentication system using a similar Android app to test its efficacy against different attacks. In our implementation, which we call Pianotaps, we have made some improvements to the same system by enabling the users to choose their own single finger tap combinations using any one finger index, middle, ring and little, one at a time. In this system, tapping two or more fingers simultaneously is not allowed and a single finger tap must only be entered at a time. A detailed explanation of implementation of the both the apps will be discussed in the following chapters which will elucidate the idea of tap authentication. We have compared the PIN, PassChord and Pianotap systems by conducting a user lab study with participants using all three authentication techniques. A detailed analysis of possible threats and efficiency of these systems against shoulder surfing, aural and visual eavesdropping will be discussed in later chapters.

Chapter 2
RELATED WORK

2.1 Vulnerabilities in smartphone authentication

Tremendous research and development is being conducted in the field of smartphone authentication and security because smartphones have become an indispensable part of our lives. A PIN is a secret knowledge authentication approach and thus relies upon some knowledge the user already has. As of now we know that PIN authentication is prone to many different types of attacks due to the users themselves. Bad practices like selecting very easy four digit PINs like 1234 or 5555, sharing PINs with other users and writing them down and never changing them are predominant [24]. A survey was conducted by Clarke et al. [24] over a period of two years involving around 297 people who responded to a questionnaire. This survey was conducted to gain insight into the user attitudes towards current and future services and practices in relation to smartphone authentication. The initial findings of the survey confirmed that convenience factors are likely to win over security if users are allowed to make a choice. It was found that the users were in fact conscious about their security and they perceived security to be linked to the smartphone rather than it being carrier or operator dependent. The survey results found 45% of the respondents never changed their PIN. In addition, just 42% of the participants had only changed their PIN once and only 13% had changed their PIN more than once. It also found that 36% of the respondents used the same PIN for multiple devices and services. Therefore, PIN authentication has some major challenges with respect to its correct and secure usage. Denis et al. [7] studied entering PINs with screen readers and found that the audio feedback produced when typing out the PIN on the touchscreen is vulnerable to keystroke attacks by using a Hidden Markov model. Their work was focused on PIN systems used in smartphones, office door locks and ATM. They had developed an app which asked the users to enter random two digit numbers and it measured the inter-key tap timings. These timings were found to be very close to the audio recording of the aural feedback from the screen readers. An attacker who has access to the audio of a person entering their PIN could possibly evaluate the number keys pressed by the user because of the linear relationship between inter-key distance and the inter-keystroke timing delays. Anthony et al. [8] has analyzed the use of biometric security and has discovered that if someone gains access to a biometric identifier, that identifier is permanently compromised and if someone were to gain access to a database of biometric identifiers, the intruder could potentially access a persons fingerprint, retinal scan or other information which a user would like to keep secret. They suggested the use of a system which hashes the minutiae (features of the finger print used to identify the user) instead of hashing the fingerprint as it improves the security of the system. An alternative is to use multimodal systems which use more than one biometric identifier to identify the user, for example: watermarking a fingerprint with a facial scan. Thus, a single biometric identifier is not enough for enhanced security systems.

2.2 Alternative authentication mechanisms

Ravi et al. [6] worked on improving the existing tactile authentication system (TAS) for a blind person. The participants agreed that TAS would offer a more secure experience from onlookers, compared to PIN, as the stimuli are not visible to the observers. But the time usage by the adapted version of TAS was higher than the previous version of the system. The time taken to select and enter tactile authentication system must be reduced. The user generated gestures and doodles for authentication on touchscreen devices was studied by Michael et al. [2]. The gestures have a practically infinite number of combinations and are less vulnerable to shoulder surfing. Through their study they have found gestures are easy to remember and difficult to shoulder surf. In their system for trained users, there are no visual cues as to the appearance of gestures. The vulnerability of this type of systems are smudge attacks and privacy evading video recording attacks, where a camera captures the user performing the gestures. The need for a gesture managing app or software is expressed by the authors for managing gesture based passwords. It would be difficult to draw a complex gesture without any visual feedback by keeping the touch screen out of sight. Hirsto et al. [28] developed a light weight and inexpensive token based authentication system which can be used to unlock the smartphone. This is a form of hardware based authentication mechanism. They created two types of tokens, magnetic and acoustic to authenticate and unlock the smartphone. These tokens have an imminent threat of replay attacks where the attacker captures the transmission and replays it to gain unauthorized access.

2.3 Tap authentication mechanisms

Many security researchers are exploring finger taps on smartphone touch screens as an alternative to using a PIN, password or biometric authentication. Tap authentication systems are difficult to shoulder surf because most tap based systems do not have any visual feedback. It is difficult for an imposter to mimic the taping style of a user because touch screens come with pressure and acceleration sensors. Nan et al. [11] developed a tap based authentication system over the usual PIN lock. This system measured time, acceleration, pressure and size of user tapping of the PIN to distinguish between a genuine user and an imposter. It involved the use of a machine learning algorithm to build a tapping profile for a legitimate user. The system could not be used independently and was only to provide auxiliary security to

PIN authentication system. Though this implementation had a lower equal error rate than the similar systems developed earlier, it still faced imminent problems of change in user tapping behavior due to physical injury or restarting the training session for developing a new user profile when the user changes their PIN or password. The tap based authentication system we have replicated was developed by Azenkot et al. [3] and they conducted a study with 16 blind participants and had based their conclusions and inferences after a quantitative analysis of this study. Through their study it was found that most blind participants were unaware or not concerned about the potential security threats. None of the participants used the available password or PIN authentication system in their smartphones. The authors inferred that the tap authentication technique which they have referred to as PassChords is nearly three times faster than using PIN with Voiceover(iOS) or Talkback(Android). Diogo et al. [9] devised a tap based authentication system which took into consideration the down-time of the finger tap and the up-time between two consecutive taps. Through their study they established that entering tap combinations without visual feedback was easier and less error-prone than entering a PIN or Android Pattern Lock combinations. Their work however, does not include a test of memorability of tap authentication which is important considering the user needs to unlock the smartphone every time he or she wants to contact someone or access some information. The TapSongs [4] presented a new tap based rhythmic authentication using a single binary sensor. It matched the rhythm of tap down/up events to a jingle timing model created by the user. Similar to the earlier authentication, TapSongs can also be entered without too many errors without visual feedback from the device. This authentication system uses the absolute match criteria and learns more about the tap input from successful logins. TapSongs had the ability to distinguish between the rhythmic tapping by a legitimate user and imposter. As with the earlier techniques the authors question the memorability of the TapSongs if they have not been entered for days or weeks. Ahmed et al. [14] developed a hybrid tap and gesture method for authenticating smartphone users. They used the touch screen keypad for taps and gestures. A user could select a combination which could contain tapping on any key or swiping up, down, left or right, on any keys of the digits 0-9. It offers a total of 6250000 unique combinations, thus, is significantly more secure than PIN or digital clock authentication technique which has less possible combinations. Through their pilot study they found that this technique is was slower and more error prone than other techniques, but with practice it could become much faster and more accurate.

