Intrusion Detection System using Keystroke Dynamics and Fuzzy Logic For Touch Screen Devices

Submitted in partial fulfillment of the requirements

of the degree of

Bachelor of Engineering

by

Amritha Agarwal (01)

Mansi Patel (49)

Candida Rodrigues (54)

Jincy Sam (57)

Supervisor: Prof. Mahalaxmi Sridhar

UNIVERSITY OF MUMBAI

Intrusion Detection System using Keystroke Dynamics and Fuzzy Logic For Touch Screen Devices

Submitted in partial fulfillment of the requirements

of the degree of

Bachelor of Engineering

by

Amritha Agarwal (01)

Mansi Patel (49)

Candida Rodrigues (54)

Jincy Sam (57)

Supervisor: Prof. Mahalaxmi Sridhar

Department of Information Technology

Don Bosco Institute of Technology
Vidyavihar Station Road, Mumbai - 400070
2016-2017

DON BOSCO INSTITUTE OF TECHNOLOGY

Vidyavihar Station Road, Mumbai - 400070

Department of Information Technology

CERTIFICATE

This is to certify that the project entitled **"Intrusion Detection System using Keystroke Dynamics and Fuzzy Logic For Touch Screen Devices "** is a bonafide work of

Amritha Agarwal	**01**
Mansi Patel	**49**
Candida Rodrigues	**54**
Jincy Sam	**57**

submitted to the University of Mumbai in partial fulfillment of the requirement for the award of the degree of **Undergraduate** in **Bachelor of Information Technology**

Date: / /

(Prof. Mahalaxmi Sridhar)
Supervisor

(Prof. Janhavi B.)
HOD, IT Department

(Dr. Prasanna Nambiar)
Principal

DON BOSCO INSTITUTE OF TECHNOLOGY

Vidyavihar Station Road, Mumbai - 400070

Department of Information Technology

Project Report Approval for B.E.

This project report entitled **"Intrusion Detection System using Keystroke Dynamics and Fuzzy Logic For Touch Screen Devices"** is approved for the degree of **Bachelor of Engineering in Information Technology**

(Examiner's Name and Signature)

1. _____

2. _____

(Supervisor's Name and Signature)

_____

(Chairman)

_____

Date:

Place:

DON BOSCO INSTITUTE OF TECHNOLOGY
Vidyavihar Station Road, Mumbai - 400070

Department of Information Technology

Declaration

I declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea / data / fact / source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Amritha Agarwal (01)	**Mansi Patel (49)**
Candida Rodrigues (54)	**Jincy Sam (57)**

Date:

ABSTRACT

Passwords play a crucial role in many mechanisms of user authentication and security. They provide the first line of defence for applications or data stored in devices ranging from computers to mobile phones. Due to tremendous increase in the usage of mobile phones today, people tend to store more and more sensitive data on their mobile devices. Passwords are shared by people for various purposes, knowingly as well as, unknowingly.Also, passwords can be predicted easily by sophisticated methods by attackers for the sake of intrusion. Therefore, it is essential to enhance the security of password based authentication.

In this application, we make use of the users' keystrokes to form a unique signature, which cannot be easily replicated by intruders, thus providing a more stringent form of security. Hence, even if an intruder gains access to the PIN or password, he will not be able to access the system without the users' keystroke biometric features.

Keywords: Biometrics, keystroke dynamics, Smart phones, Android, Fuzzy Logic, Security.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Problem Statement

To create an intrusion detection application that uses keystroke dynamics and fuzzy logic to provide additional security in touchscreen smart phones.

1.2 Scope of the Project

- The project is an Android application that compares the typing biometrics of the user entering the right password with the one that is pre-stored and unlocks the phone only if there is a greater than 90 percent match.
- It is observed that majority of the smart phones today have capacitive touch screens. Thus the application is created for smart phones with capacitive touchscreens only.

Phone Name and Brand	 Spice Flo Rainbow m-6111	 Karbonn Titanium Machfive	 Samsung Galaxy J7	 Iphone 6s plus
Price	Rs. 1,735 /-	Rs. 7, 690 /-	Rs. 15, 990 /-	Rs. 49999.49 /- ($749)
Touchscreen type	Capacitive	Capacitive	Capacitive	Capacitive

Figure 1.1: Worldwide smart phone OS market share

- Fixed phone orientation : Portrait

Feature Name	Description
Key Hold Time	Time between key press and release
Down-Down Time	Time between consecutive key presses
Up-Down Time	Time between key release and next key
Area of Screen Touched	Area of screen touched at the moment
Average Key Hold Time	Average of key hold times
Average Down-Down Time	Average of down-down times
Average Up-Down Time	Average of up-down times
Average Area of Screen Touched	Average of areas of screen touched

Table 1.1: Feature Sets[7]

1.3 Current Scenario

- Pattern, PIN or password lock

  This is an inbuilt system in almost all smart phones. It allows the user to set the password, PIN or pattern lock but it does not provide security if the intruder knows the password/PIN/patten. This is the most basic form of security.

- Anti theft using alarm

  This security application is available on playstore and is free with in-app purchases. Again this app does not provide security if the user knows the password/PIN/pattern. Also the usage of this app causes disturbance in the surrounding due to noise.

- Lock using face capture

  This application is available on playstore and is free with in-app purchases. This application helps us know the intruder who enters the wrong password/ pattern/PIN by capturing the face of the intruder without his knowledge and the image captured is sent to the registered email. Again this application does not provide security if the intruder knows the password/ pattern/ PIN.

- Lock using fingerprint

  This is an in-built application available in the some smart phones. It requires additional hardware(sensor). It first captures the fingerprint of the owner and stores in the database and every time it unlocks if the fingerprint matches the one that is stored. also if fingerprint does not match then

the user is asked to enter a password/PIN/pattern to unlock the phone where again if the intruder knows the same then he can crack the security provided. But this system has a couple of disadvantages. If the sensor is oily/greasy then the sensor cannot sense the fingerprint as a result the screen prompts for the password/PIN/pattern. Also the finger has to be placed exactly in the same position.

