KEY-LOGGING THREAT TO THE ANDROID MOBILE BANKING APPS

BY

MD. SALAUDDIN RUBEL
ID: 161-15-7033

AND

KHANDOKER SHAIDUR RAHAMAN
ID: 161-15-7389

This Report Presented in Partial Fulfillment of the Requirements for the Degree of Bachelor of Science in Computer Science and Engineering

Supervised By

Mr. Naziour Rahaman
Lecturer
Department of CSE
Daffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY

DHAKA, BANGLADESH

DECEMBER 2019

APPROVAL

This Project titled **"Keylogging Threat to The Android Mobile Banking Apps"**, submitted by Md. Salauddin Rubel, ID No: 161-15-7033 and Khandoker Shaidur Rahaman, ID No: 161-15-7389 to the Department of Computer Science and Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on 05/12/2019.

BOARD OF EXAMINERS

Dr. Syed Akhter Hossain **Chairman**
Professor and Head
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Abdus Sattar **Internal Examiner**
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Farah

Farah Sharmin **Internal Examiner**
Senior Lecturer
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Saddam

Dr. Md. Saddam Hossain **External Examiner**
Assistant Professor
Department of Computer Science and Engineering
United International University

DECLARATION

We hereby declare that this project has been done by us under the supervision of **Mr. Naziour Rahman, Lecturer, Department of CSE** Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

Supervised by:

09.12.19

Mr. Naziour Rahaman
Lecturer
Department of CSE
Daffodil International University

Submitted by:

Salauddin

Md. Salauddin Rubel
ID: 161-15-7033
Department of CSE
Daffodil International University

Khandoker Shaidur Rahaman
ID: 161-15-7389
Department of CSE
Daffodil International University

ACKNOWLEDGEMENT

First, we express our heartiest thanks and gratefulness to Almighty God for His divine blessing makes us possible to complete the final year project/internship successfully.

We really grateful and wish our profound our indebtedness to **Mr**. **Naziour Rahaman**, **Lecturer**, Department of CSE Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the field of "*Information Security*" to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts and correcting them at all stages have made it possible to complete this project.

We would like to express our heartiest gratitude to Almighty Allah and Head**,** Department of CSE, for his kind help to finish our project and also to other faculty member and the staff of CSE department of Daffodil International University.

We would like to thank our entire course mate at Daffodil International University, who took part in this discussion while completing the course work.

Finally, we must acknowledge with due respect the constant support and patience of our parents.

ABSTRACT

The android platform offered numerous services to make our life easy. Third-party android developers are getting a large area for development, it’s also become a huge interest to the modern attackers to steal user’s sensitive information using this platform. To spy on smartphone users, attackers can build own keyboard application or take advantage of existing third-party apps. Most of the mobile banking and social networking apps keystroke data such as login pin, password and credit card number can be easily stolen by key-logger app. Key-logger apps are basically blocked in android app store but using some app permission vulnerabilities key-loggers can be installed with some trusted and benign apps. Rather than other applications in the android phone Mobile Banking Application faces more security threats. In this paper, we discuss the abuse of android app permissions and installing key-logger apps to steal mobile banking data for the financial gain of attackers. We also discuss possible ways to avoid such a key-logging attack.

TABLE OF CONTENTS

CONTENTS **PAGE**

Board of examiners i

Declaration ii

Acknowledgements iii

Abstract iv

CHAPTER

CHAPTER 1: INTRODUCTION 1-3

1.1 Introduction 1

1.2 Motivation 1

1.3 Rationale of the Study 2

1.4 Research Question 2

1.5 Expected Output 2-3

1.6 Report Layout 3

CHAPTER 2: BACKGROUND 4-7

2.1 Introduction 4

2.2 Related Work 4

2.3 Comparative Studies 5

2.4 Report Layout 5

2.5 Challenge 6-7

CHAPTER 3: RESEARCH METHODOLOGY 8-14

3.1 Introduction 8

3.2 Research Subject and Instrumentation 8-10

3.3 Data Collection Procedure 10

3.4 Statistical Analysis 11-13

3.5 Implementation Requirements 14

CHAPTER 4: EXPERIMENTAL RESULT AND DISCUSSION **15-23**

4.1 Introduction 15

4.2 Experimental Result 15-16

4.3 Descriptive Analysis 16-20

4.4 Proposed Method 20-23

CHAPTER 5: SUMMARY, CONCLUSION, RECOMMENDATION AND IMPLICATION FOR FUTURE RESEARCH **24-25**

5.1 Summary of the Study 24

5.2 Conclusion 24

5.3 Recommendation 25

5.4 Implication for the Further Study 25

REFFERNCES **26-27**

LIST OF FIGURES

FIGURES **PAGE NO**

Figure 3.1: Key-logger attack scenarios…………………………………….…...……..…9

Figure 3.2: Android keyboard attack graph……………….……………………..…...…....10

Figure 3.3: App permissions used in Mobile Banking App…………………......….…..11

Figure 3.4: App permissions frequency in a Mobile Banking App………………...…..12

Figure 4.1: Keystroke values stored in our file………………………....…....……....15

Figure 4.2: Ineffective warning message on the mobile…………………..…....……...16

Figure 4.3: Application sandboxes, ICC calls and permission assigned……....……....18

Figure 4.4: Xmandroid Architecture……………………………....……….……..…19

Figure 4.5: Numeric Keyboard in Android……………………...…….….……….…21

Figure 4.6: Pin-Crypto method to avoid keylogger attack…………………………...22

Figure 4.7: Changed keyboard label value………..……………………....……....…..23

LIST OF FIGURES

FIGURES **PAGE NO**

Table 3.1 : Mobile Banking Application Permission Purpose......................................12

CHAPTER 1
INTRODUCTION

1.1 Introduction

Mobile banking becomes a trend in the banking business because of its ease in covering high mobility. Banking apps have long been the target of hackers, it is very essential to ensure app security. People using mobile banking services must know about the possible cybercrime that threatens their financial accounts. There are many types of attack can take place in a mobile banking application. One example of attacks on the mobile banking applications that may occur is criminal techniques using the keylogger [1].

A bunch of capabilities and flexibilities that android have offered is attracted by the users, developers, and attackers. As we know that before installing any app android ask for the app permissions, most of the time people ignore these permissions. A third-party app can be installed in two ways, download google play store or manually installing apk. Google play store has blocked all the types of keylogging app. So the only way to install a keylogger app is manually installing it. But keylogger gets installed as a benign app or with a benign application. User cannot understand the hidden permission of the application, keylogger take this opportunity and get installed. Keylogger started keylogging after getting installed. Mobile banking application are also not safe from keylogging attacks. A keylogger can collect applications pin and password, which can cause unauthorized access to the application. To avoid such an attack different method can be applied. In our study we have discuss about some method to avoid keylogging attack.

1.2 Motivation

The customary banking system is always an inconvenience to me, by the evaluation of the technology banking system become much easier. Now, most of the banking transactions can be done by using a mobile phone. I am using the mobile banking application for my money transactions and a large number of people also use these applications. I also used some other third-party applications for different purposes. There are some vulnerabilities in third party application and mobile banking application is all about money, so attackers have a huge interest in these applications.