Chapter 3
ALGORITHMS AND IMPLEMENTATION

3.1 PIN Authentication App

The PIN authentication app was developed specifically to emulate the working of a PIN code authentication system. It works like a usual PIN application in Android or the iOS operating system. Initially, the user selects the PIN to be kept as the combination for the lock. The user then re-enters the same combination to confirm the PIN combination. Now, while unlocking the phone, the app automatically pops up on the touchscreen and asks the user to enter the PIN. The user will have to enter the PIN combination selected to unlock the smartphone. If the user enters a wrong PIN combination then the app notifies the user that the PIN entered was incorrect. The user has to enter the correct PIN to unlock the phone. The person would only be authenticated after the correct PIN is entered. To emulate a user using the app inconspicuously, Voiceover or Talkback was used to make the user aware of the exact key being pressed while keeping the phone under the table. This will enable the user to enter the PIN slowly, but surely. The app stores numerous types of sensor data like accelerometer, light, linear acceleration, gravity, gyroscope, magnetic field, pressure, proximity, humidity and temperature. The table also consists of a column which has the timestamp of the sensor data and the result if the PIN entered was correct or incorrect. In the results column the other possible entries include CREATION STARTED which signifies the start of creating the PIN and CREATION ENDED which means the PIN has been selected and saved by the user. The difference between the timestamps of CREATION STARTED and CREATION ENDED enables us to

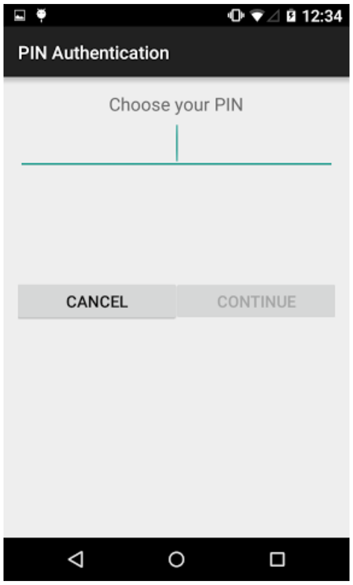


Figure 3.1: Choose PIN

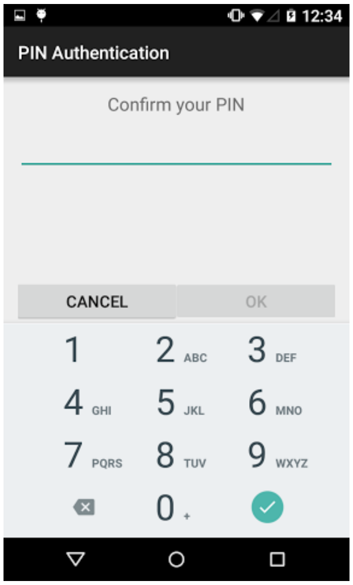


Figure 3.2: Confirm PIN

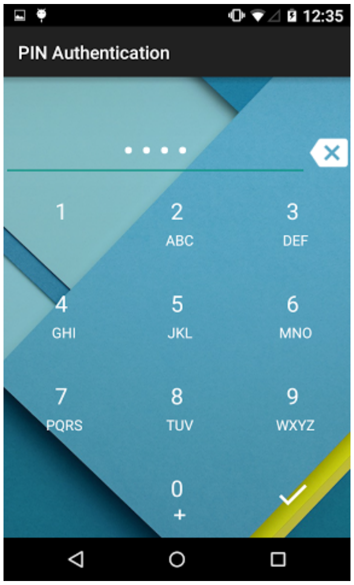
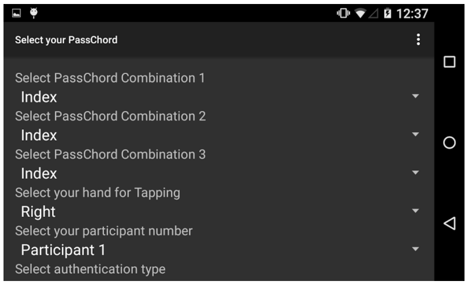


Figure 3.3: Enter PIN

find the total time for selecting the PIN. AUTHENTICATION STARTED is stored in the results column after the app is fired on the screen automatically to enable the user to enter the PIN. Finally, AUTHENTICATION SUCESSFUL is added to the results after the user enters the correct PIN. The difference between the timestamps of AUTHENTICATION STARTED and AUTHENTICATION ENDED enables us to find the total time to enter the correct PIN and unlock the phone. If the user enters an incorrect PIN the value stored in the result column of the database is INCORRECT PIN ENTERED. All of the data is stored in a SQLite database in the external memory of the Android phone. This data will be used for further analysis by using different statistic models.

3.2 PassChord Authentication

This implementation of the PassChord authentication system app is the same as described by Azenkot et al. [3]. This app was used for comparison with our own implementation of tap Authentication with Rhythmic Piano taps. The user first needs to select any three combinations out of the 15 possible combinations seen in



Select your PassChord

Select PassChord Combination 1
Index

Select PassChord Combination 2
Index

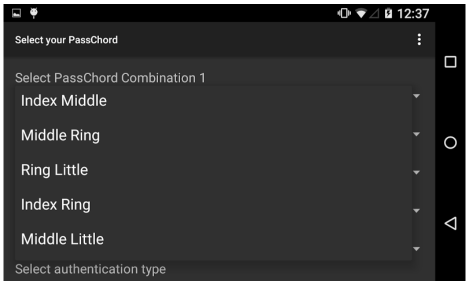
Select PassChord Combination 3
Index

Select your hand for Tapping
Right

Select your participant number
Participant 1

Select authentication type

Figure 3.4: Select tap combinations and hand to be used



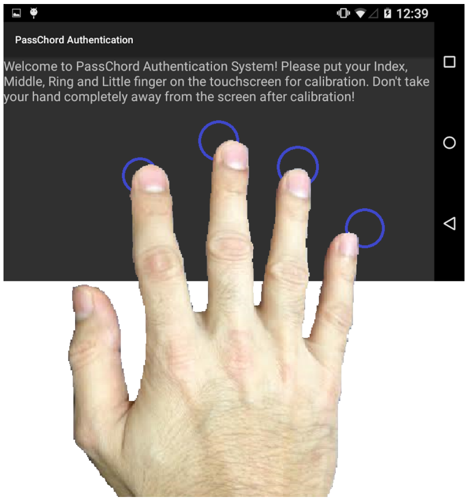
Select your PassChord

Select PassChord Combination 1

- Index Middle
- Middle Ring
- Ring Little
- Index Ring
- Middle Little

Select authentication type

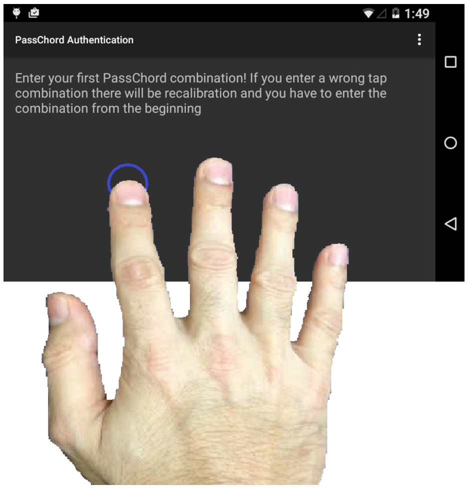
Figure 3.5: Select 1 combination from 15 combinations



PassChord Authentication

Welcome to PassChord Authentication System! Please put your Index, Middle, Ring and Little finger on the touchscreen for calibration. Don't take your hand completely away from the screen after calibration!

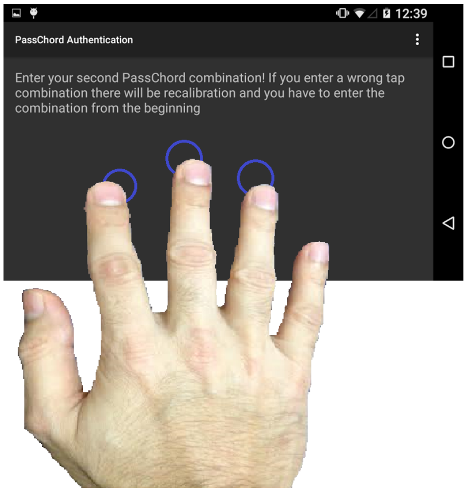
Figure 3.6: Calibrate to authenticate



PassChord Authentication

Enter your first PassChord combination! If you enter a wrong tap combination there will be recalibration and you have to enter the combination from the beginning

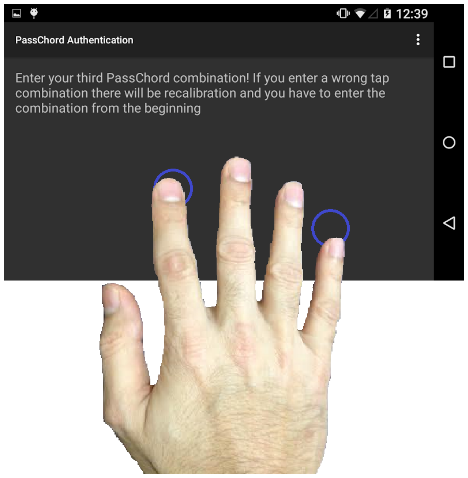
Figure 3.7: Enter combination 1 to move to the next step



PassChord Authentication

Enter your second PassChord combination! If you enter a wrong tap combination there will be recalibration and you have to enter the combination from the beginning