1.4 Need for the Proposed System

There is a need for a simple, low-cost and unobtrusive device. Most frequently used Password based authentication is not fool proof because most passwords are created 'easy to remember', hence easy to crack.
Therefore additional mechanisms are needed to enhance its security without he use of additional hardware. One such complimentary method is to use typing pattern of the user known as Keystroke dynamics

Chapter 2

Review of Literature

2.1 Summary of the investigation in the published papers

- Implementation of Fuzzy logic in web applications[1]
- Uses Pi membership function as a classifier, for efficient output[1]
- Studied keystroke dynamics on touchscreens with Symbian OS[2]
- Compares efficiency of various classifiers viz. BPNN, RBFN, Kstar, J48 and Naïve Bayes[2]
- Demonstrated experimentally that touchscreen based features improve keystroke dynamics based identification and verification.[3]
- Identification measurements were performed using several machine learning algorithms, out of which the best performers were Random forests, Bayesian nets and SVM in this order[3]
- A survey of all existing papers on keystroke dynamics for authentication, on different platforms and the decision making techniques/classifiers used.[4]
- Experimented using neural networks.[4]

2.2 Comparison between the tools / methods / algorithms

No of trials	200 Records													
	Sigmoid MF		LR MF		Bell MF		Trapezoidal MF		Triangular MF		Gaussian MF		PI MF	
	FRR	FAR	FRR	FAR	FRR	FAR	FRR	FAR	FRR	FAR	FRR	FAR	FRR	FAR
50	0.20	0.67	0.00	0.47	0.02	0.00	0.13	0.20	1.00	0.00	0.00	0.02	0.02	0.24
70	0.13	0.40	0.00	0.80	0.01	0.00	0.13	0.20	1.00	0.00	0.00	0.01	0.01	0.03
80	0.27	0.20	0.00	0.93	0.025	0.00	0.20	0.07	1.00	0.00	0.13	0.025	0.01	0.02
100	0.33	0.53	0.00	1.00	0.02	0.00	0.07	0.13	0.93	0.00	0.01	0.02	0.00	0.05

Table 2.1: Calculated efficiency in form of FRR and FAR[5]

No	Membership functions	Results	
		FAR	FRR
1	Triangular	0.00	0.93
2	Trapezoidal	0.13	0.07
3	Gaussian	0.02	0.01
4	Bell	0.00	0.02
5	Sigmoid	0.53	0.33
6	Left-Right	1.00	0.00
7	Pi	0.05	0.00

Table 2.2: Membership functions and their acceptance and rejection rates[5]

Triangular MF gives a linear relationship between values and their membership with only one value having complete membership, which is straightforward but not applicable in real life scenarios. Trapezoidal is a variation of triangular, having a range of values with complete membership to the fuzzy set. It is applicable for simple programs, but linear membership is again a drawback. Even if the FAR of PI is higher than Bell and Gaussian FRR is zero, PI MF is better because pattern detection doesn't alter much even if the hardware is changed.[5]

2.3 Algorithm with example

- Step 1 : Calculate the overall signature with
  - Average time of overall 200 samples
  - Standard deviation Of overall 200 samples
- Step 2 : Calculate

$$a = Avgtime–2 * stddevtime \tag{2.1}$$

$$b = Avgtime–stddevtime \tag{2.2}$$

$$c = Avgtime + stddevtime \tag{2.3}$$

$$d = Avgtime + 2 * stddevtime \tag{2.4}$$

- Step 3 : Now the current time will be checked to see where it lies between the above 4 values a,b,c,d
  - A) If (current time $<$ a OR current time $>$ d) then ;

$$answer = 0$$

  - B) If (current time >= b OR current time <= c) then ;

$$answer = 1$$

  - C) If (current time >= a AND current time <(a+b)/2 then ;

$$answer = 2 * \left(\frac{currenttime–a}{b-a}\right)^2 \tag{2.5}$$

  - D) If (current time >(a+b)/2 AND current time $<$ b) then ;

$$answer = 1 - 2 * \left(\frac{currenttime–b}{b-a}\right)^2 \tag{2.6}$$

  - E) If (current time $>$ c AND current time <= (c+d)/2 then ;

$$answer = 1 - 2 * \left(\frac{currenttime - c}{d-c}\right)^2 \tag{2.7}$$

  - F) If (current time $>$ (c+d)/2 AND current time <= d) then ;

$$answer = 2 * \left(\frac{currenttime–d}{d-c}\right)^2 \tag{2.8}$$

- Step 4 : Repeat steps 3 A) to 3 F) for
  - keydown-keyup time
  - time between keys
  - overall time
- Step 5 : Calculate area of screen touched and average area of screen touched
- Step 6 :
  If answer of time between keys>=0.5 AND answer of time between keys <=1

    If answer avg>=0.5 AND answer avg<=1

If answer total time >=0.5 AND answer total time <=1

If answer avg area of screen touched >=0.5 AND

answer avg area of screen touched <=1

then Accept

else Reject

else Reject

else Reject

else Reject

Chapter 3

Analysis and Design

3.1 Methodology / Procedure adopted

Iterative Model :
In this model, development begins by specifying and implementing just part of the software, which can then be reviewed in order to identify further requirements. This process is then repeated, producing a new version of the software for each cycle of the model. Less time is spent on documenting and more time is given for designing. In our project we are building and improving the product step by step. Hence we can track the defects at early stages. Using this model we can get the reliable feedback.

3.2 Analysis

- Background
  In today's fast-changing world, even the best available security is insufficient for the latest vulnerabilities. Rigidity of security is essential as mostly all confidential activities are carried out with a single click. Smartphones are used everyday for various purposes, and many a times, to access confidential /private data. This could be in a secluded environment or a public place. Its possible that one might share their password intentionally or unintentionally, making their system vulnerable to intrusion attacks.

  Thus, these security systems tend to fail, once the intruder knows the password/PIN.

  A System is needed, which provides security at a higher level, which uniquely identifies the user based on his/her biometrics, without the need of additional hardware.Hence, the risk of hacking into Smartphones is

highly reduced.

- Outcomes
  The expected outcomes of our project are to:
  - Build a lightweight application which is flexible and robust on different devices.
  - Provide Successful differentiation between the legitimate user and intruders, based of keystroke dynamics.

- Objectives
  By successfully building our IDS, we aim to achieve the following objectives:
  - Higher level of security, even when intruder knows the password.
  - Better service to the user, without additional hardware requirements and cost.

- Specifications
  Our system is an application built to provide security to applications on Android Smartphones.The system will not occupy much space, making it faster and portable.

3.2.1 Software / System Requirement Specification – IEEE format . . *Attached as Appendix*

3.3 System Architecture / Design

3.3.1 Modules and their description

The block diagram below gives an outline of the operation of the system. After successful registration, when the user logs into the system, it is verified against the database. Based on the correctness of the password and the biometric, the user is either allowed or denied access. If the valid user is denied access, it is termed as False Rejection. False Acceptance is when an imposter is allowed access. On obtaining access to the system the user can either lock or unlock the applications.