1.3 Rationale of the Study

In recent years, cyber-attacks on financial services companies have globally increased by 40% from 2014 to 2017. A study from The Cost of Cyber Crime Study [2] has found that the cost is $12.97 million per company in 2014 to $18.28 million in 2017. In the world, 2.7 billion smartphone users, it's no surprise that the mobile app industry is thriving. The mobile banking application is placed as a finance category in-app store. During September 2019 the user of finance android app categories is 25.75% [3]. The number of mobile banking application user is increasing on that basis the security level is not increased. To ensure a secured transaction we have to provide the best security to the mobile banking application. This study is specifically discussed about keylogging attack and prevention of the attack in android mobile banking applications.

1.4 Research Questions

a) How key-logger can affect Mobile Banking App security?

b) How key-logger gets installed?

c) Why too many permissions in a Mobile Banking Application?

d) How to avoid keylogger attacks?

e) Can our proposed method avoid the keylogging attack?

1.5 Expected Output

A keylogging attack can make mobile banking transactions insecure. From this study, we can find the way how the keylogger can place an attack and the method to avoid this type of attack. At first, we talk about the keylogger attacking procedure, from that we can know that how keylogger can get installed in our android devices. We have also discussed the android application permissions. It is very important to know about the purpose of the permissions and which permissions perform what action. Sometimes a little abuse of app permissions cause a major security threat to the android devices. Finally, we discussed the method of avoiding the keylogging attack. Our proposed security method can ensure better security than other security methods used in android mobile banking applications.

1.6 Report Layout

In chapter one we have demonstrated an introduction to the project with motivation, the rationale of the study, research question and expected output, the layout of the whole report is described in this section.

Chapter two describes the previous work done in this domain. Then the later section shows comparative studies and scope of the problem. The root obstacles and challenges are explained in the later section of the chapter.

Chapter three is all about the research methodology. Here we describe the research subject and instrumentation. The data collection procedure for the research is discussed here. Statistical analysis of the Android Mobile Banking Application permissions is shown in the section. And very last the implementation requirements for the project are described.

In chapter four, the experimental result, a descriptive analysis of the experiments is discussed. There is some experimental figure are shown in the chapter for better realization.

Chapter five provides a summary of the study and the conclusion of our project. A recommendation from the study and scope of the future study is discussed in this chapter.

CHAPTER 2
BACKGROUND

2.1 Introduction

There is a plenty number of work based on Keyloggers but they are basically focused on computers. The number of smartphone users is increasing day by day and mobile banking app are become more popular, ensuring the security of these apps is very important. Android is the most used operating system in the world. The numbers of mobile banking applications are increasing which creates more security flaws. Mobile banking application gives the privileged to transfer money without going a bank. It is a 24/7 service which makes our life so easy. Any type of bill payment can be done on a mobile banking application. Android is a user-friendly operating system that provides the banking firms to develop their own banking application and publish them to the huge number of android users. Key logger is one of the cyber-attack that can exploit both of the systems.

2.2 Related Work

There is a plenty number of work based on Keyloggers but they are basically focused on computers. The number of smartphone users is increasing day by day and mobile banking app are become more popular, ensuring the security of these apps is very important.

In a research paper, Fadi Mohsen has introduced several attacks from third-party keyboards [4]. Their study mainly focused on analyzing exiting keyboard permission information that could be misused which may not be sufficient.

Recently Adam Prayogo Kuncoro has described the process of stealing data from the mobile banking app [5]. But their study doesn’t specific with the mobile banking app and they didn’t propose any method to avoid the keylogging attacks. In contrast, we demonstrated how such a keylogger app monitors the mobile banking apps through the intensive test with a possible solution.

2.3 Comparative Studies

Android help us make our life easy by providing several services in smartphone. From the time when the banking system is invented, the importance of banks is always increased. We know that a bank is a financial institution that accepts deposits from the public and creates credit directly or indirectly through capital markets. But at the beginning banking is a very complex and time-consuming issue. With the invitation to computers that become very easy. And now the improvement of mobile technology makes the banking system much easier than before. Daily millions of dollar transaction are done by mobile banking applications all over the world. Hackers and scammers are very interested in this kind of application because these applications deal with money. By anyhow if anybody gets unauthorized access, he can causes a huge number of money loss. Rather than any other application in android, mobile banking applications get a high priority in terms of security issues. Android uses sandboxing for each of the applications, but the default services are the same for all the applications. Any kind of security violation can happen by breaking the sandboxing rules.

2.4 Scope of the Problem

Banks are responsible for ensuring a safe transaction of money in a mobile banking application. Fraud and scam in the banking system is daily offense, mobile banking is such a platform which at the peak of this kind of thread. Both mobile banking and keylogger are third-party applications in android. Developers can be designed their application with their requirements, here the main threat is started. Attackers can be work as a developer and can place any type of attack to other application in android. Mobile banking has more risk than other applications because it is directly related to money. Keylogger attacks can take place in mobile banking applications and can share user sensitive information including account pin. Most of the people are not conscious about mobile applications permissions and services, they install any type of application without knowing the safety of the application. For the huge number of mobile banking applications, user keylogger attacks can be causes any type of security violation.

2.5 Challenges

Android has a huge number of users all over the world and a large number of users use mobile banking applications for their banking transactions. It is very challenging when we talk about such the biggest platform's security framework. While researching this topic we have to faces a lot’s of challenges. Some of the major challenges are described below:

i. **Finding Keylogger Application**: Effective keylogger application are not publicly available. Most of them are paid or only found in dark web, some of them don’t even reveal to the public, only a hacker can access that application. There are many free keylogger applications available on different website but they are not effective. In google play store directly uploading a keylogger application is banned and if any developer uploads a keylogger application it will be removed within a few hours. To implement the keylogger effect in mobile banking application we have to find an effective keylogger.

ii. **Access to a Mobile Banking Application:** For checking keylogger effect in different mobile banking applications we have to access the banking application. As we all know that if we don’t have a banking account we cannot access the banking application. For accessing these mobile banking applications we have to open a mobile banking application. For this purpose accessing mobile banking applications from other countries is unreachable.

iii. **Choosing Keyboard**: A mobile banking application required both number and text value for different procedures. Most of the input is typed by an android built-in keyboard which can be easily traced by the keyloggers. Even mobile banking applications that have their own keyboard can be recorded by keyloggers. To stop keyloggers from accessing keystroke we have to find a suitable and reliable keyboard for the mobile banking application.

iv. **Password/Pin field type:** Consisting mobile banking application has a 4/5 digit number field. The number of the digit is not the problem, the problem happened when we try to change the original value into cryptographic value which is described later (see Section 4.4.2), from there we find very few combinations of pin/password pattern. For finding an optimal and secured password pattern we have to change the password filed type from number to text.

v. **Pin Encryption Method:** To encrypt the pin keystrokes we have to choose a better way to for encryption. Symmetric and asymmetric cryptography is used for encryption. As our proposed model is real time base encryption and decryption, we need a fast and secured method for this purpose. In our proposed model we used the symmetric encryption method which is discussed in later sections.