Figure 3.8: Enter combination 2



PassChord Authentication

Enter your third PassChord combination! If you enter a wrong tap combination there will be recalibration and you have to enter the combination from the beginning

Figure 3.9: Enter combination 3

figure 3.10. The three combinations can be same or different. The user also needs to select the hand which will be used for taping the combinations. The user can select SAVE SELECTION to confirm the PassChord combinations. Now, PassChords authentication system is a four step process. In the figures 3.6, 3.7, 3.8 and 3.9 it must be noted that the blue rings are just for the readers to understand the finger being tapped. There are no blue rings displayed on the screen while taping. In the first step the user presses the index, middle, ring and the little finger against the screen of the smartphone for calibration. The calibration co-ordinates of fingers are used further while entering all the three combinations to detect which combination of fingers were used during each of the taps. We have used the Gaussian probabilistic model to detect which fingers were tapped in each combination, which we assume was also used by Azenkot at el. [3] [5]. The app will ask the user to enter the first combination only when all the four fingers are detected and calibrated. As can be seen from the figures above, after the calibration stage, the user enters the first combination of finger taps. The user taps the index finger as the first combination. After the user has entered the first combination the next screen will show up asking the user to enter the second combination of finger taps. From the figure 3.8 we observe that the user presses the index, middle and the ring finger as the second combination. The third and last combination of tap entered by the user is index and little finger. This sequence and the combination of different fingers must only be known to the user for security of the authentication system. This is our implementation of the PassChord authentication app in Android smartphone.

3.2.1 Finger detection logic

As we know, all touchscreen phones have a particular resolution. The values of resolutions vary with the touchscreen size and quality of the display. The Android phone which has been used for developing and testing this app is the LG Nexus 4 with Android 5.1.1 Lollipop or Android API 22. This phone has a resolution

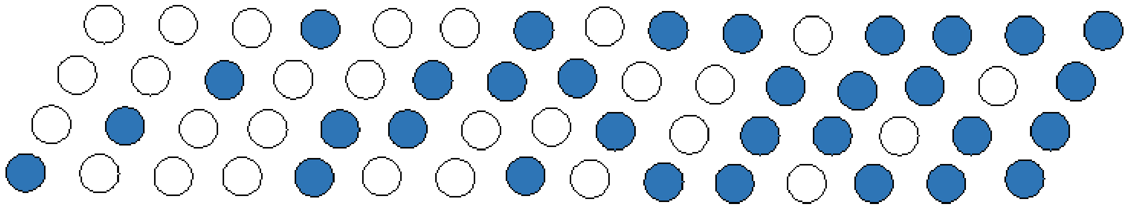


Figure 3.10: PassChord tap combinations with circles representing a tap. The bottom left circle is the index finger and the top right circle is the little finger.

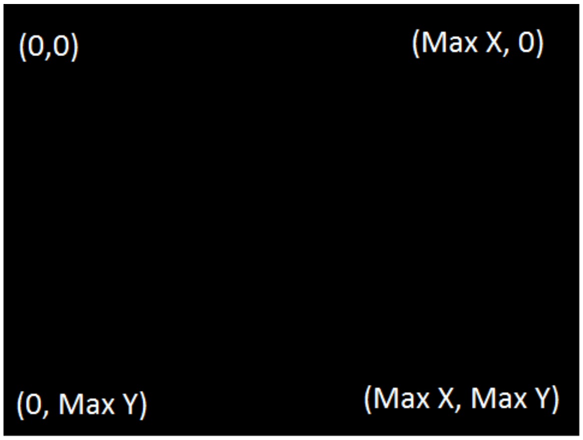


Figure 3.11: Co-ordinate system of touchscreen of Android smartphone

of 1280 x 768 pixels. This application is developed and tested only for landscape orientation of the touchscreen for the purpose of convenience for the user as four fingers would be difficult to accommodate in portrait mode. The touchscreens of all Android smartphones have been designated the Cartesian co-ordinate system with x and y axes. From the value of the resolution, the maximum value of the x co-ordinate is 1280 and the maximum value of the y co-ordinate is 760. There are around one million pixels on the touch screen of the LG Nexus 4 smartphone. The figure 3.11 above, illustrates the co-ordinate system of an Android smartphone in the landscape orientation.

This system allows the user to select which hand the user wants to use to enter the tap combinations. When the user opens the app for the first time after installation, the app asks the user to select all three tap combinations along with the hand to be used for entering the taps in the future. The user can change the finger taps and the hand used to enter the taps by pressing the CHANGE TAP COMBINATION button after authenticating correctly. Considering the case when user selects the right hand, the index finger will have the smallest pixel value of the x co-ordinate and the little finger will have the largest value of the x co-ordinate. The middle and the ring finger will have intermediate values for the x co-ordinate such that the values for the fingers would be of the order index<middle<ring<little. Considering the case when a user selects the left hand, the index finger will have the largest pixel value of the x co-ordinate and the little finger will have the smallest value of the x co-ordinate. The middle and the ring finger will have intermediate values for x co-ordinate such that the values for the fingers would be of the order index>middle>ring>little. The onTouch function in Android has a method defined which helps determine the co-ordinates for each finger tapped on the screen from the touch screen hardware. The x and y co-ordinates of each finger are stored in different arrays during the calibration stage and passed on to the next activity for detecting the fingers pressed against the screen.

3.2.2 Maximum likelihood detection

This finger tap detection technique for smartphones was implemented by researchers at the University of Washington in their project which was used for implementing Braille keyboard on a touchscreen [5]. We assume that the same technique was used by Azenkot at el. [3] to develop the PassChord app in Android. To determine which finger touched the screen, we compare the input touch co-ordinates to the reference calibrated co-ordinates set by the user in the first step. It is assumed that the user has not reposition his or her hand, so the touch points for each finger are likely to be close to their reference calibration points. This detection assumes the distribution of touch points around their respective reference points is Gaussian with center at their respective reference points. Let d be an observed data point (tapped input) and θ be the calibration point saved in the first step. The hypothesis that θ is the TRUE

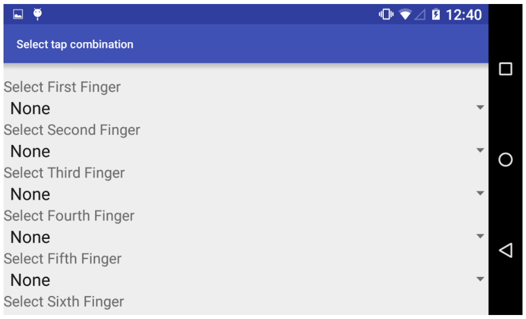
VALUE of the finger tapped is hθ. Suppose d is one point that corresponds to one reference point θ. We have four hypotheses for θ: h1, h2, h3, and h4 where h(i) is the hypothesis that d was input with finger i. Here h1 is the hypothesis that the tapped finger is index finger, h2 is the hypothesis that the tapped finger is middle finger, h3 is the hypothesis that the tapped finger is ring finger and h4 is the hypothesis that the tapped finger is the little finger. P (d | hθ) is the probability of observing d given hθ and the ML detection is the value of θ that maximizes P (d | hθ). We express P(d | hθ) as a Gaussian distribution centered around reference point θ with variance σ^2 [5]. The equation below is used to calculate the probability for each input finger with respect to the calibration points of the first step.

$$P(d|h\theta) = \frac{1}{\sqrt{2\pi}\sigma} exp\left[- \frac{(d-\theta)^2}{2\sigma^2} \right]$$

In this implementation the x co-ordinate of each touch point is used as the reference or the calibration point. The tapped fingers do not differ by much in their y co-ordinates so it is not possible to identify the tapped finger based on the y co-ordinates on the touchscreen. The probability calculation function calculates the probability for the unknown touchpoint with respect to the calibration co-ordinates of each finger. The one with the highest probability is assumed to be the tapped finger.