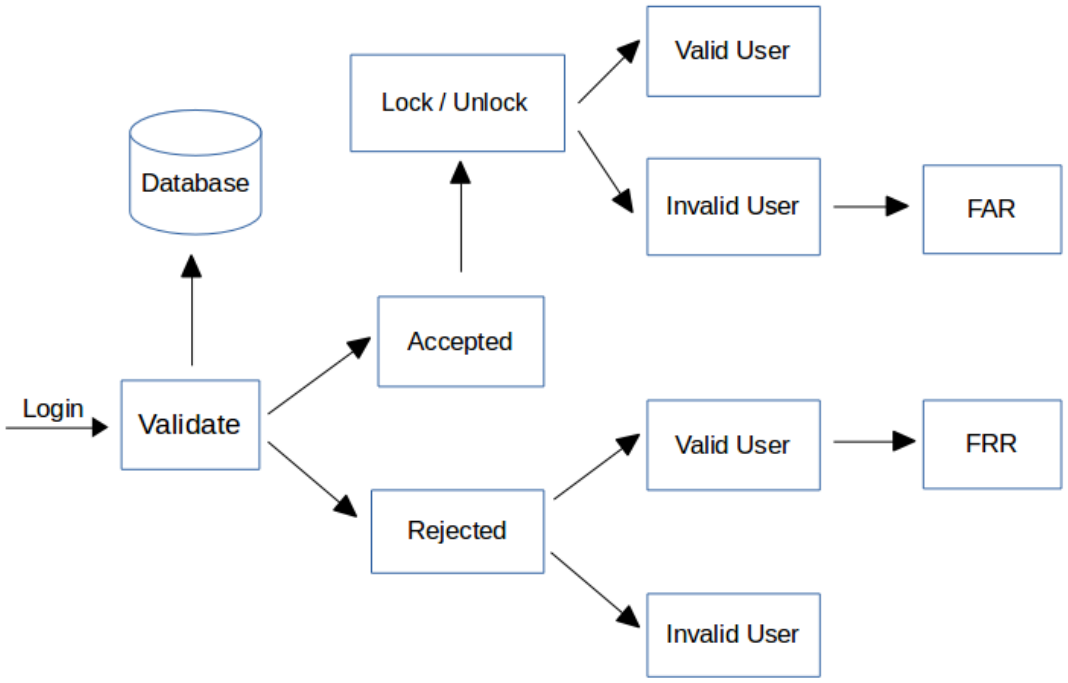

Figure 3.1: Block Diagram of IDS

The different UML diagrams as per the project requirements are as follows:

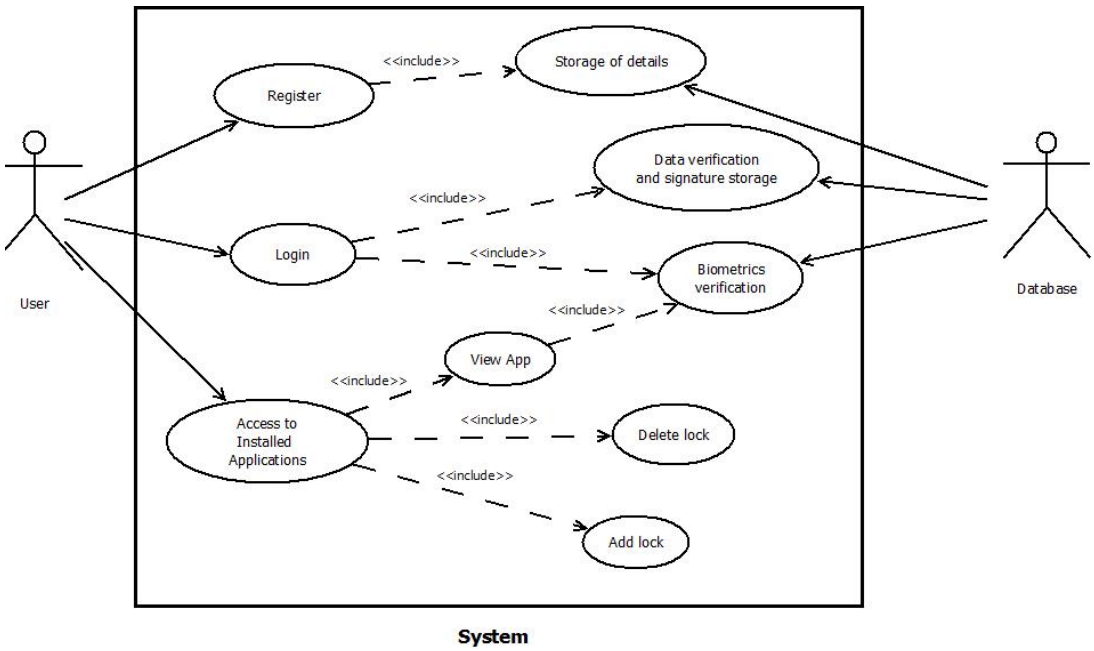

Figure 3.2: Use case diagram

In the above use case diagram, user is an actor who can register, login or access the applications.

1) Registration leads to storage of details in the database.

2) Login lead to data and biometric verification

3) Access to the application allows the user to view the application after verification.