CHAPTER 3
RESEARCH METHODOLOGY

3.1 Introduction

We have learned all the convenience of approaches and their abridgments and all available possibilities after investigating all exiting approaches. Different approaches has a different solution but we should come up with the optimal solution. Our proposed methodology is described in this chapter of our study.

3.2 Research Subject and Instrumentation

Our research is based on the security of the Android application system along with the mobile banking system. Mobile-based baking services have increased, our study is mainly focused to ensure proper security of these applications. There are several attacks can take place in an android application. We are talking about one of the major attacks which are known as keylogger attacks. Mobile banking app provides services of a bank or other financial institution that allows users to conduct financial transaction remotely using a mobile device such as smartphone or tablet [6].

3.2.1 Mobile Banking Services

Account Information: Information about mini-statement, account transaction history, and alert on account activity. Give access to card statements, loan statements, and mutual funds. Also monitors the term deposit.

Transaction: Check remote deposit, paying third parties bill, funds transfer between users linked account.

Support: Checkbook and card request, ATM location, exchange of data messages and email, including complaint submission and tracking.

Investments: Real-time stock and portfolio services.

Content Services: Loyalty-related offers, location-based services and General information such as weather updates, news.

3.2.2 Key-logger Attacking Procedure

Keylogger is a very old cyber-attack. With the development of technology security is level is updated and the key-logger attacking approach has been updated also. Keyloggers keep track of all keystrokes while using an android smartphone and then transfers the information to a remote server. An app can be installed from the play store or manually installing apk. Keylogger app can be installed with a benign app or pop-up ad. After getting installed app started to record the keystroke of the user. Keystroke data saved in a file and this file uploaded to the attacker server. There is also some keylogger that can upload the keystrokes in a real-time.

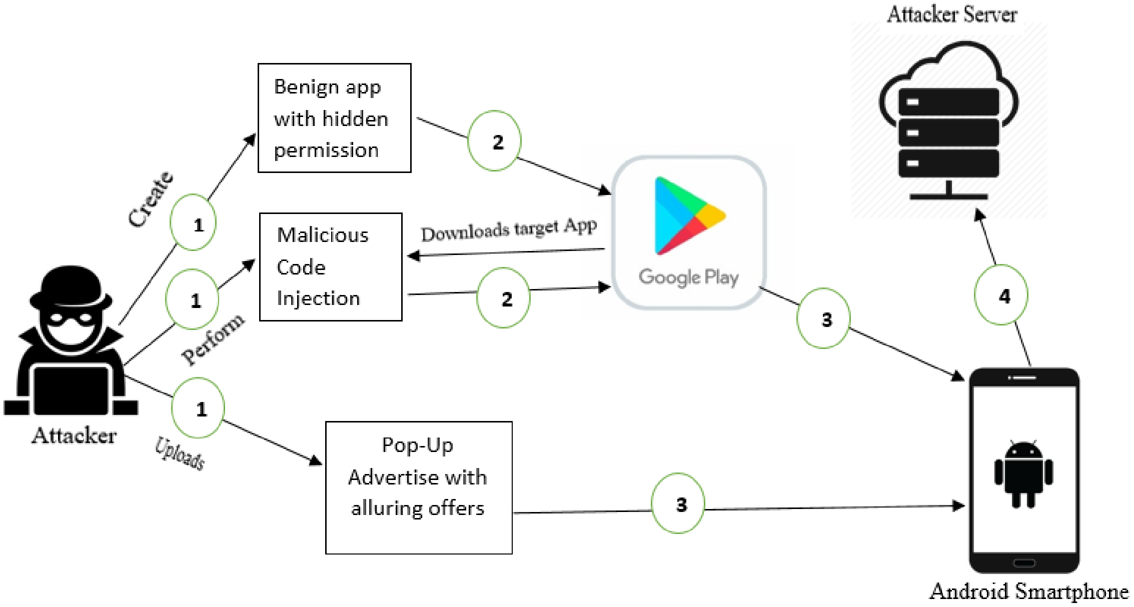


Figure 3.1: Key-logger attack scenarios

Another attack graph of keylogger according to android permission is described by Fadi Mohsen (see figure 3.2) [4]. This figure shows the process of collecting, sending and storing the process of keylogger.

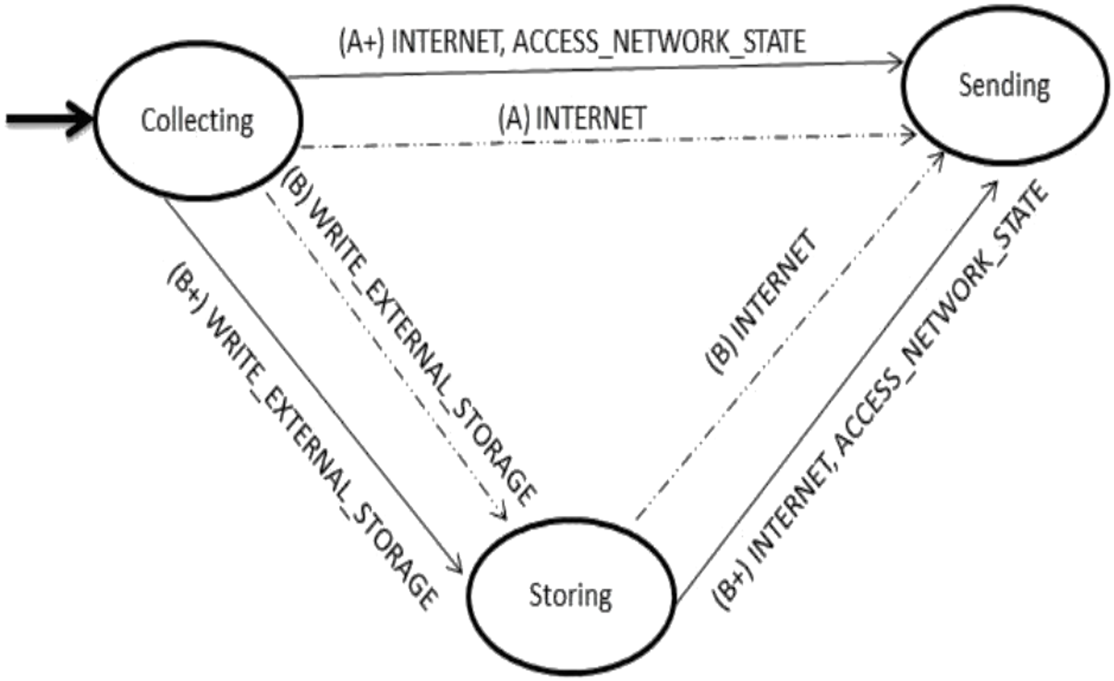

Collecting
Sending
Storing
(A+) INTERNET, ACCESS_NETWORK_STATE
(A) INTERNET
(B) WRITE_EXTERNAL_STORAGE
(B+) WRITE_EXTERNAL_STORAGE
(B) INTERNET
(B+) INTERNET, ACCESS_NETWORK_STATE

Figure 3.2: Android keyboard attack graph

For the following method there are several permissions needed to perform the attack which includes WRITE_EXTERNAL_STORAGE, INTERNET, and ACCESS_NETWORK _STATE. This method is applicable for keylogging by a third-party keyboard.