3.3 Pianotap Authentication (Tap authentication with rhythmic piano taps)

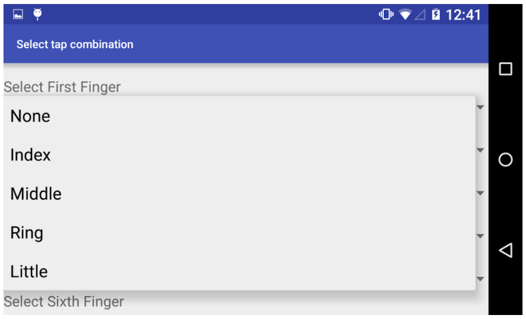
People commonly tap the edges of tables, covers of laptop or any tough sheet of metal or wood [4]. These rhythmic taps can be used as a tap combination to unlock a smartphone. The rhythmic combinations are easy to remember and tap on the screen of a smartphone. Previous attempts at tap authentication require some sort of a rhythm, but in our version of tap authentication the user can tap any combination



Select tap combination

Select First Finger
None
Select Second Finger
None
Select Third Finger
None
Select Fourth Finger
None
Select Fifth Finger
None
Select Sixth Finger

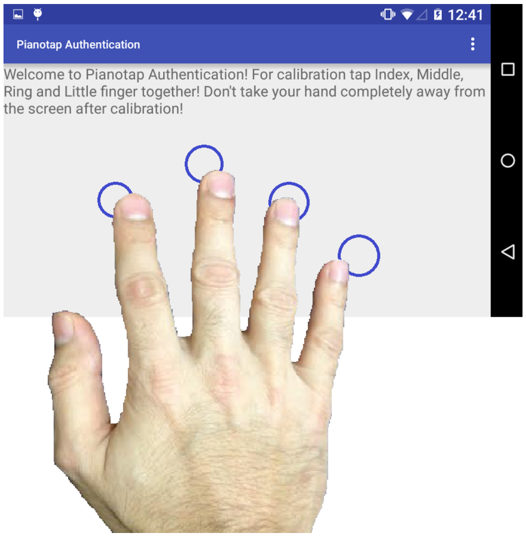
Figure 3.12: Select taps



Select tap combination

Select First Finger
Select Sixth Finger

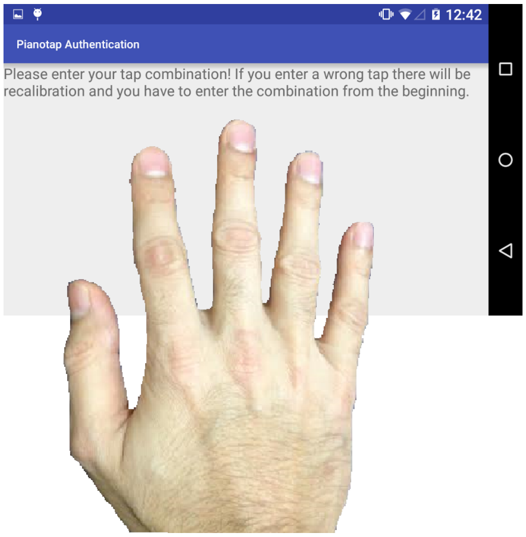
Figure 3.13: Select first tap



Pianotap Authentication

Welcome to Pianotap Authentication! For calibration tap Index, Middle, Ring and Little finger together! Don't take your hand completely away from the screen after calibration!

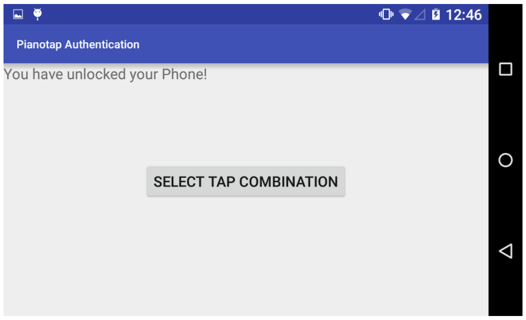
Figure 3.14: Calibrate to authenticate



Pianotap Authentication

Please enter your tap combination! If you enter a wrong tap there will be recalibration and you have to enter the combination from the beginning.

Figure 3.15: Enter tap combination



Pianotap Authentication

You have unlocked your Phone!

Figure 3.16: Authentication successful

of fingers after the calibration phase to get authenticated. We have allowed the user to choose up to 10 finger taps. Entering more than 10 finger taps every time would be a time consuming affair for the user. This would defeat the purpose of having an authentication system in the first place. A user typically unlocks the smartphone numerous times a day. As the number of finger taps increase, the time taken to authenticate also increases linearly. It would be cumbersome to remember more than ten taps and enter them correctly every time while authenticating. A password containing eight to ten taps would be quite secure because there are a plethora of possible combinations using four fingers to enter ten taps on the touch screen. To be precise there are about 1,048,576 possible tap combinations in the Pianotaps authentication system.

When the user installs and starts the app for the first time the app tells the user to select each of the ten taps. There is an option where the user can select which hand he/she would like to use while tapping the combinations each time the phone needs to be unlocked for access. The user can select one finger out of the index, middle, ring and little finger for each tap by using the spinner menu. If the user wants to authenticate using five taps only, the other finger taps, from six to ten, are by default selected to none. For the study we have put an additional spinner where the participant can select the appropriate participant number assigned during the study. The user can set the taps and the hand to be used for tapping by pressing the SAVE SELECTION button. Clicking the SAVE SELECTION button saves the taps and takes the user to the finger calibration screen. It must be noted that the blue rings in the figure are just for the readers to understand the finger being tapped. There are no blue rings displayed on the screen while taping. This authentication system has the calibration screen which is similar to the one in the earlier PassChords authentication system. The user presses all the four fingers i.e. index, middle, ring and the little on the touchscreen to calibrate the position of the fingers (see fig. 4.14). When all four fingers are pressed against the touch screen the app starts another activity and it is displayed on the screen. In the background, the app stores the location of each finger position in the form of x and y co-ordinates of screen resolution. These co-ordinates will be further used to detect which finger is tapped each time the user enters the tap combination. The user now needs to enter the combination which was set earlier. When the user taps all of the correct fingers, one by one, the phone is unlocked. If the user presses any wrong finger then the app tells the user to recalibrate and enter the taps from the beginning. The phone would only be unlocked when the user enters the exact combination of finger taps. The user would know when an incorrect combination is entered because a toast message will be posted and there will be a short vibration of 300 milliseconds. The user can change the finger taps in the future by clicking the CHANGE TAP COMBINATION button. This button is displayed on the screen after the user successfully authenticates.

We are currently capturing all data from the sensors such as accelerometer, light, linear acceleration, gravity, gyroscope, magnetic field, pressure, proximity, humidity and temperature in the Android SQLITE database. These values are stored in the database whenever the user enters each tap of the full combination and while creating the tap combination. If the tap entered is incorrect then the database has a result column which stores the tap which was entered incorrectly. When the user enters the correct combination of finger taps the database stores an entry with all the above sensor values and the string AUTHENTICATION SUCCESSFUL in the results column. This database will be used for statistical analysis and comparison of all the authentication mechanisms.

Chapter 4
EXPERIMENTAL METHODOLOGY

4.1 Participants

We had recruited 12 participants for the informal lab study. The participants were both females and males over 18 years of age. The participants included 10 males and 2 females and the average age of the participant was 25.91 years. Our participants were limited to college students as the study was informal. All of the participating college students were from Rutgers University. The data collection tools included the three different Android apps which were installed on the Android phone given to the participant during the study and a video camera used to record the participants entering the PIN or tap combination. The data collected using the video recording was used to investigate shoulder surfing attacks against all the authentication systems. The recordings were kept securely to maintain the confidentiality and privacy of the participants during and after the research. The data collection was performed using the Android smartphone provided during the session. The sessions for the study were conducted in a room in the Core Building in the Busch Campus at Rutgers University, the Alexander Library on College Ave and in a residential house on Louis Street, New Brunswick.

4.2 Procedure

There were two sessions in total. In the first session the participants had to use the app to create a PIN and tap combinations and authenticate. In the second session the participants had to attempt to crack the PIN and tap combinations of the other participants by carefully viewing the video captured during the first session. The second session was for the PIN and tap combination recall session. The participants were asked to recall their own PIN and tap combination created during the first session. A detailed information of both sessions can be found below.