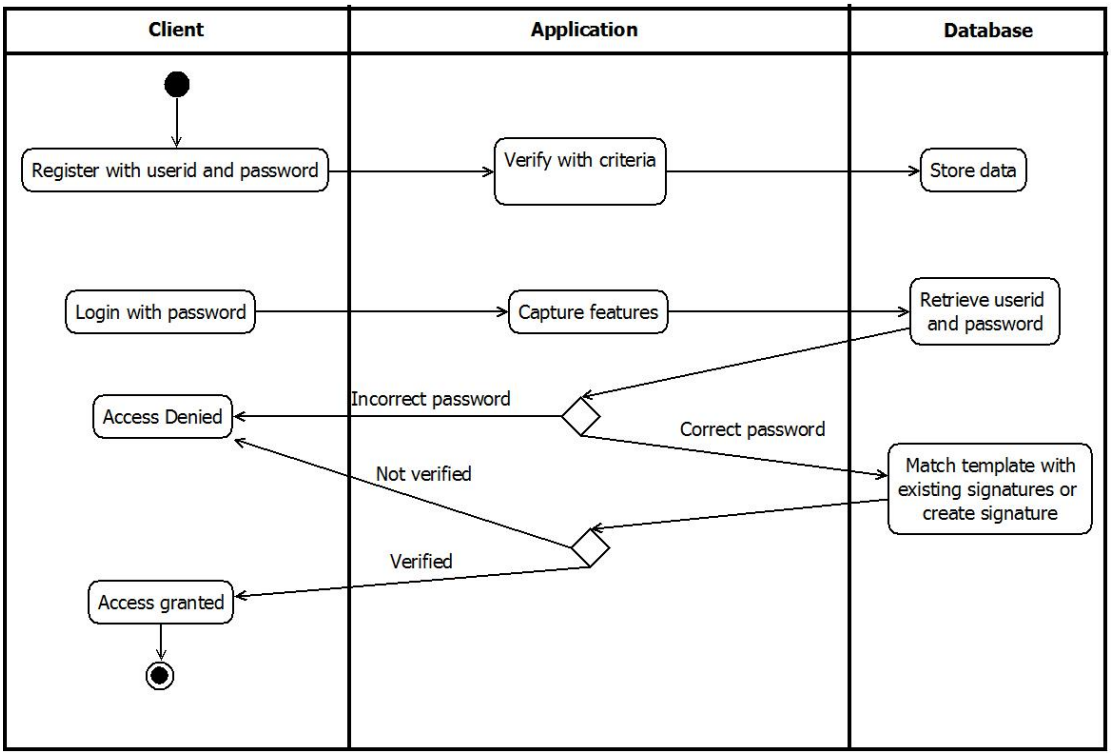

Figure 3.3: Activity diagram

The above diagram represents the various activities performed by the client, application and the database.

1) The username and password given by the user during registration is verified by the application and then stored in the database if it meets the application requirements.

2) When the user logs in with the registered password, the application captures the features. The application then decides to allow or deny access by matching the current template with the user's stored template.

3) If they match, access is granted.

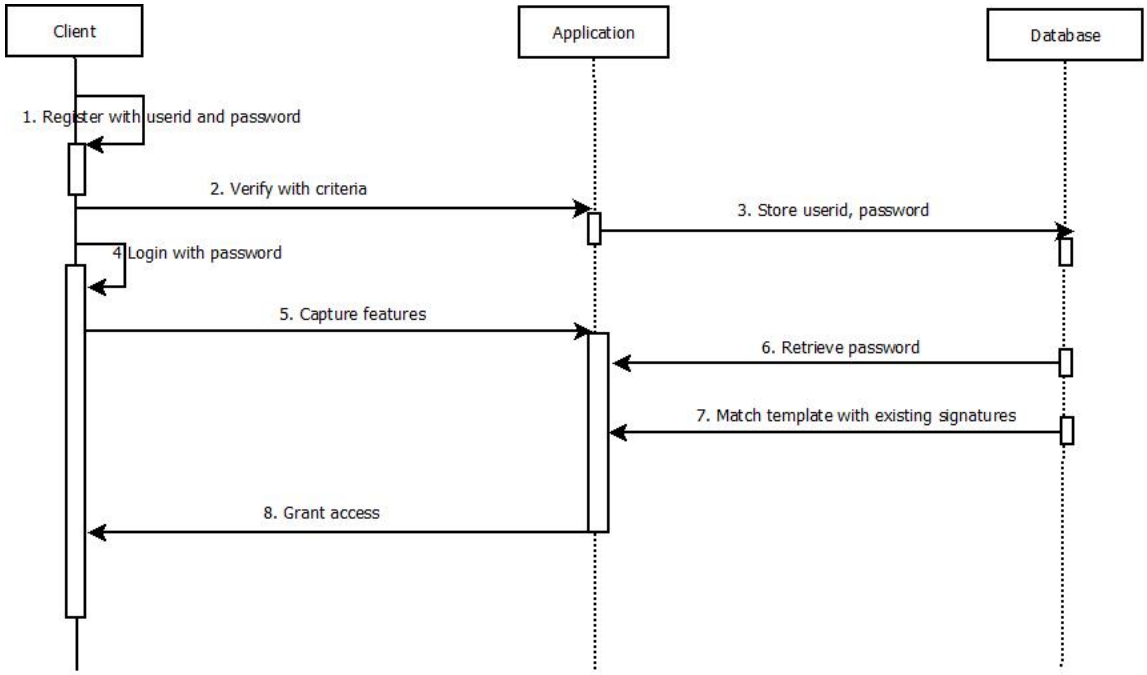

Figure 3.4: Sequence diagram

The above sequence diagram displays the sequence of operations which are explained as follows:

1) The user registers with his username and password.

2) The username and password are verified to check if it meets the requirements.

3) Then the application stores the data in the database in the form of a signature.

4) The user attempts to enter in the system.

5) When the user enters the password, the application captures the biometric features and forms a new signature.

6) The application then retrieves the actual biometrics of the user from the database for comparison.

7) The application then compares the two signatures and generates a result using fuzzy logic.

8) If they match, then the application grants access to the user.

Chapter 4

Implementation

4.1 Implementation Plan *for Sem – 8*

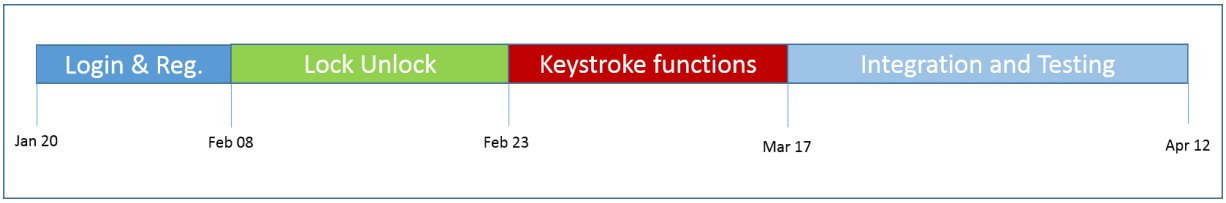

Figure 4.1: Implementation plan for Semester 8

The various sections of the above plan is mentioned as follows:

Sr.no	Activity	Total % work	Person assigned
I]	Login & Registration	10%	Mansi & Amritha
1	Login screen	2.5%	
2	Registration screen	2.5%	
3	Database connectivity	5%	
II]	App listing	20%	Mansi
1	Display list	5%	
2	Lock and Unlock	10%	
3	Database connectivity	5%	
III]	Keystroke measurements	30%	Candida & Jincy
1	Functions	5%	
2	Computations	10%	
3	Storage	5%	
4	Fuzzy logic measurements	10%	
IV]	Integration & Testing	20%	Candida & Jincy
1	Module Integration	5%	
2	Creation of test cases	5%	
3	Testing	10%	
V]	Literature Survey & Documentation	20%	Amritha