3.3 Data Collection Procedure

We collect our primary data from the Google Play store. At first, we collect different mobile banking application names available google play store according to country. We have collected about 50 different application from 15 different countries. Then we find all the permissions that each mobile banking app asks before install. We also collect some statistical information from different websites. Google developer website has a huge number of content that helps us to find the required data [7].

3.4 Statistical Analysis

We explore potential risks associated with the Mobile Banking app in the Google play store. We empirically analyzed whether those existing mobile banking app can avoid key-logger or not by observing their characteristics. We downloaded 50 mobile banking

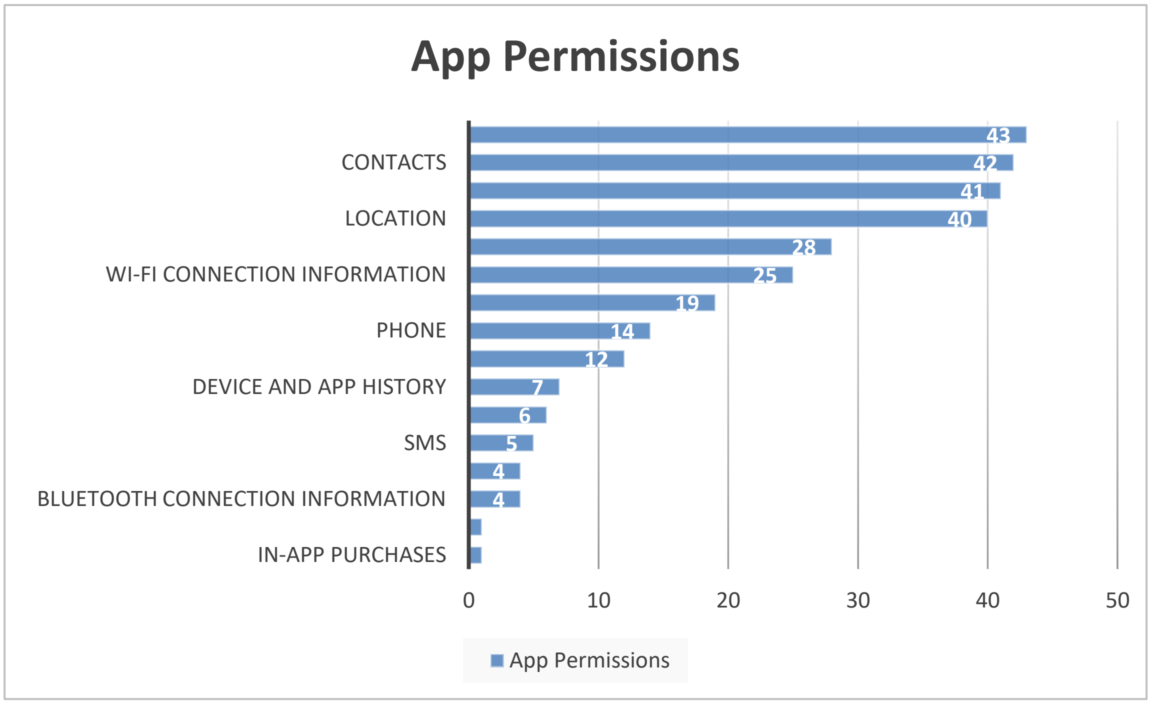


Figure 3.3: App permissions used in the Mobile Banking App

-applications that were freely available on the Google Play store. Our analysis particularly focused on two points: (1) number and types of requested permissions and (2) purpose of that permission. To analyze the characteristics of these permissions, we looked at the most commonly requested permissions. Figure 3.4.2 shows the list of the top 16 most commonly requested permissions and their distribution.

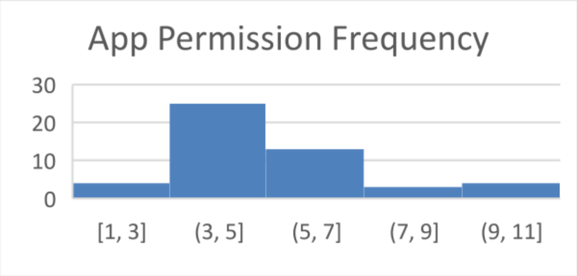


Figure 3.4: App permissions frequency in a Mobile Banking App

Android has supports inter-component communication (ICC) that allows App communication across the sandbox [8]. Using too much permission can cause a security violation. In inter-component communication, one application can use other application permissions. We have found the purpose of the application for using different permissions. From the following table, we can analyze the permission usage in detail.

Table 3.1: Mobile Banking Application Permission Purpose

Permission Name	Permission Description	Permission in Manifest	Secured Thread
In-app purchases	Allows users to purchases within app	com.android.vending.BILLING	Low
Device and app history	Information about activity on the devices, which apps are running, browsing history and bookmarks	android.permission.READ_LOGS android.permission.DUMP com.android.browser.permission.READ_HISTORY_BOOKMARKS android.permission.GET_TASKS	Medium
Identity	Uses one or more of account on the device, profile data	android.permission.GET_ACCOUNTS	Dengerous
Calendar	Uses calendar information	android.permission.WRITE_CALENDAR android.permission.READ_CALENDAR	Low
Contacts	Uses contact information	android.permission.READ_CONTACTS	Medium
Location	Uses the devices location's	android.hardware.location.gps	Dengerous
SMS	Uses one or more of: SMS, MMS. Charges may apply.	android.permission.SEND_SMS	Medium
Phone	Allow the app to make and manage phone calls.	android.permission.ANSWER_PHONE_CALLS	Low
Photos/Media/Files	Files on the devices, such as images, videos, or audio, the device's external storage	android.permission.READ_EXTERNAL_STORAGE android.permission.WRITE_EXTERNAL_STORAGE	Dengerous
Camera	Uses the devices camera	android.permission.CAMERA	Medium

Microphone	Uses the devices microphone	android.permission.RECORD_AUDIO	Dengerous
Wi-Fi connection information	View information about Wi-Fi networking such as whether Wi-Fi is enabled and names of connected Wi-Fi devices	android.permission.INTERNET android.hardware.wifi	Medium
Device ID & call information	Allows the app to determine the phone number and devices IDs, whether a call is active and the remote number connected by a call	android.permission.READ_CALL_LOG	Low
Bluetooth connection information	control bluetooth including broardcasting to or getting information about nearby bluetooth devices	android.permission.BLUETOOTH android.permission.BLUETOOTH_ADMIN	Low
Wearable sensors/ activity data	Allows the app to access data from wearable sensors such as heart rate monitors. Can receive periodic updates on physical activity levels.	Depends on devices	Low
Others	Draw over the apps	Depends on devices	Low

3.5 Implementation Requirements

For our study we need an android development environment to analyze the security of different applications. Our study finds security flaws in mobile banking applications. That’s why we also need a banking application to test our different approaches. All the requirements are described below:

i) Android System: In our study, we discuss the mobile banking application security in android. We need an android platform to check our different methods. Android is the most used mobile operating system and mobile banking applications are dependent on it.

ii) **Banking Overview**: To understand the unauthorized money transfer, at first we have to know the legal money transaction process. The system can define if the transaction happened legal or illegal.

iii) **Mobile Keyboard Structure**: Keylogger attack is based on keyboard. User typed keystroke can be traced by a keylogger. So we must know about the structure of a keyboard system.