4.2.1 Session 1: PIN and tap combination creation and authenticating in the app

PIN Authentication

In this session the participant was instructed on how to use the PIN app. It emulated the mechanism of the PIN authentication system used in Android and iOS smartphones. Though the app could save more than four digit combination PINs, the participants were restricted to create four digit combinations only because most modern phones use four digit PINs. Six digit PINs were introduced just recently in the latest iterations of iPhones. These phones also have the biometric fingerprint detection sensor and the system is popularly called Touch ID authentication. This is much faster than entering six digit PINs and is generally preferred and perceived more secure by the users. The users were also restricted from creating simple PINs like 1234, 5555 etc.

Once the user had selected the PIN and confirmed it the PIN creation step was complete. On confirming the PIN, the app would automatically close and run in the background. The camera was adjusted to record the participant entering the PIN. This recording was to be used in the second session for testing it against shoulder surfing attacks. Once the camera was adjusted the participant locked the phone and unlocked it under the supervision of the camera. The app would pop up and ask for the PIN to unlock the smartphone. This marked the end of the authentication stage of the normal PIN unlock. This is called a normal PIN unlock because it was done by the user by looking at the screen.

The next part was authenticating without looking at the phone i.e. by keeping it under the table. It is very difficult to enter a PIN without looking at the smartphone. This emulates a person using PIN authentication inconspicuously. There are limitations to this study with respect to comparing inconspicuous authentication to authentication used by a blind or visually impaired person. We have discussed this in the limitations section below. The blind user interacts with the smartphone using screen readers mentioned earlier. Authenticating without any visual or aural feedback was nearly impossible for the participants. The user would never know the buttons pressed on screen. To make it possible for participants to authenticate by typing the PIN without looking at the phone, we enabled the Talkback and provided them a headset. This option can be found in Android under Settings ⇒ Accessibility ⇒ Talkback. After activating the Talkback, on single tap the screen reader reads aloud the app icon or option being tapped. The double tap in Talkback is like a single tap for normal users which is used to select some option or open any app. The voice feedback enabled the participants to have knowledge of their interaction with the phone.

As none of the users were blind, they were not aware of this accessibility option available in the phone. It took some time and practice to get used to the new way of interacting with the smartphone. After getting a sufficient hold over Talkback the users were asked to lock their phone again and hold it under the table. The participants were now asked to unlock the phone with the voice enabled guidance provided by the screen reader. Though this took a much more time all the participants were able to unlock the phone. All the participants empathized the pain the blind users have to go through just unlocking the smartphone. They realized how difficult and cumbersome it is for a blind person to use a phone regularly. This completed the use of the first authentication technique of the first session.

PassChord Authentication

A replica of the PassChords app has been developed for use in the study. The authentication system was explained to the participants. In the creation stage, the participants selected three combinations for each tap entry. The participants were instructed to use single finger taps only once. They were encouraged to select two, three or four finger tap combinations. The participants had to enter the tap combinations to unlock the phone while the camera took a video recording of the same. This video was captured to test it against shoulder surfing attacks. This enabled us to know the time taken for PassChord selection and authentication with visual feedback from the screen.

Next, the participants had to unlock the phone without any visual feedback, i.e. under the table. The participants used the same PassChord created earlier to authenticate, only this time they could not see the screen. No video recording was taken for this part. If any combination was entered incorrectly, the phone would vibrate and the participant had to restart from the calibration step. Activating the screen readers was not necessary as phone vibration was enough to suggest the participants to re-enter the tap combinations. The participants found this technique less time consuming and user friendly.

Pianotap Authentication

The final authentication app for the study was the authentication using rhythmic piano taps. Each participant had to create a rhythmic tap combination using at least five fingers. The participants were restricted from creating overly simple rhythmic taps like using just one finger for all the taps. The participants selected different rhythmic tap combinations due to the plethora of permutations and combinations possible. After creating the combination, the participant was asked to unlock the phone in the presence of the camera recording the authentication process. This recording was used to test the technique against shoulder surfing attack in the next session.

Now, the users had to enter the same tap combination under the table. The participants found it very easy to enter this combination without any visual feedback, because, like PassChords, the phone vibrated upon entering an incorrect combination and it needed to be recalibrated. Activating the screen reading option was not necessary as no keys had to be pressed specifically for authentication. It hardly took participants any time to enter the rhythmic tap combinations. This ended the first of the two session study. The participants were instructed to remember the PIN and tap combinations entered during the session as they would have to recall the same in the next session for testing the memorability of PIN and tap combinations.

4.2.2 Session 2: PIN/Tap combination recall and test for shoulder surfing

In this session the participant was shown the video recording of some other participant entering the PIN and tap combinations. This emulates the shoulder surfing attack used to crack different types of combinations and passwords. The participants were given at the most three attempts to crack the PIN and tap combinations looking carefully at the video. The response of all the participants was recorded and was used to analyze the security of the authentication system.

In the recall part the participants were asked to recall and write their own PIN and tap combinations. These are compared with the actual combinations created in the first session to determine whether the participant remembered all of the combinations correctly and in the correct order. The recall part was used to compare the memorability of PINs with the memorability of tap combinations.

4.2.3 Limitations of this study

Our study has been designed to capture an individual using all the three authentication systems and an imposter trying to crack the combinations using shoulder surfing.

Although, most of the study relates to an actual real world scenario, there are some limitations.

First, all of the participants in this study were non-impaired individuals with regular vision. This study was conducted taking into consideration that the system can also be used by a blind person. Thus, we cannot accurately reproduce the way a blind person would interact with the authentication system. There are studies indicating blind people are more sensitive to touch than a non-impaired person. They can perceive tactile information much faster than anyone else. They are more familiar with screen readers and interacting with the phone without any visual feedback. In our study most users were not aware of screen readers and had never used them. They tried to adapt to the screen readers when using PIN authentication, but it is difficult to grasp the system in only 15 minutes. It requires practice and usage over a longer duration of time. This is one of the important factors to consider during the analysis.

Second, in the shoulder surfing part using a camera, the camera cannot be considered as a replacement for the eye. We used a high definition camera to capture the video of a participant entering the PIN and tap combinations, but in the case of PIN authentication it was observed that numbers on the keypad could not be seen clearly. The numbers could be seen clearly with eyes at a distance much farther than the camera. We have compensated for this by giving the shoulder surfers 3 attempts to crack the correct PIN and taps. In reality, however, the shoulder surfer may or may not get more than one chance to shoulder surf combinations depending on various factors. The solution to this problem would be a scenario where one or more shoulder surfer observes the participant entering the PIN/Tap combinations. The number of volunteers were less for our informal study and can be considered in a formal study.

4.3 Analysis

Data analysis was performed using the data collected by the Android smartphone and the video recorded by the camera. In the recall session a failed attempt is where a participant could not recall the PIN or tap combination correctly. This metric reveals how easy it is to memorize the PIN and the tap combination. In the shoulder surfing part the participant was shown the video of the other participants entering the PIN and taps and the responses were recorded for all of the attempts. We analyzed the easy and difficult to crack combinations using the recorded data. We used the Friedman test to statistically compare the significance of authentication time for authentication with and without visual feedback. The post-hoc was done using the Wilcoxon rank test.

Chapter 5
RESULTS

All of the participants were graduate students who had already been using smartphones and already had used some form of authentication system to secure their phones from unauthorized usage. The popular authentication systems used by the participants included PINs, Android pattern lock and Touch ID. All of the participants had fully functional vision and used smartphones on a daily basis for calling, text messaging, email, social networking and using various other apps. Thus the participants had an enormous amount of confidential data on their phones. All of the participants completed both sessions. The participants used the LG Nexus 4 Android smartphone provided during each of the sessions. All of the apps to be tested were installed on this smartphone.

In the figure 5.1 we can observe the average authentication time using the PIN, PassChord and Pianotap authentication with visual feedback and inconspicuously by keeping the phone under the table.