Table 4.1: Work distribution

4.2 Testing

4.2.1 Test Cases and Results

TC ID	Test case	Steps	Expected Results	Actual Results	Result
1	To check invalid username and/or password	1) Enter invalid username and/or password 2) Click on Login or Enter	System displays “Invalid login”	System displays “Output: Invalid”	Pass
2	To check valid username and password	1) Enter valid username and password 2) Click on Login or Enter	System displays “Login successful”	System displays “Output: Valid”	Pass
3	To check if passsword doesn’t contain a lowercase character	1) Enter username and password without a lowercase character 2) Click on Register	System displays “Password must contain a lowercase character”	System displays “Password must contain a lowercase character”	Pass
4	To check if passsword doesn’t contain a uppercase character	1) Enter username and password without an uppercase character 2) Click on Register	System displays “Password must contain an uppercase character”	System displays “Password must contain an uppercase character”	Pass

TC ID	Test case	Steps	Expected Results	Actual Results	Result
5	To check if passsword doesn't contain a special character	1) Enter username and password without a special character 2) Click on Register	System displays "Password must contain a special character"	System displays "Password must contain a special character"	Pass
6	To check if passsword doesn't contain a numeric character	1) Enter username and password without a numeric character 2) Click on Register	System displays "Password must contain a numeric character"	System displays "Password must contain a number"	Pass
7	To check if password is greater than 20 characters	1) Enter username and password greater than 20 characters 2) Click on Register	System displays "Password too long"	System displays "Password length should be between 8 to 15 characters"	Pass
8	To check if password is less than 8 characters	1) Enter username and password less than 8 characters 2) Click on Register	System displays "Password too short"	System displays "Password length should be between 8 to 15 characters"	Pass

Table 4.2: Test cases and results

4.2.2 Results of Testing and System Performance

The following are the performance measure for the system:
1) FAR (False Acceptance Rate): Percentage of invalid inputs that were accepted by the system.
2) FRR (False Rejection Rate): Percentage of valid inputs that were rejected by the system.

The system was tested for 50, 100 and 150 samples and it was observed that the FAR and FRR for 50 and 100 samples were high.
When calculated for 50 attempts,
With 150 samples, FAR was 0.15 and FRR was 0.2.
With 200 samples, FAR was 0.12 and FRR was 0.15
Therefore, for 150 samples and above, the results were approximately same. Performance improved with increase in the number of samples.

Chapter 5

Results and Discussion

The application starts with the registration screen as shown in figure 5.1 where the user has to enter his/her email address, a username, password and retype the password in the confirm password field. On clicking the register button, these values are stored in the database. Second is the login screen as shown in figure 5.2 where the user enters the username and password that was created during registration. If these values match with the one stored in the database then the user is allowed access. In order to make it further difficult for the intruder to enter the system, a customised keyboard is implemented which has keys in the A-B-C-D-E sequence rather than QWERTY as shown in figure 5.3. Figure 5.4 shows the output when the system identifies a valid user. The system also displays the feature sets calculated for the user. Figure 5.5 shows the output when an intruder attempts to enter the system with the right password but is not allowed access as his/her biometrics differ from the actual user. Figure 5.6 shows the output when an intruder enters the wrong password.