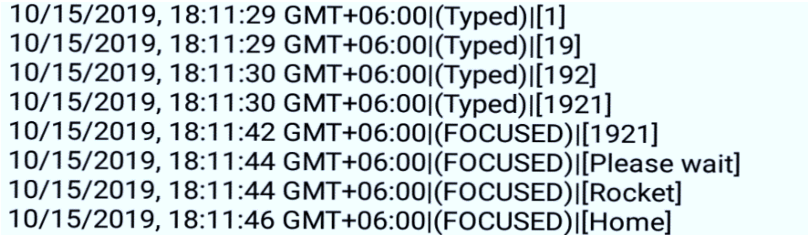
CHAPTER 4
EXPERIMENTAL RESULT AND DISCUSSION

4.1 Introduction

Finding the way how a keylogger can attack a mobile banking application is complex. Although there is a numerous security protocol used in a mobile banking application, still there is some security threat left. This section is all about to find out the security threat in details especially keylogger attacks and perform a suitable solution for the problem. We have used a bunch of android tools for our experimental procedure.

4.2 Experimental Result

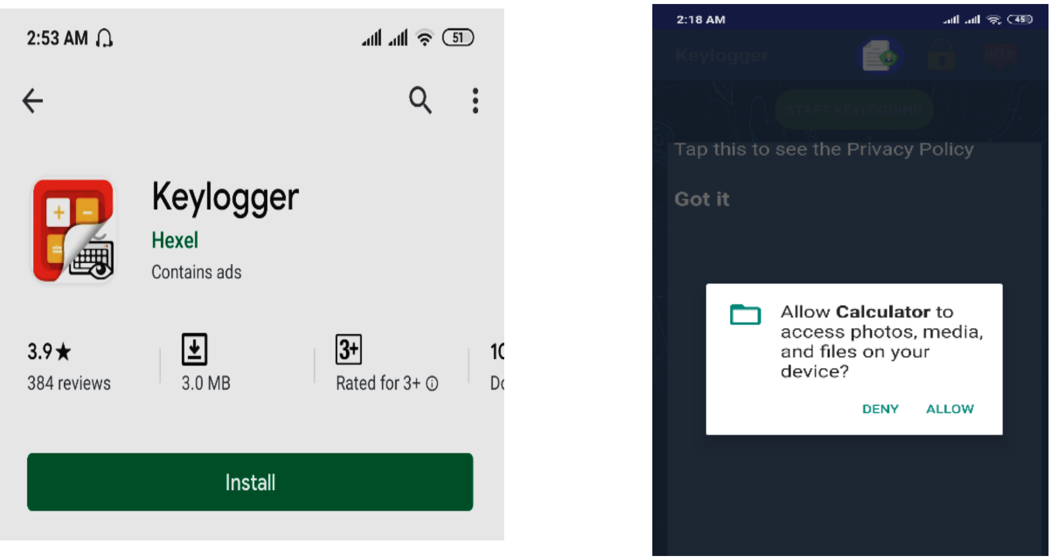
In section 3.2.2 we have discussed about the keylogger attacking scenarios. That scenario is based on third-party keyboard keylogger attack. Now for a keylogger attack, it is not mandatory to create a keyboard, it can be done by even without designing a keyboard model. The terms keylogger without keyboard is recently developed, many keyloggers can place an attack without visually connected to the system keyboard [9]. Advanced keylogger app can check whether the typed keystrokes are valuable or not, it can tokenize the keystrokes into word set according to input pattern. For example, if user type pin 1234, it will be recorded as a single-digit 1,2,3,4 but at end of typing it will organize the single-digit into input pin 1234 (see Figure 4.1). For sending recorded keystroke file the keylogger app uses the INTERNET permission, this network delivery process run on the app background which the victim user may not be aware of that activity. We tested this overall keylogger process with Rocket Mobile Banking App and confirmed that our proof-of-concept keylogger application works well.



```
10/15/2019, 18:11:29 GMT+06:00|(Typed)|[1]
10/15/2019, 18:11:29 GMT+06:00|(Typed)|[19]
10/15/2019, 18:11:30 GMT+06:00|(Typed)|[192]
10/15/2019, 18:11:30 GMT+06:00|(Typed)|[1921]
10/15/2019, 18:11:42 GMT+06:00|(FOCUSED)|[1921]
10/15/2019, 18:11:44 GMT+06:00|(FOCUSED)|[Please wait]
10/15/2019, 18:11:44 GMT+06:00|(FOCUSED)|[Rocket]
10/15/2019, 18:11:46 GMT+06:00|(FOCUSED)|[Home]
```

Figure 4.1: Keystroke values stored in our file

We performed additional experiments to ensure the keylogger competency. We login into some of the most used social networking applications in android, check whether the login data are successfully saved in our keylogger file. From our test, we find that this keylogger can also collect their login password and even trace the messenger incoming messages and Gmail inbox data. When a user tries to install our application, the warning message about permissions requested by the application is not too effective (see Figure 4.2(b)).



Tap this to see the Privacy Policy

Got it

Allow **Calculator** to access photos, media, and files on your device?

(a) App Information

(b) Installation Warning

Figure 4.2: Ineffective warning message on the mobile

4.3 Descriptive Analysis

Individually keylogger attacks can be prevented by anti-virus software. Users consciously do not install keylogger application. There are some security flaws which is visually untraceable, but they can causes a keylogger attack. One application can use other applications permission by Inter-App Communication system. There is another term which is known as XManDroid, can monitor and analyze communication links over the applications. Mobile banking application can work properly with its required permission. But the problem occurs when another application accesses mobile banking application permission. Both can be causes a keylogger attack in mobile banking applications. Let’s consider Application A has calling permission but it has no message sending permission and Application B have message sending permission. By using ICC / Xmandroid framework Application A can be accessed Application B’s permission. So, if any benign application has much permission, other fraud applications with few permission can get unwanted privileged. For Example, You have both Paytm and CandyCrush application installed in your android device. Paytm have Camera permission for QR scanning but CandyCrush has no camera permission. But CandyCrush can access Camera by using the ICC framework. For more clarification, we discuss it in the below sections.

4.3.1 Inter-App Communication

Sandbox test/detect the malware by execution codes behavior and output activity. In this time code keep in save. Signature-based detection is traditional and works by pattern identification. Traditional methods detect with have attacking pattern previously where sandbox provides another dimension of security. Alongside with signature-less detection (artificial intelligence), that’s are model powering solutions and still needed advanced malware detection. According to organizational requirements, several options exist.

Verities of sandbox implementations:

i. **Full System Emulation**: According to full system emulation. The sandbox simulates the host machine’s physical hardware, along with CPU and memory, providing deep visibility into program behavior and impact.

ii. **Emulation of Operating Systems**: In this method, the sandbox emulates the end user’s operating system without caring about machine hardware.

iii. **Virtualization**: Using a virtual machine (VM) based sandbox to contain and examine suspicious programs.

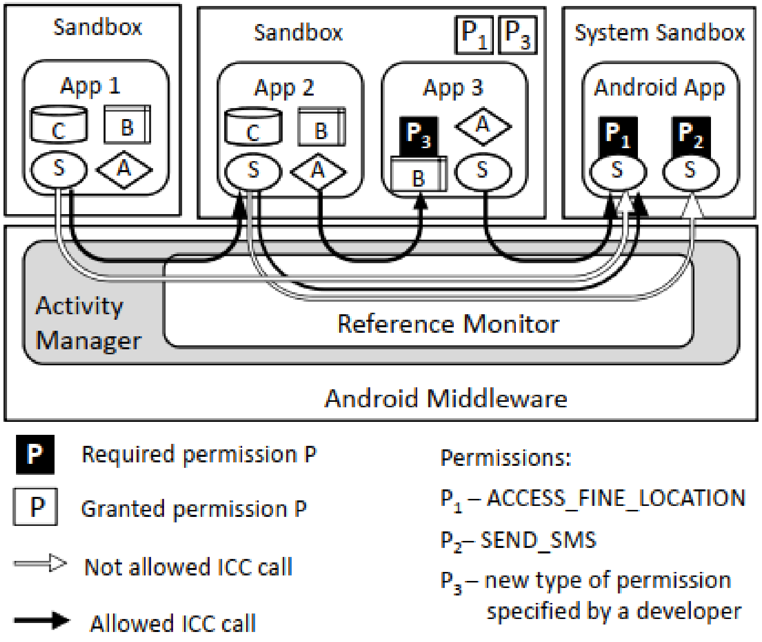


Figure 4.3: Application sandboxes, ICC calls, and permission assigned

This android security mechanism is used in ICC, by violating any of the protocol attackers can get some unprivileged permissions [10]. At first isolate applications from system resource and isolation done by UID according to underlying Linux kernel enforces discretionary access control to resources by user ownership. Resources are owned by root or system. Applications can access which have the explicitly marked ass worldwide readable.

4.3.2 Extended Monitoring

In recent years, most of the android device is face to face different kinds of attacks. Nowadays most of the attacks are framework base attack and application-level privilege escalation attacks. XManDroid is monitoring and analysis the communication links over the applications. It provides a system where application installed and established communication links. XManDroid is being called when an android reference grants an ICC call [11]. It also stores the decisions and conditions over time. Next time these decisions are applied for the same kinds of ICC calls.

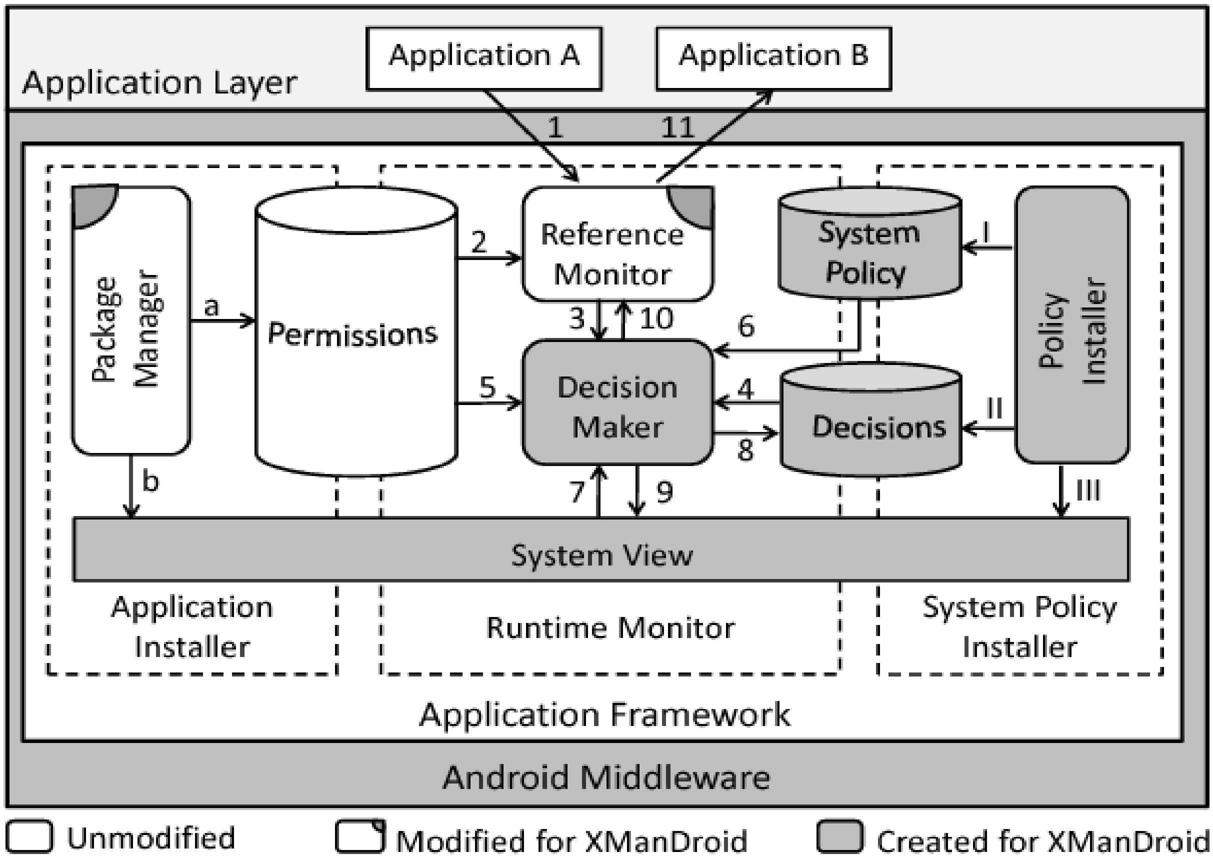


Figure 4.4: Xmandroid Architecture

The architecture of XManDroid is depictured in the previous figure. Providing a framework extends Android's middleware.It also deploys mandatory access control (MAC) on the kernel level to enforce access control to the file system (files, Unix domain sockets) and access Internet sockets. This framework relies on a system-centric policy to introduce and uses an appropriate high-level policy language at the middleware layer. At the kernel level, we have adopted TOMOYO Linux to enforce MAC. Here provide a callback channel between the kernel and middleware. Xmandroid framework is described below:

Runtime Monitor: The Runtime Monitor provides the core functionality of XManDroid. It addresses the components Reference Monitor, Decision Maker, System View, Permissions, System Policy and Decisions. Reference Monitor is the standard reference monitor of Android. Which provides the facilities of making a decision according to ICC call, maintain the state of the running system, permission database, and store decision.

Application Installer: Application Installer provides standard Android application installation procedure. It is responsible for the installation and un-installation of applications.

System Policy Installer: System Policy Installer provides a facility to mechanism to install or update the system policy into the Android middleware and calls the components Policy Installer, System Policy, Decisions and System View.

ICC call Handling: It makes ensure the ICC permission assignments by gathering information about that permissions.

Policy Installation: At the time of the policy installation process, Policy Installer traces and updates the system policy rules to the System Policy database for next time requirements.

4.4 Proposed Method

To avoid keylogging attack there is a numerous number of anti-virus in the market. Even google play store has a play protect security for the vulnerable application in play store market. But keylogger application can still be installed in different ways which are previously discussed. To ensure the security of mobile banking applications password from keylogging attacks, we recommend using own customized keylogger for inputting password filed. Most of the mobile banking applications used the built-in keyboard for inputting passwords and other fields. A keylogger can easily track those keystrokes and store the data. If every mobile banking application used their own customized keyboard for password input, then it will be hard for keylogger to track keystrokes. Virtual Keyboard can be an alternative for this purpose [12]. Third-party keyboards can be start key logging by itself. It will be better to avoid using third-party keyboards in mobile banking applications [4]. A customized keyboard for the password filed can be designed as a crypto-keyboard. The purpose of using crypto-keyboard is, a keylogger cannot find the absolute value typed, it will also prevent the man-in-the-middle attack (MITM) [13]. Mobile banking application now on the market have a 4/5 digit pin field. Most of them used only the number as input. If we try to crypto the pin value and we must convert the pin value into a number value. Example: if my logging pin is 1234 then, the converted crypto pin maybe 5678. This approach may cause ambiguity value or applying easy decrypting algorithm can retrieve the pin value. We proposed to change the password field from "*numberPassword*" to "*textPassword*". Changing the password filed typed, gives the access of using character, letter and symbol value as a crypto value. There will be more combinations of crypto password than the previous one. But there is a tricky policy which we recommend to use for ensuring more security. While typing pin there will be the keyboard which will contain only the numeric keyboard (see Figure 4.5).

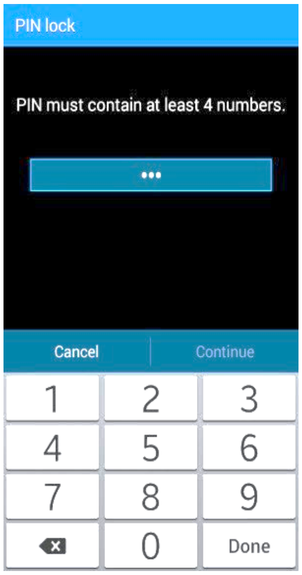

PIN must contain at least 4 numbers.

Figure. 4.5: Numeric Keyboard in Android

The mobile baking application will send the crypto pin and other information to the mobile banking server. Then there will be an engine which will decrypts the pin and then submitted to the server. The server will check all the information of the user and give the authentication for further queries. Our proposed method for avoiding a keylogging attack is given below:

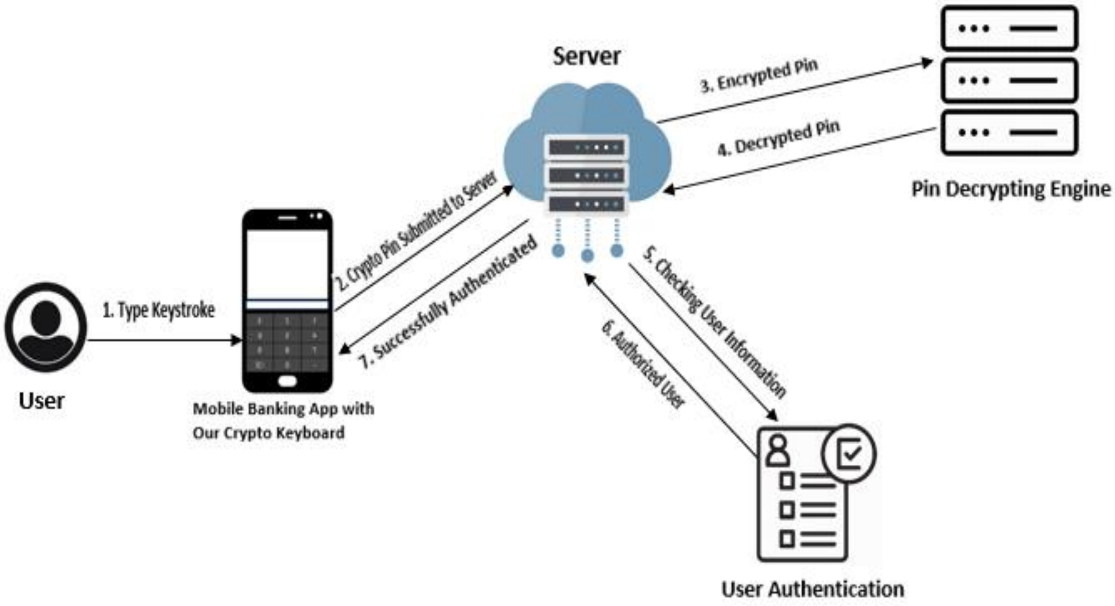
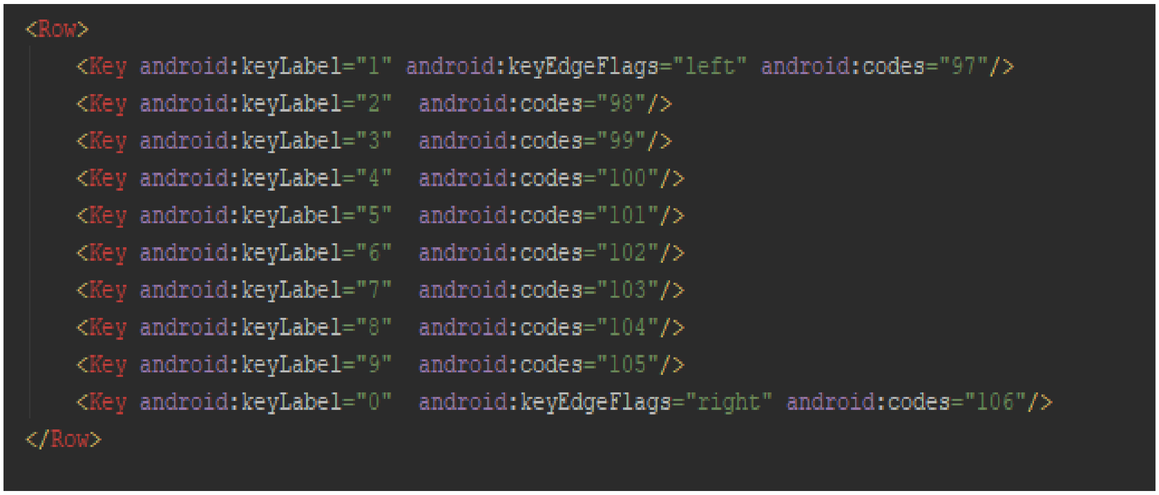


Figure 4.6: Pin-Crypto method to avoid keylogger attack

From the above scenarios, we can find the working process of Pin-Crypto method to avoid key-logging attacks. We know that most of the banking information saved in the database is cryptographic value. When the user typed their pin in mobile banking application they typed the plain value. So it is very easy for a keylogger to trace the value. Keylogger record the typed value in typing sequence. In a keyboard application, there are two parts for the value one is label of the keyboard and value of the keyboard. Label of a keyboard is visible to us but value is not visible. Value of the key is ASCII value of the character. Specific key have a specific ASCII value and every keyboard have the same value. Keylogger get the label value of the key and record into file. In our proposed model, key label will be same as other keyboard. But the value for the key will be different. If user type a key, the key value will be not same as label key. Keylogger will record the typed key value but it will be different from the label value. For Example: if user type ‘1’ in keyboard it will be recorded as ‘a’, because ‘1’ is assigned as an ASCII value of ‘a’. So keylogger recorded value will not be the actual value. The crypto typed value will decrypted in decryption engine and server will receive the real vale. By using this Crypto-Pin method can help to overcome the problem.



```
<Row>
    <Key android:keyLabel="1" android:keyEdgeFlags="left" android:codes="97"/>
    <Key android:keyLabel="2"  android:codes="98"/>
    <Key android:keyLabel="3"  android:codes="99"/>
    <Key android:keyLabel="4"  android:codes="100"/>
    <Key android:keyLabel="5"  android:codes="101"/>
    <Key android:keyLabel="6"  android:codes="102"/>
    <Key android:keyLabel="7"  android:codes="103"/>
    <Key android:keyLabel="8"  android:codes="104"/>
    <Key android:keyLabel="9"  android:codes="105"/>
    <Key android:keyLabel="0"  android:keyEdgeFlags="right" android:codes="106"/>
</Row>
```

Figure 4.7: Changed keyboard label value

Key-label is the value which is visible to the user and key codes is the ASCII value of the digit. Consisting keyboard system has key-label according to the ASCII value of the character. It is very easy for keylogger application to get the label-code values according to the key-label value. In our crypto method, we suggested that the keyboard label value should be the same as the consisting keyboard layout. But the code value of the key-label will differ from the original label value. When any keylogger gets the typed keystrokes from the keyboard it will take the crypto value of that digit, which prevent the keylogger from getting the original value. Encrypted key-label value than submitted to the server than the value will decrypted by the decrypting engine. That's will make sure the integrity of the password/pin of the mobile banking application.

CHAPTER 5

SUMMARY, CONCLUSION, RECOMMENDATION, AND IMPLICATION FOR FUTURE RESEARCH

5.1 Summary of the Study

Any kind of cyber-attack causes a huge loss to a company. Keylogger attack is one of the attacks which is virtually invisible, but it can cause a big cyber-attack. Availability of android smartphone to the mass people make the security thread more worsen. Unnecessary permissions and accessing system services by a third-party android developer, simulate a great threat to the whole system. Users can consciously or unconsciously install harmful application including keyloggers. To avoid keylogger attack system developers should built their application with a keylogger prevention method. Our proposed method can be a solution to the problem.

5.2 Conclusion

The user of android is tremendously increased day by day. Android application has a huge market share in the technology market. Both paid and free applications in the Google Play store, a third-party developer built an application and publish it into the application market. With simple security testing, the application gets authorized for the application store. Now banking services are available in android mobile. Bank authority should design an application and upload it to the play store. The banking application is secured with a security pin and password. If any unauthorized user gets access to this application, he can perform an illegal transaction. Keylogger is an old approach to steal keystroke include pin and password. For the mobile banking application keylogger, attackers can get user sensitive information. A keylogger can get installed as an application or a service. Most of the time the keylogging process cannot be identified easily. There are many prevention methods to avoid keylogging, but with advanced technology, keyloggers can still harmful. We have proposed a method which can avoid keylogging from the primary level.

5.3 Recommendation

To avoid any type of cyber-attack knowledge about security is essential. Android smartphone is available at low price and provides a lot's of services so the user is increased. But most of the user is not conscious about security about their phone. In this situation using sensitive applications like mobile banking can lead to a major cyber-attack. In our study we have shown how keylogger gets installed and the way to prevent it. But this may not be the best solution for the problem. If we want to make both android and banking system secure we must know about the security of our mobile banking application. We recommend the android users check the application permission before installing. The banking authority should build more secured application for its users. For ensuring the security of mobile banking application our proposed method can be used.

5.4 Implication for Further Study

To get the more sustainable and optimal solution for the keylogger attack, this research work can be extended to many extents. In the future, the keyboard design can be upgraded for mobile baking applications especially. There can be one special keyboard framework for android mobile banking applications so that users should not worried about their pin and password security. Mobile banking application permission can be reduced, this helps the application get secured from any type of intent vulnerabilities attack. An authentication system for the mobile banking application user can be upgraded for more reliabilities.

REFERENCES

[1] The register [online]. Available at: <<https://www.theregister.co.uk/2017/08/02/banking_android_malware_in_uk>> last accessed on 30-10-2019 at 11:55 AM.

[2] Help Net Security [online]. Available at: <<https://www.helpnetsecurity.com/2018/02/14/financial services-security-investments>>, last accessed on 25-10-19 at 12.48 AM.

[3] Statista [online]. Available at: <<https://www.statista.com/statistics/200855/favourite-smartphone-app-categories-by-share-of-smartphone-users>>, last accessed on 26-10-19 at 11.20 PM.

[4] F. Mohsen and M. Shehab, "Android Keylogging Threat", Proceedings of the 9th IEEE International Conference on Collaborative Computing: Networking, Applications and Worksharing, pp. 545-552, 2013.

[5] A. Kuncoro and B. Kusuma, "Keylogger Is A Hacking Technique That Allows Threatening Information On Mobile Banking User", 2018 3rd International Conference on Information Technology, Information System and Electrical Engineering (ICITISEE), (pp. 141-145). IEEE 2018.

[6] Wikipedia [online]. Available at: << https://en.wikipedia.org/wiki/Mobile_banking>> last accessed on 26-10-19 at 11:41 PM.

[7] Android Developers [online]. Available at: <<https://developer.android.com>> last accessed on 28-10-2019 at 06.03 PM.

[8] T. Wu and Y. Yang, "Detecting Android Inter-App Data Leakage Via Compositional Concolic Walking", Intelligent Automation and Soft Computing, pp. -1--1, 2019.

[9] Ardamax.com [online]. Available at: <<https://www.ardamax.com/keylogger>> last accessed on 29-10-2019 at 12.03 PM.

[10] Bugiel, S., Davi, L., Dmitrienko, A., Fischer, T. and Sadeghi, A.R., 2011. Xmandroid: A new android evolution to mitigate privilege escalation attacks. Technische Universität Darmstadt, Technical Report TR-2011-04.

[11] Technische Univeritat Darmstadt [online]. Available at: <<https://www.informatik.tu-darmstadt.de/systemsecurity/research_sys/projects_sys/previous_projects/mobile_security/ >> last accessed on 27-10-2019 at 11:40 AM.

[12] The Economic Times [online]. Available at: <<https://economictimes.indiatimes.com/industry/banking/finance/banking/online-banking-primer-use-virtual-keypad-for-safe-transaction/articleshow/7512038.cms>> last accessed on 27-10-2019 at 02.47 PM.

[13] Norton Security [online]. Available at: << https://us.norton.com/internetsecurity-wifi-what-is-a-man-in-the-middle-attack.html>> last accessed on 27-10-2019 at 03.10 PM.