5.1 Session 1: PIN/Tap combination creation and authenticating in the app

5.1.1 PIN, PassChord and Pianotap unlocking conspicuously (Screen visible to the participant)

The mean time required to unlock the smartphone using PIN, PassChord and Pianotap were 5.33, 13.33 and 8.83 seconds respectively. In this part the effect of method

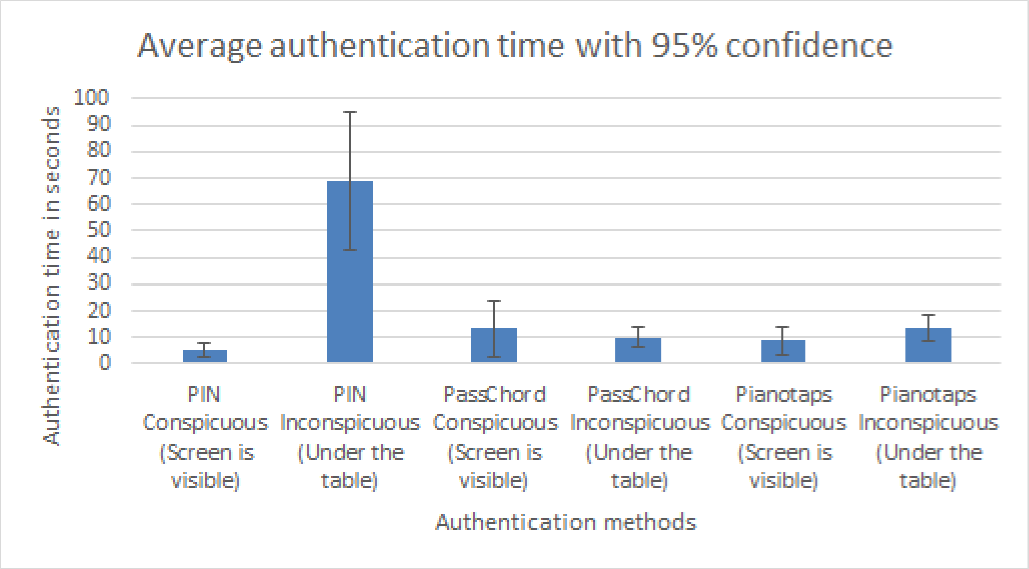


Figure 5.1: Average authentication time with 95% confidence

on authentication time was found not be significant ($\chi^2(2) = 4.31$, p = 0.1158). Thus, statistically there was not a significant difference between the authentication time of PIN, PassChord and Pianotap authentication. This was found using the Friedmans test. The minimum time required for PIN, Passchord and Pianotap was 3, 2 and 4 seconds respectively. The maximum time required for PIN, PassChord and Pianotap was 18, 59 and 31 seconds respectively.

The PIN authentication system took the least time to authenticate followed by Pianotap and PassChord authentication. PassChord and Pianotap needed more number of taps to be entered hence, it was much more secure to shoulder surfing attacks which will be observed in the next section. Here we have considered the total authentication times for all of the authentication mechanisms. The total authentication time includes the time required for recalibration and entering the PassChord and Pianotap from the beginning in case of an input error. Input error is generally caused because the participant moves the hand away from the calibration co-ordinates on the touchscreen. The participants were very comfortable using the PIN authentication because of prior experience. There would be a great reduction in the authentication time after the participants get more comfortable with Pianotap and PassChord. Prior experience is a major factor which distinguishes the authentication time for all the systems and is thus a limitation to this study. The difference between the mean authentication time for PIN and Pianotap is hardly 3.5 seconds which is comparatively less than the number of taps and level of security it has to offer. Pianotap beats PassChord to be the faster tap authentication mechanism.

5.1.2 PIN, PassChord and Pianotap authentication inconspicuously (Under the table)

The mean time required to unlock the smartphone using PIN, PassChord and Pianotap is 68.75, 10.08 and 13.41 seconds respectively. The effect of the method on the authentication time was found to be significant ($\chi^2(2)$ = 20.47, p = 0.000035). Pairwise significance was found between PIN and PassChord authentication (Z = 4.13, p = 0.000036, r = 222) and also between PIN and Pianotap authentication systems (Z = 3.9, p = 0.000095, r = 218). There was no evidence of significant effects between PassChord and Pianotap authentication systems (Z = -1.39, p = 0.1642, r = 125.5). The pairwise significance was found using the Wilcoxon rank test. Hence, PIN was found to be more time consuming compared to PassChord and Pianotap authentication systems. The minimum time required to authenticate using PIN, PassChord and Pianotap authentication systems was 23, 4 and 4 seconds respectively. The maximum time required to authenticate using PIN, PassChord and Pianotap authentication systems was 142, 21 and 29 seconds respectively.

The PIN authentication system was completely outmatched by tap authentication systems in this section. The lowest time required by PIN was 23 seconds which is around 13 and 10 seconds more than the mean of PassChord and Pianotap authentication respectively. Thus, we can easily conclude the PIN system is very difficult to use inconspicuously. The only limitation here is that the systems were not used by actual blind people. Considering that blind people can perceive tactile and touch better than non-visually impaired people the time taken by a blind user would be comparatively less. It would be interesting to see the margin of difference in authentication time if it actual blind person uses all three systems.

5.1.3 PIN/Tap authentication selection by the participants

The average number of taps required for PIN, PassChords and Pianotaps was 4, 6 and 7 respectively. We observe that average number of taps entered per second were greater in Pianotap than PassChord even though PassChord uses multiple finger tap combinations. Thus it can be inferred that the input error rate for PassChord is more compared to Pianotaps.

The most popular tap combinations involved the index and the middle finger. The little finger was the least preferred finger while selecting the tap combinations. There were some irregularities observed in selection of each combination for PassChord. The participants completely ignored some combinations like tapping together index middle little and middle little fingers. The participants avoided these combinations because they would be difficult to enter due to the limitations of movement of fingers of the human hand. This considerably decreases the tap combination selection space because there are just 15 possible combinations and some of them employing single fingers are easy to crack using shoulder surfing. In case of Pianotap, the user only needs to tap one finger at a time. Tapping one finger at a time is much easier and does not obstruct the use of any combination. There is no physical human limitation to tapping single finger combinations on the touchscreen of the phone.

5.2 Session 2: PIN/Tap combination shoulder surfing, recall and feedback

5.2.1 Shoulder surfing

In this session, as mentioned earlier, the participants had to watch a video recording of some other participant and try to crack the PIN and tap combinations entered by the user. All participants were given only three chances to view the video of each authentication technique and crack the combination.

The PIN was easily shoulder surfed by all the participants at the most in three attempts. The same cannot be said about both the authentication systems which employ finger taps. It was nearly impossible to identify the tap combinations in the first shoulder surfing attempt. Almost all of the participants required more than one attempt to get the shoulder surfed tap combination correct. The participants found shoulder surfing PIN authentication to be very easy and were confident while cracking the PIN combinations of other participants. The participants were muddled and confused after seeing the video of participants entering the tap combinations of PassChord and Pianotap. Even though they had used both the tap authentication systems in the first session, most of them could not crack the exact tap combination tapped by some other participants in the video. This is of great importance because there is no visual feedback unlike PIN where the number buttons can be seen as they are highlighted on being tapped on the touchscreen.

Most of the PINs were cracked by the participants in the first attempt by simply looking at the video. Just three participants took two attempts to crack the PIN and only one participant took three attempts to crack the PIN. The average number of attempts taken by participants to successfully shoulder surf the PIN was only 1.33 which is close to just one attempt. This is alarming considering that this authentication system is still used by millions of smartphone users throughout the world. All of the users who rely on this system in public places like trains, buses, offices and colleges etc. are extremely vulnerable to shoulder surfing. Thus, PIN is the weakest method of authentication compared to the other methods studied in this thesis.

Tap Combination	**No. of times selected**	**No. of times cracked**	**Percentage of times cracked**
Index	4	4	100%
Middle	1	0	0%
Ring	1	1	100%
Little	0	0	0%
Index Middle	5	3	60%
Index Ring	1	0	0%
Index Little	2	2	100%
Middle Ring	3	3	100%
Middle Little	0	0	0%
Ring Little	1	1	0%
Index Middle Ring	6	4	66%
Middle Ring Little	3	0	0%
Index Ring Little	0	0	0%
Index Middle Little	0	0	0%
Index Middle Ring Little	5	2	40%

Table 5.1: PassChord cracking statistics

It can be observed from the table above that among the tap combinations cracked by shoulder surfing the PassChord, one and two finger combination taps were easily cracked by the participant watching the video. It was observed that PassChord combinations which did not involve the little finger were also cracked by many participants. This was partially due to the reason that in the video recordings it could be easily observed that the little finger did not touch the screen at all. It was also observed that many participants particularly pulled away the little finger while entering the combinations, which did not use the little finger. Thus, the length of the little finger, and pulling away the little finger while entering the combinations are a serious threat to this system because it further reduces the secure tap combinations increasing the chances of cracking. Surprisingly, the four finger tap combination, which is easy to observe, was not cracked easily because the participant could not guess whether all the fingers were touching the screen simultaneously in a fraction of a second. This combination was easily confused as the calibration stage in PassChord authentication by some of the participants.