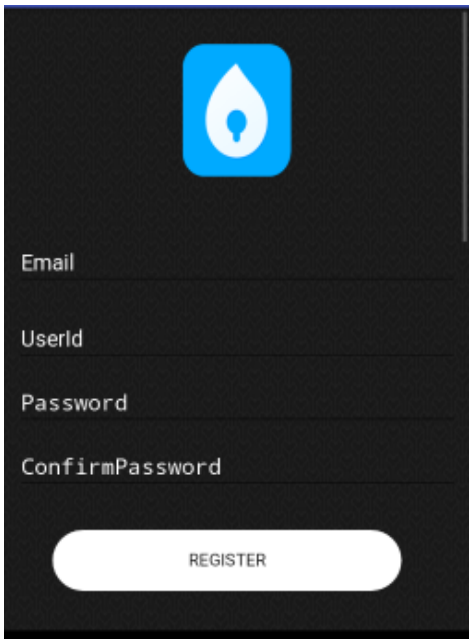

Figure 5.1: Registration Screen

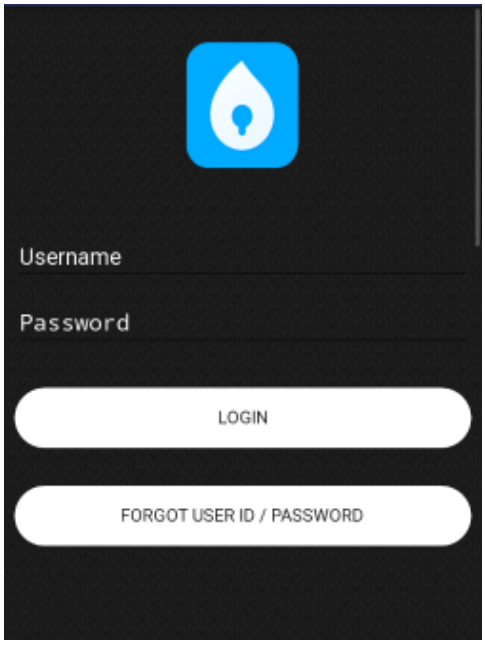

Figure 5.2: Login Screen

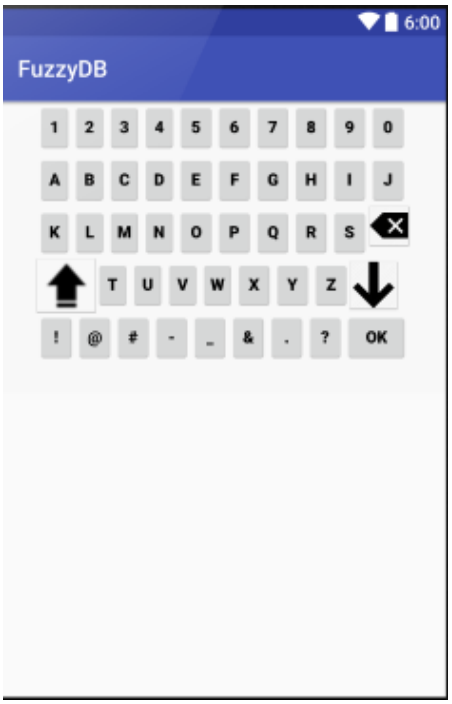

Figure 5.3: The customized keyboard

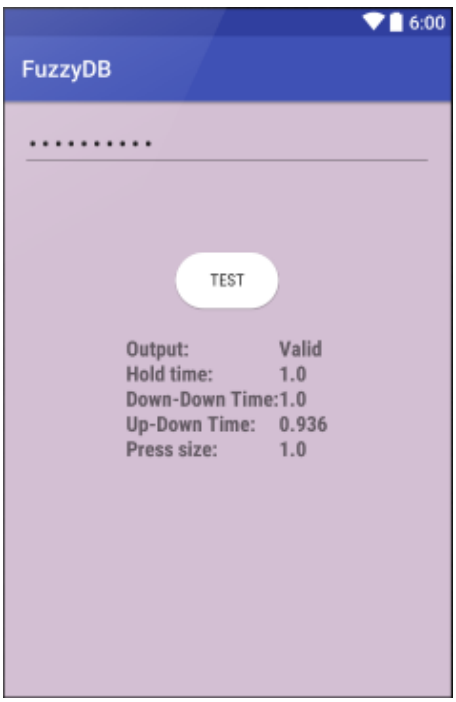

Figure 5.4: Output for valid user

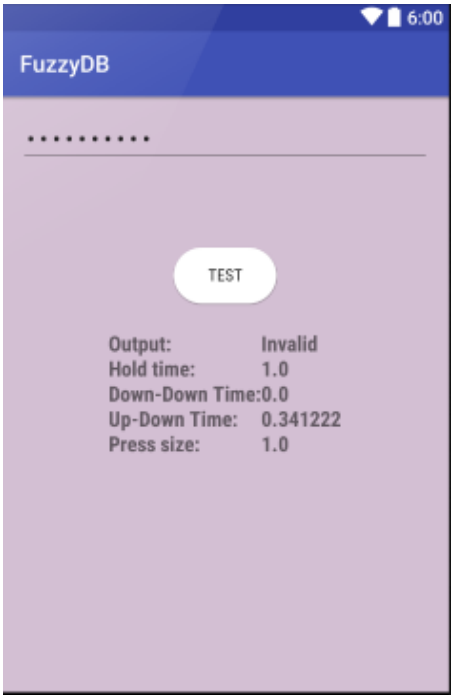

Figure 5.5: Output for invalid user (Wrong keystroke dynamics)

Figure 5.6: Output for invalid password

Chapter 6

Conclusion & Future Work

With this application, an approach is presented to implement keystroke dynamics and fuzzy logic as a system to improve smartphone security. PI membership functions are used for validating a user based on behavioural biometrics as its value of FAR and FRR is low and its efficiency is hardware independent. The signature generated by the user is validated and only after verification the user is able to login.

Keystroke dynamics allows for the design of more robust authentication system than traditional based alternatives alone. It is a leap into the new dimensions of the biometric systems which provides foolproof security without the use of a dedicated hardware device.

In future, a simulated clock can be generated which provides increased efficiency in keystroke dynamics as compared to the usage of system clock.
The security can also be generalized to free text rather than a static password. Keystroke analysis can also be implemented in security of online accounts and systems.

Appendix

System Requirement Specification (SRS)

- Overview
  In today's fast-changing IT world, even the best available security is insufficient for the latest vulnerabilities. Rigidity of security is essential as mostly all confidential activities are carried out with a single click. Our project implements intrusion detection for Smartphones to ensure system security. The concept of keystroke dynamics is used, wherein the users typing biometrics form the identification criteria. Biometrics of authenticated users cannot be duplicated by intruders. A detailed description of this system is provided in the SRS document.

- Target Audience
  The Intrusion detection application targets the common man, as well as enterprise environment and business class, where strict security is essential for day-to-day transactions so as to ensure confidentiality and integrity of data.
  However, it call be implemented by any Smartphone user, if multiple files require strict security.

- Project Team Members:

NAME	**ROLL NO.**
Amritha Agarwal	01
Mansi Patel	49
Candida Rodrigues	54
Jincy Sam	57

1. Introduction
   Smartphones have become a crucial part of modern society. We use it for a variety of purposes ranging from everyday activities to storing and accessing confidential data. Also we might access this data in an unsecure environment such as public places. Since we depend so much on Smartphones to store and process sensitive information, it has become all the more necessary to secure them from intruders.
   For user authentication and identification in phone-based applications, there

is a need for simple, low-cost and unobtrusive device.

Use of biometrics such as face, fingerprints and signature requires additional tools to acquire the biometric which leads to an increase in cost. Use of a behavioral biometric which makes use of the typing pattern of an individual can be obtained using existing systems such as the standard keyboard/touch screen, making it an inexpensive and extremely attractive technique. One of the major advantages of this biometric is that it is non-intrusive and can be applied covertly to existing security systems. The advantage of using behavioral biometrics such as keystroke dynamics is that it can be collected without the knowledge of the user of the phone. Analysis of keystroke dynamics can be broadly classified into two types – static or structured text and dynamic or free text.

Static analysis involves analyzing keystroke behavior of an individual on predetermined phrase(s) at certain points in the system. For example, when logging in a system the user's typing pattern is analyzed when he/she types the user-id and password. It can also involve the use of a particular phrase which is common for all the users of the system. Static text entry can be deployed in systems where there is no scope for further text entry. For example, when a user logs in to check his bank accounts online there is usually no further scope of text entry.

Dynamic analysis involves periodic monitoring of keystroke behavior. It is first checked when a user logs in the system continues thereafter. For example, if a person is browsing the web, certain websites maybe frequented by the user. A list of the commonly occurring websites and the typing behavior of the user while entering the string can be stored. In this case, a training phase would be needed where the user types a particular string several times so that a model can be built for that string. During the test phase, as the user types, the string is recorded along with its timing info which can then used for authentication. However, dynamic monitoring may lead to privacy issues due to its intrusive nature.

We will be making use of static analysis in the form of password.

(a) Product Overview

FuzzKey is an Android application designed to fulfil the purpose of stringent security to confidential apps stored in the users smartphone. The installed applications locked using FuzzKey cannot be accessed

directly but require a password set by the authenticated user. While registering to this application, user biometrics is noted, and identification is done using fuzzy logic. Thus, even if an intruder knows the password, unless the biometrics matches, he will not be granted access to the secured applications.

2. Specific Requirements

   This section deals with the external requirements of our application, for better understanding of the users and to help them interact with the application effectively.

   (a) External Interface Requirements

      i. User Interfaces

         The interface will meet the following requirements to conform to the users' needs:

         A. It will be simple and easy to understand.

         B. All necessary instructions and information will be provided in lucid language.

         Figure 6.2 shows the complete user interface of our application. Since the application is specially designed for Android smartphones, the interface is touchscreen-oriented. Data is entered via the touch screen keypad and user biometrics are analyzed. Each activity in the application is easy to understand and user-friendly. Underlying logic and computation is efficiently encapsulated.

      ii. Hardware Interfaces

         The application is intended to be stand-alone, single user system. The application will run on an Android Smartphone and Android Emulator.

         The application is built for capacitive touch screens.

         No further hardware devices or interfaces will be required.

      iii. Software Interfaces

         The software will run on Android Operating System, version 4.4 (KitKat) and above.

      iv. Communication Protocols

         The application shall communicate with the database and Android software services via API function calls. Because the application

will be written in Java, Java functions will make these calls to the APIs. The exact formats and protocols for incoming and outgoing messages is abstracted by the APIs.

(b) Software Product Features

(a) Authentication

(b) Integrity

(c) It provides the user with a secure and simple registration and login screen.

(d) The interface to lock/unlock screens is user-friendly.

(e) The efficiency of keystrokes properly identifying the legitimate user is reliable, and the capturing features runs in the background without interrupting the user.

3. Software System Attributes

(a) Reliability
FuzzKey application would be highly reliable, as, even if the system fails, it won't affect the applications that were locked using it.

(b) Availability
It stipulates the performance levels required of the system for various kinds of activities. Numerical lower and upper limits set conditions on False Rejection Rate (FRR), False Acceptance Rate (FAR) and Equal Error Rate (EER) etc of the system.
A high FRR may degrade the availability of the system.

(c) Security
The system is highly secure as authentication depends on how the user types the password, and not what the password is. So, even if the password is compensated, the confidential data is not at risk. As security is the most important attribute, the goal of our system is to achieve lowest possible FAR.

(d) Portability
Our application can run on any Android Smartphone, KitKat version (v4.4) or above, and is responsive to various devices currently trending in the market i.e. around 80

4. Database Requirements

   A fast and easily accessible database is required to store user information. Keystroke dynamics requires the storage of large amounts of user biometrics for comparison and verification, so as to ensure the identification of the authenticated user. The system requires a user to register, in order to carry out the necessary operations. At the time of installation of the app the user will be asked to fill in the following information at the least – a user id, a password and an Email ID. Our System makes use of Androids inbuilt SQLite database.

Future Requirements

In future, we plan to generate simulated clock which provides increased efficiency in keystroke dynamics as compared to the usage of system clock. We also intend to generalize the security to free text rather than static password. We also aim to implement keystroke analysis in security of online accounts and systems.

Additional Material

1. References

   The SRS document uses the following documents as references:

   (a) Shahzad-Paper-RAID-2009: Keystroke-based User Identification on Smart Phones

   (b) Future Generation Computer Systems 16 (2000): Keystroke dynamics as a biometric for authentication

2. Appendix

   ER Diagram

   The ER diagram is drawn to have a better understanding of the whole scenario, it was used to conceptualize the phenomena, actions and interactions between various entities and to arrive at the specific requirements in a comprehensive manner. The ER diagram is attached with this SRS.

3. Definition of the terms used

   (a) **False Rejection Rate (FRR)** refers to the percentage ratio between falsely denied genuine users against the total number of genuine users accessing the system. Occasionally known as False Nonmatch Rate (FNMR) or type 1 error. A lower FRR implies less rejection and easier access by genuine user.

(b) **False Acceptance Rate (FAR)** is defined as the percentage ratio between falsely accepted unauthorized users against the total number of imposters accessing the system. Terms such as False Match Rate (FMR) or type 2 error refers to the same meaning. A smaller FAR indicates less imposter accepted.

(c) **Equal Error Rate (EER)** is used to determine the overall accuracy as well as a comparative measurement against other systems. It may be sometimes referred to as Crossover Error Rate (CER).