The Pianotap which were cracked by the participants were usually the ones which repeated the same finger twice or more consecutively or were entered at a very slow pace by the participant. It was observed that combinations which involved less usage of repeating the same finger taps were more secure. This is due to the difficulty in observing one out of four fingers which will be tapped and simultaneously remembering it to form a combination in the back of the ones mind. Thus, skill and speed of taps plays a major role in the security of tap authentication systems.

5.2.2 PIN/Tap combination recall

In this session the participants were asked to recall the PIN and tap combinations selected as secret combinations in the last session. The recall session was held after 3-4 days after the first session. The following paragraphs and tables explain how successful the participants were in recalling their combinations. We have also compared the three authentication systems based on combination recall success rates.

Participant	PIN recalled	PIN actual
1	2592	2592
2	5508	5508
3	0588	0588
4	4268	4268
5	2412	2412
6	1475	1473
7	0891	0891
8	0859	0859
9	8426	8426
10	4419	4419
11	2614	2614
12	0828	0828

Table 5.2: PIN recall table

From the table we can infer that it is easy to remember PINs. This is partially due to the fact the participants may have selected PIN which they were using currently on their phones. All participants had used the PIN style of authentication for some purpose in their lives even if they are not using it currently. Besides, it is very easy to remember a four digit number because it is generally associated with something like last the four digits of their social security number, a special date like a birthday, cellphone number, Bank ATM PIN etc. It can be noted that only one participant got the PIN wrong on recall. The PIN, which the participant recalled incorrectly, was only by the last digit. Thus, we can say PINs are the most easy to remember secret keys for an authentication system.

Participant	**PassChord recalled**	**Passchord actual**
1	MRL;IMRL;IMR	MRL;IMRL;IMR
2	IMRL;IR;IMR	IMRL;IR;IMR
3	I;IM;IMR	I;IM;IMR
4	M;IMR;IM	M;IMR;IM
5	IMRL;IR;MRL	IMRL;IM;MRL
6	IL;MR;I	IL;I;MR
7	IM;MR;RL	I;IM;RL
8	IM;IL;IMR	IM;IL;IMR
9	MR;R;IMR	MR;R;IMR
10	I;MR;IMR	I;MR;IMR
11	M;IM;IR	IM;MR;I
12	IM;MR;RL	IMR;MRL;IMR

Table 5.3: PassChord recall table (I: Index finger, M: Middle finger, R: Ring finger and L: Little finger)

From the table above we can clearly observe that there were exactly 5 participants who got their recalled PassChord combinations wrong. In the PassChord technique the participants need to select three tap combinations out of fifteen total combinations. Many participants were confused while recalling the order of the three combinations. After looking at the table above we realize that participant 6 faced this problem.

Participant	**Pianotap recalled**	**Pianotap actual**
1	IMRRMIMMM	IMRRMIMMM

2	IMRLIRM	IMRLIRM
3	LIMIIMM	LIMIIMM
4	MILRR	MILRR
5	LRMIII	LRMIII
6	IIMMLL	IIMMLL
7	IMRLIMR	IMRLIMR
8	IMILIMR	IMILIMR
9	MIMLMIRI	MIMLMIRI
10	IMRLRMI	IMRLRMI
11	MLIRML	MLIRML
12	IMRLRMIRMI	IMRLRMIRMI

Table 5.4: Pianotap recall table (I: Index finger, M: Middle finger, R: Ring finger and L: Little finger)

The above table demonstrates that Pianotap were most accurately recalled. All participants recalled their tap combinations correctly without any errors for even a single finger tap. These taps are more easy to memorize because of the rhythm associated with the tapping which is lacked by the PassChord tap authentication technique. Many of the participants used unique techniques to remember the tap combinations of Pianotap. The most common technique described by many participants was they associated each finger with a digit and remembered the number formed by all of the taps. Some participants recalled it correctly because it was the rhythm of their recent favorite song. We can conclude that PIN and Pianotap are much easier to remember as compared to PassChord combinations.

The limitation of this session is that the participants did not use the authentication systems in the 3-4 days between the two sessions of the study. It is highly probable that they might have used the PIN authentication on their phones and have the same PIN saved as their secret key. This might also be one of the reasons of highly correct memorability of PINs. It would be interesting to know the difficulties or ease in remembering the combinations after conducting a formal study where the app would be installed on their phone and they would be using this authentication system every day. We think and believe that it would be much easier to remember the combinations when they are entered multiple times in a day.

Chapter 6
DISCUSSION

The study indicates that tap based authentication techniques like PassChord and Pianotap are a good match for PIN authentication in terms of ease of use and speed of authentication. The PassChord and Pianotap are more resistant to shoulder surfing as compared to PIN which is very weak against such attacks. In case of security, both the tap authentication systems outclass the PIN authentication system. This can be elucidated by the number of successful attempts in shoulder surfing on the PIN system.

Tap authentication systems deemed to be difficult to use as compared to PIN unlock. The PIN authentication seems to be more convenient because we have been using the system for a very long time and are habituated with it. The study indicates that the use of tap authentication is easy and much more secure. Among the two tap authentication systems studied, Pianotap holds the edge over PassChord, because the former provides same level of security as the latter. Pianotap emerged victorious for the case of possible number of tap combinations and user selectivity of tap combinations. It is easier to remember Pianotap as compared to PassChord tap combinations. In case of Pianotap, the user can also associate a number with respect to the Pianotap combination. This number can be formed by associating numbers 1, 2, 3 and 4 with index, middle, ring and little finger respectively. This would be similar to remembering the cellphone contact number of a person. Thus, it is easy to memorize Pianotap than PassChord.

The threat to Pianotap and any SOMETHING YOU KNOW system is video recording attacks. Though it is a little difficult to understand the fingers tapped while entering the finger taps in the first few attempts, it is not impossible to shoulder surf the combination if the attacker can replay the video numerous times at will. This attack is can be mitigated by authenticating inconspicuously by putting the phone under the table, occluding by clothes (e.g. jacket), leaning device against the body, using device upside down, moving to an isolated location, postponing etc. [13]

The security of the Pianotap can be improved by selecting a tap combination having as many taps possible. We would like to lay down the guidelines for selecting a secure Pianotap combination:

1) Select at least five finger taps as a tap combination.

2) Use all the four fingers i.e. index, middle, ring and little for the tap combination.

3) Try to avoid choosing tap combinations which involves the use of consecutive double or more tapping of the same finger.

Pianotap would have a great impact in studying and developing ‘Something you know’ authentication systems to be used inconspicuously or by disabled people. It will also contribute in developing authentication systems which can be used in a variety of systems and scenarios universally. By comparing PIN authentication with Pianotap authentication we observe that tap authentication has a very good potential to replace the former.

Chapter 7
CONCLUSION

Smartphones save the private data of the users and are exposed to several security risks discussed in the above chapters. We have presented a comparative study of the most commonly used PIN authentication, PassChord authentication and our own implementation of the Pianotap authentication and have concluded that the tap based authentication including ours, provides more security against shoulder surfing attacks. Our method allows the user to select a tap combination of their favorite rhythm and unlike PassChord, it is not bounded by just 15 combinations. Many of the 15 combinations are not secure and some are difficult to enter because of the limitations of the human hands. Hence, there are a limited set of combinations that can be used in reality and thus renders the system susceptible to shoulder surfing attacks. PIN authentication is the fastest in terms of authentication, but the least secure against attacks. Due to the amount of private and confidential data stored on the phone it is better to be safe and secure rather than regret the consequences.

The security of tap based authentication systems is also based on the individual tap combination selection and the tapping style of the user. Therefore, it is very important that the users do not choose easy tap combinations nor enter tap combinations too slowly. If this is not taken care of then it defeats the purpose of any authentication system including Pianotap. Our system provides secure inconspicuous authentication system and may be able to provide secure authentication system to the blind and visual impaired community which is one of the most important goals and achievements of the Pianotap authentication.

References

[1] Janne Lindqvist and Alexander De Luca. Is secure and usable smartphone authentication asking too much? In *Indistinguishable from Magic*, IEEE Computer Society, May 2015.

[2] Michael Sherman, Gradeigh Clark, Yulong Yang, Shridatt Sugrim, Arttu Modig, Janne Lindqvist, Antti Oulasvirta and Teemu Roos. User-generated free-form gestures for authentication: Security and memorability. In *Proceedings of the 12th annual international conference on Mobile systems, applications, and services*, MobiSys '14, pages 176 - 189, ACM, New York, USA, June 2014.

[3] Shiri Azenkot, Kyle Rector, Richard E. Ladner, and Jacob O. Wobbrock. PassChords: Secure multi-touch authentication for blind people. In *Proceedings of the 14th international ACM SIGACCESS conference on Computers and accessibility*, ASSETS '12, pages 159 - 166, ACM, New York, USA. 2012.

[4] Jacob O. Wobbrock. TapSongs: Tapping rhythm-based passwords on a single binary sensor. In *Proceedings of the 22nd annual ACM symposium on User interface software and technology*, UIST '09, pages 93 - 96, ACM, New York, USA, 2009.

[5] Shiri Azenkot, Jacob O. Wobbrock, Sanjana Prasain and Richard E. Ladner. Input finger detection for nonvisual touch screen text entry in perkinput. In *Proceedings of Graphics Interface*, GI '12, pages 121 - 129, Canadian Information Processing Society, Toronto, Canada, 2012.

[6] Ravi Kuber and Shiva Sharma. Toward tactile authentication for blind users.

In *Proceedings of the 12th international ACM SIGACCESS conference on Computers and accessibility*, ASSETS '10, pages 289 - 290, ACM, New York, USA, 2010.

[7] Denis Foo Kune and Yongdae Kim. Timing attacks on PIN input devices. In *Proceedings of the 17th ACM conference on Computer and communications security*, pages 678 - 680, CCS '10, ACM, New York, USA, 2010.

[8] Anthony Delehanty. Security issues in biometric identication. In *Information Assurance Workshop*, IAW 05, Proceedings from the Sixth Annual IEEE SMC, Pages 8 - 13, June 2005.

[9] Diogo Marques, Tiago Guerreiro, Lus Duarte, Lus Carrio. Under the table: Tap authentication for smartphones. In *Proceedings of the 27th International BCS Human Computer Interaction Conference*, BCS-HCI '13, British Computer Society Swinton, UK. September 2013.

[10] Yimin Chen, Jingchao Sun, Rui Zhang, and Yanchao Zhang. Your song your way: Rhythm-based two-factor authentication for multi-touch mobile devices. In *Computer Communications (INFOCOM)*, 2015 IEEE Conference, pages 2686 - 2694, May 2015.

[11] Nan Zheng, Kun Bai, Hai Huang and Haining Wang. You are how you touch: User verification on smartphones via tapping behaviors. In *Network Protocols (ICNP)*, 2014 IEEE 22nd International Conference. pages 221 - 232, October 2014.

[12] Md Mohaiminul Haque, Shams Zawoad and Ragib Hasan. Secure techniques and methods for authenticating visually impaired mobile phone users. In *Technologies for Homeland Security (HST)*, 2013 IEEE International Conference, pages 735 - 740, November 2013.

[13] Diogo Marques, Lus Carrio and Tiago Guerreiro. Assessing inconspicuous smartphone authentication for blind people.

[14] Ahmed Sabbir Arif, Michel Pahud, Ken Hinckley and Bill Buxton. A tap and gesture hybrid method for authenticating smartphone users. In *Proceedings of the 15th international conference on Human-computer interaction with mobile devices and services*, MobileHCI '13, pages 486 - 491, ACM New York, USA, November 2013.

[15] Ioannis Leftheriotis. User authentication in a multi-touch surface: a chord password system. In *Extended abstracts on human factors in computing systems*, CHI '13, pages 1725 - 1730, ACM, New York, USA, April 2013.

[16] Shaun K. Kane, Chandrika Jayant, Jacob O. Wobbrock, and Richard E. Ladner. Freedom to roam: A study of mobile device adoption and accessibility for people with visual and motor disabilities. In *Proceedings of the 11th international ACM SIGACCESS conference on Computers and accessibility*, pages 115 - 122, ACM, New York, USA, October 2009.

[17] Vassilis Kostakos and Eamonn ONeill. Human-in-the-loop: rethinking security in mobile and pervasive systems. In *Extended Abstracts on Human Factors in Computing Systems*, CHI '08, pages 3075 - 3080, ACM New York, USA, April 2008.

[18] Markus Jakobsson, Elaine Shi, Philippe Golle and Richard Chow. Implicit authentication for mobile devices. In *HotSec'09 Proceedings of the 4th USENIX conference on hot topics in security*, pages 9 - 9, USENIX Association Berkeley, CA, USA, November 2009.

[19] Wood, Helen M. The use of passwords for controlled access to computer resources. In *National Bureau of Standards Special Publication 500-9*, U.S. Department of Commerce/NBS, May 1977.

[20] NoamBen-Asher, Niklas Kirschnick, Hanul Sieger, Joachim Meyer, Asaf Ben-Oved and Sebastian Mller. On the need for different security methods on mobile phones. In *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services*, pages 465 - 473, ACM New York, NY, USA, August 2011.

[21] Ian Jermyn, Alain Mayer, Fabian Monrose, Michael K. Reiter, and Aviel D. Rubin. The design and analysis of graphical passwords. In *Proceedings of the 8th conference on USENIX Security Symposium*, SSYM'99, Pages 1-1, USENIX Association Berkeley, CA, USA, August 1999.

[22] N. Asokan and Cynthia Kuo Usable mobile security. In *Proceedings of the 8th international conference on Distributed Computing and Internet Technology*, ICDCIT'12, pages 1 - 6, Springer-Verlag Berlin, Heidelberg, February 2012.

[23] David Kim, Paul Dunphy, Pam Briggs, Jonathan Hook, John Nicholson, James Nicholson and Patrick Olivier. Multi-touch authentication on tabletops. In *Proceedings of the SIGCHI Conference on Human Factors in Computing*, CHI '10, pages 1093 - 1102, ACM, New York, USA, April 2010.

[24] N.L. Clarke a, S.M. Furnell. Authentication of users on mobile telephones: A survey of attitudes and practices. In *Computers and Security*, pages 519 - 527, Elsevier Advanced Technology Publications Oxford, UK, October 2005.

[25] Frode Eika Sandnes and Xiaoli Zhan. User identification based on touch dynamics. In *Ubiquitous Intelligence Computing and 9th International Conference on Autonomic Trusted Computing (UIC/ATC)*, Pages 256 - 263, September 2012.

[26] Jingchao Sun, Rui Zhang, Jinxue Zhang, and Yanchao Zhang. TouchIn: Sightless two-factor authentication on multi-touch mobile devices. In *The 2nd IEEE Conference on Communications and Network Security*, CNS 2014, San Francisco, USA, October 2014.

[27] David Kim, Paul Dunphy, Pam Briggs, Jonathan Hook, John Nicholson, James Nicholson and Patrick Olivier. Multi-touch authentication on tabletops. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, pages 1093 - 1102, ACM, New York, USA, April 2010.

[28] Hristo Bojinov and Dan Boneh. Mobile token-based authentication on a budget. In *Proceedings of the 12th Workshop on Mobile Computing Systems and Applications*, HotMobile '11, pages 14 - 19, ACM, New York, USA, March 2011.

[29] Alexander De Luca, Alina Hang, Frederik Brudy, Christian Lindner and Heinrich Hussmann. Touch me once and I know its you! Implicit authentication based on touch screen patterns. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, pages 987 - 996, ACM, New York, NY, USA, May 2012.

[30] Apple VoiceOver. http://www.apple.com/accessibility/ios/voiceover/

[31] Android TalkBack. https://support.google.com/accessibility/androidtopic=3529932

[32] Apple Touch ID. https://support.apple.com/en-us/HT201371