4. Acronyms and Abbreviations
   IDS- Intrusion Detection System

5. ER Diagram

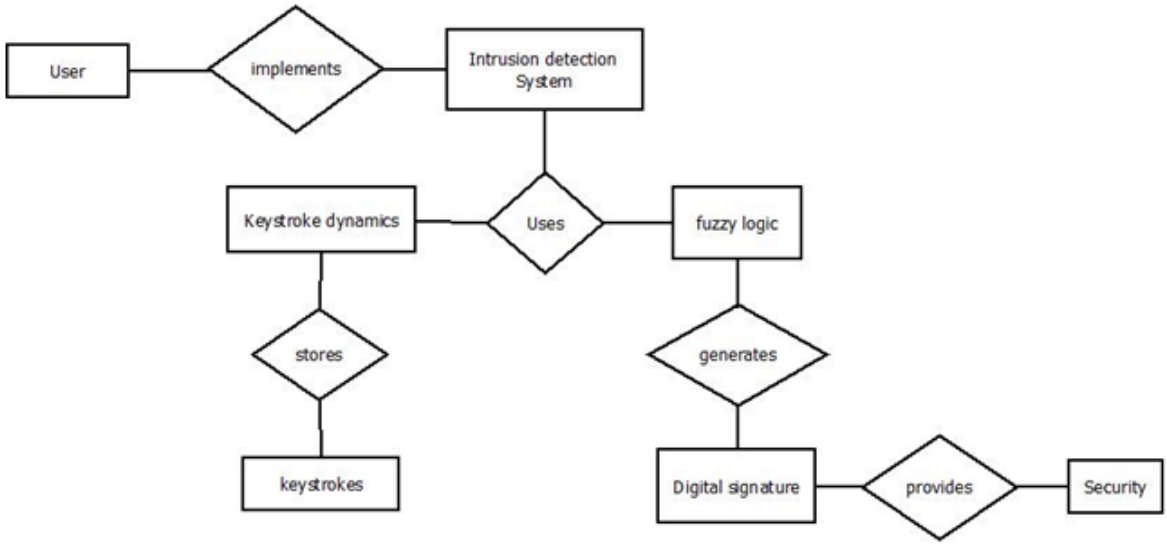

Figure 6.1: ER Diagram of IDS system

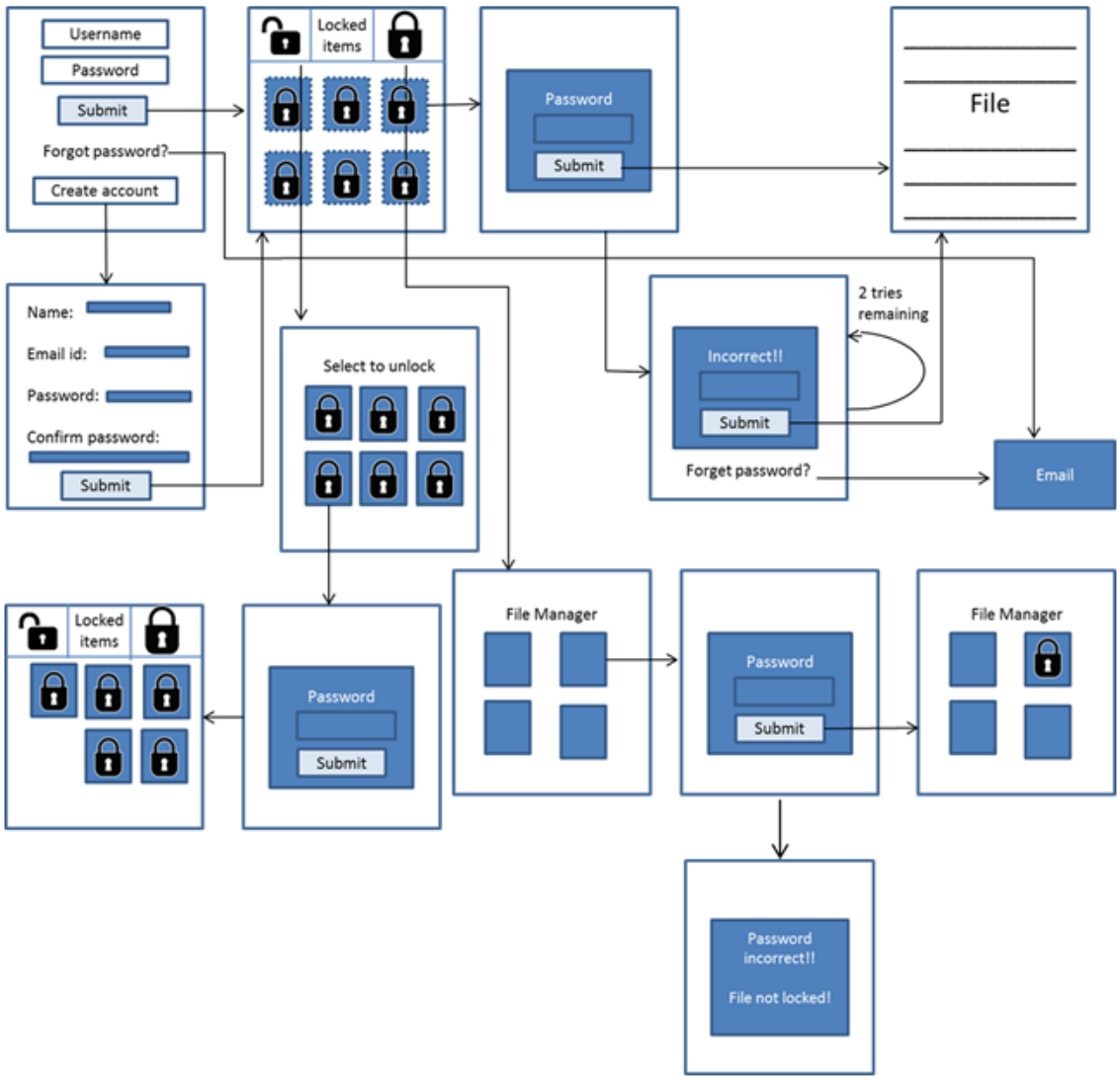

Figure 6.2: GUI of IDS

References

[1] Mahalaxmi Sridhar, Amogh Joshi, Dwarkesh Naik, Udayan Srivastava, Venus Dias *"Intrusion detection using Keystroke Dynamics and Pi membership function for web applications"*, Department of Information Technology, Don Bosco Institute of Technology, Mumbai, India.

[2] *"Keystroke-based User Identification on Smart Phones"* Saira Zahid , Muhammad Shahzad , Syed Ali Khayam , Muddassar Farooq Next Generation Intelligent Networks Research Center (nexGIN RC) National University of Computer and Emerging Sciences (FAST–NUCES) Islamabad 44000, Pakistan School of Electrical Engineering and Computer Sciences (SEECS) National University of Sciences and Technology (NUST) Islamabad 44000, Pakistan saira.zahid, muhammad.shahzad, muddassar.farooq@nexginrc.org,ali.khayam@seecs.edu.pk

[3] 8th International Conference Interdisciplinary in Engineering, INTER-ENG 2014,9-10 October 2014, Tirgu-Mures, Romania. *"Keystroke dynamics on Android Platform"* Margit Antal, László Zsolt Szabó, Izabella László

[4] Baljit Singh Saini, Navdeep Kaur and Kamaljit Singh Bhatia *"Keystroke Dynamics for Mobile Phones:A Survey"* CSE, Sri Guru Granth Sahib World University, Fatehgarh Sahib - 140407,CSE, Lovely Professional University, Phagwara, Punjab, India.CSE, Sri Guru Granth Sahib World University, Fatehgarh Sahib - 140407, Punjab, India.

[5] Mahalaxmi Sridhar, Siddhesh Vaidya, Piyush Yawalkar, Nigel Lobo, Mitali Gawade *" Intrusion Detection Using Keystroke Dynamics and Fuzzy Logic Membership Functions"*

[6] Mahalaxmi Sridhar, Neeraj Yadav, Narendra Vishwakarma, Nikhil Dhavale, Mohit Nijai *"Intrusion Detection Using Keystroke Dynamics on Virtual Keyboard for Web applications"*

[7] Mahalaxmi Sridhar, Alishia D'souza,Johnelle Rebello, Teby Abraham, Winchell D'souza *"Intrusion Detection using Keystroke Dynamics"*

Acknowledgements

A project is a team work which involves the contribution of many people. We would like to thank everyone who have contributed by taking interest in our work and motivating us all the way through. We would like to thank our principal Dr. Prasanna Nambiar and the management for their support and encouragement.

We appreciate the continuous feedback and support of our guide, Mrs. Mahalaxmi Sridhar, which helped throughout our project. We would like to acknowledge the guidance and support of our HOD, Prof. Janhavi Baikerikar, our project coordinator Prof. Tayyabali Sayyad, Prof. Prasad Padalkar and all the faculty members.

Date: