A MULTI-MODEL KEYSTROKE DYNAMICS ANOMALY DETECTOR FOR USER AUTHENTICATION

نموذج متعدد لكشف الاختلاف في ديناميكية الكتابة على المفاتيح للتحقق من هوية المستفيد

By

Sajjad Ali Al-Robayei

Supervisor

Dr. Mudhafar Al-Jarrah

This Thesis is Submitted in Partial Fulfillment of the Requirements for the Master Degree in Computer Science

Faculty of Information Technology

Middle East University

Amman, Jordan

January, 2016

بسم الله الرحمن الرحيم

﴿ اقْرَأْ بِاسْمِ رَبِّكَ الَّذِي خَلَقَ ۝١ خَلَقَ الْإِنْسَانَ مِنْ عَلَقٍ ۝٢ اقْرَأْ وَرَبُّكَ الْأَكْرَمُ ۝٣ الَّذِي عَلَّمَ بِالْقَلَمِ ۝٤ عَلَّمَ الْإِنْسَانَ مَا لَمْ يَعْلَمْ ۝٥ ﴾

صدق الله العظيـــــم

سورة العلق

Authorization Statement

I, Sajjad Ali Abbood Al-Robayei, authorize Middle East University to supply copies of my thesis to libraries, establishments or individuals upon their request, according to the university regulations.

Signature: .....

Date: 9 / 1 / 2016

Middle East University

Examination Committee Decision

This is to certify that the thesis entailed "A Multi-Model Keystroke Dynamics Anomaly Detector for User Authentication" was successfully defended and approved on, 2016.

Examination Committee Members **Signature**

Dr. Mudhafar M.Al-jarrah **(supervisor & Member)**
Assistant professor, Faculty of information technology
Middle East University (MEU)

Dr. Hebah H.O.Nasereddin **(chairman)**
Associate Professor, Faculty of information technology
Middle East University (MEU)

Dr. Khaled Walid Mahmoud **(external member)**
Assistant Professor in CS department
Zarqa University (ZU)

Acknowledgments

First of all I would like to express my gratitude to ALLAH the greatest creator; I have a full faith for that supporting to the all steps in my life.

Also I would say thank to my supervisor, Dr.Mudhafar.M..Al-jarrah, whose expertise, understanding, and patience, added considerably to my graduate experience. I appreciate his vast knowledge and skill in many areas without him I would not have finished this thesis.

I must also acknowledge my life light the great mother without her I can't reach for what I'm on it.

I must also acknowledge my brother Salah and my sisters whom gave me the motivation to make this thesis done.

Appreciation also goes out to all my friends Mohammed.A.Habour, Mohammed.A.Al-Qurayeni and especially Abbas.A.Alhashimi who always gives me the Bravery to do everything that I thought I can't do it, and to all my other friends.

I would also like to thank my friend Dr.Sadiq.H.Al-Zobaidi, who gave me a little bit of light when I went to give up, and to my colleague Abdullah.Al-Maswadeh I appreciate his vast knowledge and skill in programming.

Dedication

This dissertation is lovingly dedicated to my mother, and to my lovely country IRAQ.

Table of Contents

List of Tables

List of Figures

Abbreviations

Abs-Min: Absolutely Minimum Model.

ATM: Automated Teller Machine.

CMU: Carnegie Mellon University

DD: Down-Down.

DTM: Distance to Median.

EER: Equal Error Rate.

EMM: Enhanced median-Median.

FAR: False Alarm Rate.

FMR: False Match Rate.

FN: False Negative.

FNMR: False Non-Match Rate.

FP: False Positive.

FRR: False Rejection Rate.

GA: Genetic algorithm.

H: hold.

KD: Keystroke Dynamics.

LL: Lower Limit.

Medstd: Median-Standard Deviation model

MEU: Middle East University.

MMD: Multi-Model Detector System.

NN: Neural Network.

OTP: One Time Password.

PIN: Personal Identification Number.

PM: Pass Mark.

PP: Press Press.

PR: Press Release.

RP: Release Press.

RR: Release Release.

STD: Standard Deviation.

SVM: Support Vector Machines.

TAR: True Acceptance Rate.

TRR: True Rejection Rate.

TRR: true Rejection rate.

UD: Up-Down.

UL: Upper Limit.

V: Vector.

Abstract

A Multi-Model Keystroke Dynamics Anomaly Detector for User Authentication

By

Sajjad Ali Al-Robayei

Supervisor

Dr. Mudhafar M. Al-Jarrah

January, 2016

The rapid increase in cyber-attacks targeting personal, business and government information assets and the damages resulting from such attacks is emphasizing the need for strengthening defenses of information technology resources. Access control is the first line of defense which includes several authentication methods. To improve access control, several biometric features have been used lately with various degrees of cost and complexity.

The keystroke dynamics is a behavioral biometric that can be part of an access control system; its main advantage is that it does not need extra hardware. This thesis aimed at enhancing the authentication power of the keystroke dynamics method through providing better anomaly detector models. The research adopts an empirical analysis approach in formulating anomaly detector models by examining a major keystroke dynamic benchmark dataset. The thesis presents a multi-model anomaly detector that comprises three statistical models that measure features of the typing rhythm to determine the authenticity of the typist based on a comparison with training templates of genuine users.

The three models use the distance to the median of a feature element to classify it as a genuine or imposter feature. The feature set consists of key-hold, the latency between two keys, and a composite feature of hold and latency. Two of the three models were formulated in this study; these are the Enhanced Med-Med model and the Absolute-Minimum model, and the third is an already published model that uses the standard deviation as a measure of distance to the median. Also, the work involved the development of keystroke dynamics software for data collection during the training phase, and to be used as a dynamic authentication tool during the testing phase. The benchmark dataset was analyzed using the proposed models, and the results showed that the multi-model, the enhanced median-median model and the absolute-minimum models had equal error rates of 0.062, 0.063 and 0.069, whereas the best equal error rate from previous studies of 16 models, using the same dataset, was 0.071.

The analyses included another, more informative, comparison of models' error rates, where the miss-rate of a model is measured at the point of 5% rejection rate of genuine users, which is an acceptable rate of rejection. The miss-rate for the enhanced median-median model was 14.4%, and 20.4% for the absolute-minimum model, while the previously reported best performing model using the same dataset had a miss-rate of 23%. The research was complemented by the collection of a dataset for 20 subjects,

using the developed software tool, in which there were 30 repetitions of training attempts and 30 repetitions for the testing phase. The reason for choosing a relatively small number of repetitions was to make the tool less of a burden on the user during the training phase. An analysis of error rates using the proposed models showed that in spite of a low number of repetitions, the obtained results were close to the results using the more extensive benchmark. The error metrics FAR, FRR and EER for the proposed multi-model is obtained by considering votes of the three models, where a typing attempt is classified as genuine if two models gave it a genuine vote.

The thesis ends with several conclusions and recommendations for future work.

Keywords: Keystroke dynamics, FAR, FRR, EER, multi-model, training phase, testing phase, behavioral biometric.

نموذج متعدد لكشف الاختلاف في ديناميكية الكتابة على المفاتيح للتحقق من هوية المستفيد

إعداد

سجاد علي عبود الربيعي

إشراف

د. مظفر منير الجراح

الخلاصة

إن الزيادة السريعة في الهجمات الالكترونية التي تستهدف المعلومات الشخصية و التجارية وحتى المعلومات الحكومية حيث الاضرار الناجمة عن مثل هذه الهجمات تؤكد على ضرورة تعزيز الدفاع عن مصادر تكنولوجيا المعلومات. تعتبر مراقبة الدخول الى النظام هي خط الدفاع الأول ولتحسين هذه المراقبة هنالك عدة أساليب للوثوق بالمستفيد من الدخول الى النظام، وقد استخدمت العديد من الميزات الحيوية في الآونة الأخيرة مع درجات مختلفة من حيث التكلفة والتعقيد. تعتبر حيوية الكتابة على لوح المفاتيح هي الطريقة التي يمكن أن تكون جزءا من نظام مراقبة الدخول. ميزته الرئيسية هي أنه لا يحتاج إلى أجهزة إضافية.

ان العمل في هذه الأطروحة يهدف الى تحسين من زيادة الاعتماد على طريقة حيوية كتابة لوح المفاتيح من خلال من خلال تقديم نماذج للكشف عن أفضل المميزات والفوارق التي تحدث عند التنقل بين الازرار للكتابة على لوح مفاتيح جهاز الحاسوب. هذا البحث يعتمد على التحليلات التجريبية لتكوين نماذج الكشف عن الفوارق او المميزات في ضربات لوح المفاتيح من خلال دراسة مجموعة كبيرة من البيانات. تقدم هذه الاطروحة نموذج متعددة يضم ثلاثة نماذج إحصائية لقياس ملامح إيقاع الكتابة لتحديد هوية المستفيد من الدخول الى النظام على أساس المقارنة مع ملامح تم التدريب عليها من قبل. هذه النماذج الثلاثة تستخدم (الوسيط) لحساب الفوارق في ازمان الضربات على المفايح لتحديد هوية الشخص على انه الشخص الحقيقي ام المزيف. مجموعة الميزات تتكون من ثلاث انواع بصورة عامة (الفترة الزمنية المستغرقة للنقر على المفتاح الواحد بصورة كاملة، فترة الانتقال من المفتاح الاول الى الثاني ابتداءاً من وقت تحرير المفتاح الاول وانتهاءاً بلحظة الضغط على المفتاح التالي والفترة الثالثة وألأخيرة هي الفترة الزمنية التي تبدأ من لحظة الضغط على المفتاح الاول وتنتهي بلحظة تحرير المفتاح الثاني) تحسب هذه الفترات الزمنية لكل اجزاء كلمة المرور عند كتابتها من قبل المستفيد عندما يدخل الى النظام او عنمدا يتدرب النظام على طريقة كتابته.

اثنان من النماذج الثلاث موضوع البحث اعلاه قد قدمت من قبل البحث حيث تم تحسين اداء عمل نموذج موجود مسبقا (وسيط-الوسيط)، وتم عمل نموذج جديد (اصغر قيمة في الاعداد المطلقة) فحص هذان النموذجان من خلال بيانات كبيرة الحجم ومعتمدة قبل الاخذ بدمجهما مع النموذج الثالث والاخير- الذي تم عمله مسبقا في عام 2012 من قبل (د.مظفر الجراح) المشرف على هذه الرسالة – الى النظام المتعدد موضوع البحث.

كذلك تم عمل اداة لجمع واستخلاص المميزات الخاصة بايقاع الكتابة من خلال مرحلة (التدريب) التي تدرب النظام على كتابة كلمة مرور موحدة من قبل 20 شخص 60 عملية تدريب لكل شخص، استخدمت هذه البيانات في المرحلة الثانية وهي مرحلة (الفحص) التي تستخدم نصف البيانات ال60 المجهزة مسبقا للمقارنة معها وفحص تحديد هوية المستفيد. كما كان هنالك التحليل الذي اعتمد في المقارنة مع نماذج الدراسات السابقة وهو تحليل بيانات جامعة كرنج ميلون الامريكية والذي يحتوي على 20,400 محاولة تدريب جمعت من خلال 51 شخص على ثمان جلسات كل جلسة 50 محاولة للشخص الواحد وتم الحصول على النتائج التالية حيث اعطى النموذج المحسن (وسيط الوسيط) 0.063 (معدل الاخطاء المتساوية) الذي جعل هذا الانموذج على راس قائمة نماذج المقارنة في الاداء وكان النموذج الثاني يحمل الرقم 0.069 بينما كان افضل نموذج في الدراسات السابقة يحمل الرقم 0.071 ، كان اداء النظام المتعدد من بعد دمج النماذج قد حصل على قيمة 0.062 من الاخطاء المتساوية حيث كان حتى افضل من النماذج عندما عملت بصورة منفردة.

تم عمل نوع آخر من التحليلات يهدف الى قياس معدل الانذار الخاطء عن تثبيت معدل القبول الخاطء الى 5% حيث حصل النموذج الاول على 14.4% والثاني على 20.4% بينما كان قد حصل افضل النماذج الدراسات السابقة على 23%

في نهاية هذه الاطروحة العديد من الاستنتاجات، التوصيات والاعمال التي يمكن ان تنجز في المستقبل.

الكلمات المفتاحية: حيوية الكتابة على المفاتيح، معدل القبول الخاطئ، معدل الانذار الخاطئ، مميزات ايقاع الكتابة، ملامح ايقاع الكتابة.

Chapter 1

Introduction

1.1 Overview

The demand for more secure methods of access control to protect computer resources is increasing exponentially due to the rapid rise in cybercrimes. The traditional password method is no longer a solid defense as passwords can be easily compromised. The field of biometrics based authentication is gradually becoming an essential part of access control to information systems, computers, and networks.

User authentication based on the typing profile of a user, using the keystroke dynamics (KD) method, is one of the behavioral biometrics that requires no additional hardware and can be easily implemented.

An authentication system can rely on a multi or single modality of features in the verification of user identity. For example, an automated teller machine (ATM) machine user needs to provide two types of evidence to be allowed to withdraw money; these are the credit card and the personal identification number (PIN) code. In such a case two categories of authentication factors are combined, the PIN code which is in the category of something you know, and the credit card which is of the something you have category. The typing profile of a person belongs to a third category of factors which is the something you are.

A multi modal system aims to strengthen systems security by combining features from several categories, like a combination of fingerprint or iris, both methods belong to the physiological authentication category, with PIN or password, which is something you know, and behavioral features such as typing rhythm.

The KD method is a behavioral measurement approach which requires the collection of keystroke timing data over a training session, in order to determine the typing signature of a user, referred to as the template, to be used during authentication session. The

training session is similar in objective to the process of making several signatures of the customer during the process of opening a bank account.

KD attempts to build a typing profile of a user through a learning process in which features of the user typing behavior are recorded in a database for later use in the authentication process. Unfortunately the KD approach still suffers from false positive (FP) and false negative (FN) errors.

In order to enhance the acceptance of geniuses users and the rejection of impostors users, in the log-in attempts, better anomaly detector models are needed. Apart from formulating new models, a composite multi-model authentication system based on individual anomaly detector models is an approach that can improve the anomaly detection power and thereby reduce authentication error rates. A multi model is developed in this thesis that uses several models that have lower error rates compared to other models. The first model is an enhancement on a published med-med model, the second model is a new minimum of absolution model formulated in this work, and the third an existing med-std model.

1.2 Background on KD

KD is a method of analyzing the way a user types on a keyboard and classify him based on his regular typing rhythm according to the model features that used to detect the time of events accords on the keyboard. It is the study of people who can be identified by their typing rhythms much like handwriting/signatures/walk style is used to detection of a text written. User's typing pattern is unique because of the neuro-physiological factors that also make written signatures unique and all of it in biometrical field. KD as biometrics characteristics is not a new one. (Roy, S., Roy, U., & Sinha, D. D., 2014).

1.2.1 Keystroke Dynamics Authentication

KD was formally investigated first time by Bryan and Harter in 1897 as part of a study on skill gaining in the telegraph operators. In 1975 (Spillane, R., 1975) suggested in an IBM technical to identify a user at a computer keyboard by bulletin that typing rhythms. That bulletin described KD in concept (Forsen, G. E., Nelson, M. R., & Staron Jr, R. J., 1977) conducted preliminary tests of whether KDs could be used to distinguish typists. (Gaines, R. S., Lisowski, W., Press, S. J., & Shapiro, N., 1980) in 1980 produced an extensive report of their investigation with seven users into KD. After then (Bleha, S. A., Knopp, J., & Obaidat, M. S., 1992) submitted his PhD thesis on Recognition system based on KD. (Joyce, R., & Gupta, G., 1990) prepared an identity authentication based on keystroke latencies in (Monrose, F., & Rubin, A., 1997).

KD is the expression given to the procedure of measuring and evaluation a user's typing style. These measures, based largely on the timing latencies between keystrokes hitting's to generate that user typing style to matched with a user profile that already taken before at the training part, procedure a match or not can be used to decisional the user will access or not to the system (Singh, K., & Kaur, H., 2013)

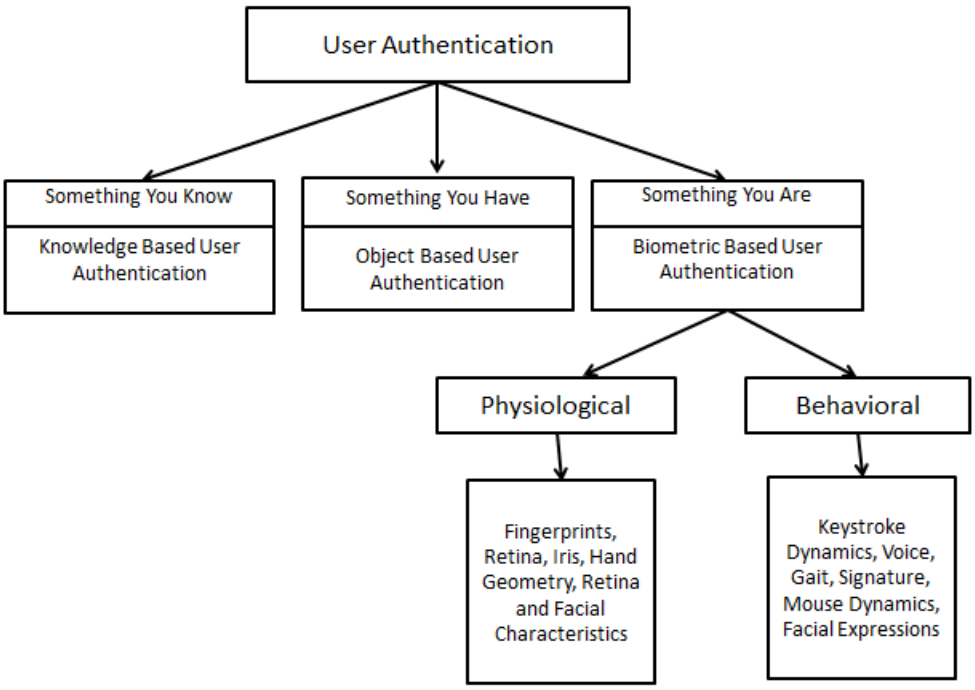

Figure1.1: User Authentication Topology (Singh, K., & Kaur, H., 2013)

Figure1.1 shows the topology of user authentication clarifies all authentication types in general, it focus on the biometric based authentication types with giving some examples of its implementation ways.

1.2.2 Keystroke Dynamics Types

KD verifications techniques can be classified as two types either static or continuous. Static verification are static approaches analyze keystroke verification characteristics only at specific times, that discusses by this research. Static approaches provide more

robust user verification than simple passwords, but do not provide continuous security; they cannot detect a substitution of the user after the initial verification. Continuous verification, on the contrary, monitors the user's typing behavior throughout the course of the interaction. KD can be described by several features which are extracted from the typing rhythm of the user. These features are extracted from data which are recorded by the event recording module.

1.2.3 KD Evaluation Metrics

There are many metrics to measure the decision performance for KD authentication system, the popular metrics are the

1. FAR: False Acceptance Rate: the impostors' acceptance. Also known miss rate
2. FRR: False rejection Rate: the Geniuses' rejection. Also known false alarm.
3. EER: The equal point of FAR and FRR as shown in the Figure 1.2

On other hand, FRR is the number of FN test and FAR the number of FP divided by the number of samples used to test for the type of error, often the standard metric for evaluating biometric systems is equal error rate the values of FAR and FRR are equal. (Sedenka, J., Balagani, K. S., Phoha, V., & Gasti, P., 2014) (Abernethy, M., & Rai, S., 2012).

This research uses an extended metrics TAR and TRR, which are the inverse of FAR and FRR.

TAR: True Acceptance Rate of genuine user, based on testing same user data.

TRR: True Rejection Rate of Impostor, based on testing another user data.

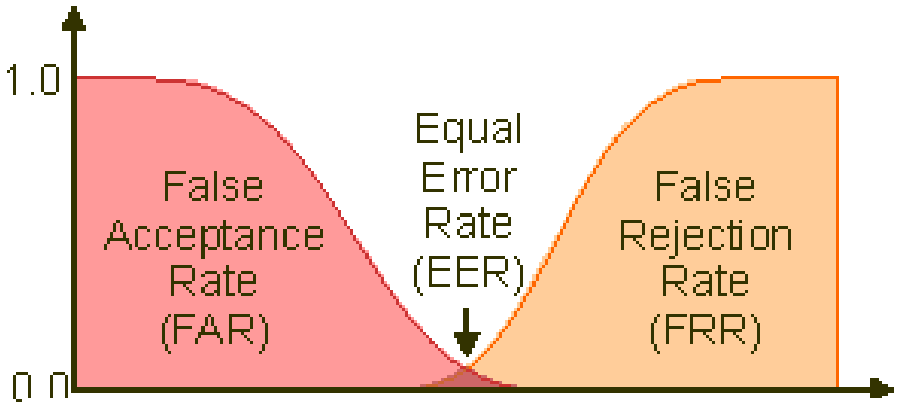

Figure1.2: Equal error rate (e_bias_detail.php?BiasID=3, 2016)

1.2.4 Feature Set of Research

Every anomaly model has a numbers of features, this numbers are deferent from model to another depending on the study and the input method ex: in a study created by (Antal, M., & Szabó, L. Z., 2015) they tested their system on 42 users using two type of android mobiles. Each user types a specific password, they extracted 41 purely touch keystroke features and 71 features mixed of keystroke and touch features.

In another studies such as (Idrus, S. Z., 2015) (Giot, R., El-Abed, M., & Rosenberger, C., 2009) it introduced the data consisting of five different features or timing vectors PP, RR, PR, RP and V

a) ppTime (PP) : the latency between pressing $key_{(i)}$ and $key_{(i+1)}$

b) rrTime (RR) : the latency between releasing $key_{(i)}$ and $key_{(i+1)}$

c) rpTime (RP) : the latency between releasing $key_{(i)}$ and pressing $key_{(i+1)}$

d) prTime (PR) : the duration of hold press on one key.

e) Vector (V): the concatenation of the previous four timing values.

As same features concept of those studies, this research used the down and up instead of press and release and (H) as a hold duration for each key pressing instead of (PR).

The proposed models in this research used three types of features according to compare the result with past models that used (H, DD, and UD).

Extracting features started at the first key down ending with last key up in password typing, features will stored individually in the database as (Hold, DD, UD) to calculate and build the template.

Numbers of features according to the Carnegie Mellon University (CMU) benchmark was 31 features regarding to the password length that used 10 characters regarding to the password that used (.tie5Roanl).

10 Down-down from first key to enter key.

10 up-down from first key to enter key.

10 Hold for password + 1 for enter.

Table 1.1 shows sample of the published CMU benchmark with the following details:

a) The column "subject" refers to number of subject in CMU benchmark.
b) The column "sessionindex" refer to number of session.
c) The column "rep" refers to the number of iteration in the session.
d) The column "H.period" refers to the hold duration of first character in password (.)
e) The column "DD.period.t" refers to the latency between down (.) to down (t).
f) The column "UD.period.t" refers to the latency between (.) up to (t) down.

Table 1.1: Sample of CMU benchmark password (.tie5roanl)

	A	B	C	D	E	F	G	H	I
1	subject	sessionIndex	rep	H.period	DD.period.t	UD.period.t	H.t	DD.t.i	UD.t.i
2	1	1	1	0.1491	0.3979	0.2488	0.1069	0.1674	0.0605
3	1	1	2	0.1111	0.3451	0.234	0.0694	0.1283	0.0589
4	1	1	3	0.1328	0.2072	0.0744	0.0731	0.1291	0.056
5	1	1	4	0.1291	0.2515	0.1224	0.1059	0.2495	0.1436
6	1	1	5	0.1249	0.2317	0.1068	0.0895	0.1676	0.0781
7	1	1	6	0.1394	0.2343	0.0949	0.0813	0.1299	0.0486
8	1	1	7	0.1064	0.2069	0.1005	0.0866	0.1368	0.0502
9	1	1	8	0.0929	0.181	0.0881	0.0818	0.1378	0.056
10	1	1	9	0.0966	0.1797	0.0831	0.0771	0.1296	0.0525
11	1	1	10	0.1093	0.1807	0.0714	0.0731	0.1457	0.0726

1.2.5 User Distinction through the Typing Rhythm

Many models have been proposed during that time. Models based on traditional statistics such as mean times and their standard deviations STD are common. Over the years, different pattern recognition models have come into vogue and been applied to KD.

The anomaly detector model distinction the user rhythm after converts the strokes to time and extract the features that depended on it according to its algorithm an initially observation can produces as the following:

A. Training a number of repetitions for user X to extract a threshold called template consist of two limits, upper and lower, used these limits to later to test the login attempt with it.

B. Testing a number of repetitions belong to the same user X with his two limits to calculate the false rejection if the anomaly detector model reject some of these login attempts.
C. Testing a number of repetitions belong to another users/user with user X limits to calculate the true acceptance if the anomaly detector model accept some of these login attempts.

1.2.6 Data set Benchmark

- **CMU benchmark**

Research used the public benchmark that collected by CMU, it contained of 20,400 dataset rows came from 51 subjects 50 repetitions for each collected in 8 sessions to generate 400 rows for each subject. This benchmark chooses because it published, available and large of data that make the criteria of the comparisons with the past studies more accurate.

- **Middle East University benchmark**

To analyze some of proposed models errors, another dataset will use that collected according this research analyses by the Middle East University (MEU).

1.2.7 Data Collection

This research aimed to collect MEU dataset which contains of 1200 rows of data by 20 subjects' 60 repetitions for each subject separated into two sessions. This is a minimum number compared with the main benchmark used to evaluate the methods in this research because of the time span for the data collection.

This research not need to record the metadata subject's characteristics such age, gender and handedness (right hand, left hand and use both hand in typing on keyboard) because of the research is static KD and the metadata important in the continues KD researches.

The big issue for the data collection that password mistyping, the KD very sensitive because it depending on the key press/release time.

Data collection operations can be as the following 5 steps:

Step1: Enter username of subject with considering the session number.

Step2: Enter the sample of password that will be (.tei5roanl).

Step3: Enter the same password for 30 times.

Step4: If has any password mistype this repetition will cancel because it not matches with step2 and the subject will inform by the repetitions count down number.

Step5: The subject will confirm if all repetitions well done.

Keyboard that will be uses in data collection is the ordinary QWERTY keyboard for (Lenovo laptop yoga 13) and the password length is 10 characters to extract the same features set numbers of the public CMU benchmark to make a fair evaluation and better to understanding the result that will outcome after.

Data collected used to convert the keystrokes to time the stroke real time method, which that mean stamping for each time events when password are type start with first key down end with last key up, instead of the method that used by past studies such as med-med anomaly detector model for (AL-Rahmani, 2014) where used the time span method

to calculate the key down and key up time by calculating the difference between the time events and the midnight, January 1 1970 UTC.

1.3 Problem Statement

Recent researches in KD shows an increasing interest in this method as a viable behavioral authentication measure. In addition, the KD method has the advantage of hardware independence as it requires no special equipment. The proposed research work addresses the problem of improving imposter detection power through considering alternative models of anomaly detection in typing rhythms.

The main issues of this research is to enhance an existing model, try to creating new detector model and to improve the KD anomaly detection through combining more than one model in a Multi-Model anomaly detector to investigate that combining better than work individually or not.

1.4 Goal and Objectives

The main goal of this thesis is to increasing the dependability of the KD biometric authentication system.

The research objectives are

1- Increasing the ability for the anomaly detector models to reach the best equal error rate by enhancing an existing model and creating new model.
2- Enhancing the final decision on the login attempts by building Multi-Model detector uses three of best models according to its equal error rate.

1.5 Significance of Work

The authentication that conducted with (something you are) determined to the system exclusive acceptance to check if the behavior of login attempt authorized access to that information area or not.

Because of increasing in dependability on the computer or any device aimed to store and process the information the authentication technique must be supported against the cybercrimes that trying to theft the user identity and/or password. user consider this issue on a top thing thinking about it when creating new authentication account and the other side the service providers always investigated new solution against this warning for example the security question and the alternative email account when the user forget or stolen the password

The main goal for the authentication is to provide insurance that only who have the password can access to that information area. Thus, authentication will down if anyone knew that password because it (something you have). Now the revolution of the internet of thing IOT make almost the thing connect together by the cloud with one account, one user ID and password connected many served things, so if anyone knew the password then he can get and control to part of things, hence the most important thing is how to protect our login to the personal/privacy area without give the hacker a little bit of chance without extra devices.

The biometric based-authentication have very significant reasons to be the most robust way to protect a system from any impostor login attempt and without more devices required like the other biometric authentications techniques (iris, walk style, finger print, etc...) biometrical authentication is just a software!

The other reason to make it on a top of significance authentications is the password can be stolen by a "Trojan" or any spyware app but without any benefit for the hacker because the password like the human signature it's difficult to copy, it depended on the human behavioral and every human has a unique rhythm than others.

1.6 Thesis Outlines

This thesis consists of five chapters organized as the following:

Chapter one: Introduce the thesis, background on KD, problem statement, goal and objective and significance of work.

Chapter two: Literature review and related Work

Chapter three: Anomaly detector models and the Multi-Model system.

Chapter four: Data collection result and discussion.

Chapter five: Conclusions and future work.

Chapter 2
Literature Review and Classifiers

2.1 Introduction

Enhancements to low cost password based authentication which provides an additional level of trustworthiness are more appealing against the other authentication metrics such as typically username and password even than the biometric authentication that required an extra device like finger print or Iris scanner.

KD evaluates the typing behavior of the typist by calculates the duration of each key press, latency between neighbors key presses. These time periods are called the hold and delay times, respectively. Hold times will always exhibit positive values as a finite amount of time is required to press a key, while delay times may be positive or negative. A negative time delay occurs when a user presses the succeeding key prior to releasing the current key, negative values are popular for the typist who have blinding fast write on the keyboard. (Syed, Z. A., 2014)

In generally KD in this research had two major steps training and testing. In the first step the KD need to determined who are the typist started with extracting features during password typing N times to generate and store a template that abstracted these n time repetitions. Second step is to test a login attempt with that template to measure if it will get false reject or true accept and to measure the efficiency of a model this step should repeated many times to measure the power of that model by calculating the average of FAR and FRR by setting the pass mark for each that testing login attempts.

2.2 Literature Survey

KD field rich with many studies which uses in generally touch screen and keyboard when measuring the user typing rhythm, this research literature with some studies that used the keyboard as a major input device and able to compare with it outcome analyses such as the CMU and middle east university MEU past studies in this field, especially they evaluated with a unique benchmark published by CMU.

(Giot, R., El-Abed, M., Hemery, B., & Rosenberger, C., 2011)

This paper proposed a new method that allowed users are authenticated through the KD of a shared secret, this method based on the Support Vector Machine (SVM) learning satisfying industrial conditions (i.e., the enrollment phase that aimed to create its template needed for few samples per user). They used a large database that consists of (100) user for validation purposes. The proposed method compared with six methods from the past studies (selected based on their ability to work with few enrollment samples). Experimental results improved that, even though the computation time to build the template can be longer with their method 54 s against 3 s for most of the others, its performance outperforms the other methods and the Equal Error Rate of 15.28% against 16.79% and 17.02% for the two best methods of past studies, on their dataset and five samples to create the template, with a better computation time than the second best method.

(Al-Jarrah, 2012)

This paper presented an anomaly detector for KD authentication, based on a statistical measure of proximity, evaluated through the empirical study of an independent benchmark of KD data. The anomaly detection in the authentication process of

determining genuine users and impostors depended on classifying the password typing-rhythm.

Two phases are involved in the proposed user authentication methods. First a training phase in which a user typing profile is created through repeated entry of password. In the testing phase, the password typing rhythm of the user is compared with the stored typing profile, to determine whether it is a genuine user or an impostor. The typing rhythm is obtained through keystroke timings of key-down / key-up of individual keys and the latency between keys. The training data are stored as a typing profile, consisting of types of vectors, a vector of median values of elements of the feature set, and as a vector of standard deviations (STD) for the same elements. The proposed classifier algorithm computes a score for the typing of a password to determine authenticity. A measure of proximity is used in the comparison between feature set medians vector and feature set testing vector. Each feature in the testing vector is given a binary score of 1 if it is within a proximity distance threshold from the stored median of that feature, otherwise the score is 0. The proximity distance threshold for a feature is chosen to be the STD of that feature in the training data. The typing of a password is classified as genuine if the accumulated score for all features meet a minimum acceptance threshold. Analysis of the benchmark dataset using the proposed classifier has given an improved anomaly detection performance in comparison with results of 14 algorithms that were previously tested using the same benchmark. As presented in this paper, the Medians Vector Proximity algorithm (the proposed algorithm) has the lowest equal error rate (0.08), indicating that it has the highest anomaly detection performance in comparison with the literature 14 algorithms.

(Killourhy, K. S., 2012)

This study investigated many of classifiers and tests it with a unified dataset to compare the result of these classifiers and sort the result ascending. In the past thirty years, dozens of classifiers have been proposed for distinguishing people using KD; many have obtained excellent results in evaluation. However, when evaluations are replicated, the results are often wildly different; one classifier's error rate jumped from 1% to 85% upon replication .Classifier error rates depend on a multitude of factors; until the effects of these factors on error rates are understood, KD cannot realize its promise. To tackle this multitude-of-factors problem, they developed the following methodology: (1) evaluate multiple classifiers under systematically ranging conditions; (2) analyze the results with linear mixed-effects models (LMMs), a technique for inferential statistics well suited to understanding how various factors affect classifier error rates; and (3) validate the models, demonstrating that they accurately predict error rates in subsequent evaluations.

Table 2.1 Miss-rate comparison (*Killourhy, K. S.*, 2012)

Classifier	False-Alarm Rate	Miss Rate
ScaledManhattan	5.0	23.6
KNN	5.0	29.8
SVM	5.0	30.2
OutlierCount	2.9	31.7
MahalanobisKNN	5.0	33.7
KMeans	5.0	35.0
Mahalanobis	5.0	39.1
Manhattan	5.0	41.8
AutoAssocNNet	5.0	56.3
Euclidean	5.0	61.0

Table 2.1: Shows the average error rates for the 12 classifiers on the benchmark data. False-alarm and miss rates are presented as percentages values. Classifiers were tuned to have a 5% false-alarm rate (insofar as possible) to focus on what are the miss rate will be, and results are sorted by miss rate.

(Zhong, Y., Deng, Y., & Jain, A. K., 2012)

In this study they investigated the problem of user authentication using keystroke biometrics. A new distance metric that is effective in dealing with the challenges intrinsic to keystroke dynamics data, i.e., scale variations, feature interactions and redundancies, and outliers is proposed. They keystroke biometrics algorithms based on this new distance metric are evaluated on the CMU keystroke dynamics benchmark dataset and are shown to be superior to algorithms using traditional distance metrics.

They proposed a new distance metric combining both Mahalanobis distance and Manhattan distance such that one complements the other. First, they applied the principle of Mahalanobis distance to de-correlate and normalize the KD feature variables so that the covariance matrix of the transformed feature vectors becomes an identity matrix. This rectifying process is accomplished by applying the following linear transform to the input keystroke dynamics data.

This study evaluated the proposed keystroke authentication algorithms using the CMU KD benchmark dataset because it comes with the performance numbers for a range of existing KD algorithms for objective comparison. And the equal error rate outcome with 8%.

(Al.Jarrah, 2013)

This study discussed a combination of KD with one time password OTP technique, the author presented a multi-factor authentication scheme based on a combination of typing rhythm, user chosen password and system generated passcode. The aim was to strengthen user authentication, which has traditionally been based on passwords, with additional factors that can improve the rate of impostor detection. The proposed authentication scheme involves four levels: password, passcode, typing rhythm and re-typing rhythm.

There are four levels in this study, in the first level, the password is verified and at the same time the typing rhythm is recorded through keystroke timings. If the password is correct the user enters a second level where he/she types a short 4-digit personal identification number PIN that was previously generated by the system. If the PIN is correct, the system enters the third level in which typing rhythm of the password is matched against the stored typing rhythm profile (Template).

If the three types of password, passcode and typing rhythm are successfully matched then login attempt is accepted. In case of typing rhythm mismatch, the user is given a second chance, having already succeeded in password and PIN, so the user enters a fourth level in which he/she re-types the password. If the keystroke timings of re-typing the password gave an acceptable match to the stored profile, the user will be identified as legitimate, otherwise even though having given correct password and PIN the login is rejected as an impostor attempt.

(AL-Rahmani, 2014)

This research examined KD approach as a biometric authentication scheme that does not require extra hardware. The study was focused on enhancing an anomaly detector

that is based on a statistical model of classifying the typing rhythm of a person who is trying to access a computer system, whether it is a genuine user or an imposter.

Anomaly detector model was proposed, which uses the (median vector) for each typing feature element of as the point of center to measure acceptance against, and a distance to median (DTM) threshold value which gives the upper and lower limits for an acceptable feature element. The proposed model was evaluated using a public benchmark dataset of 20,400 records of password typing time measurement, collected by the biometrics lab of Carnagei Melon University CMU. The proposed model achieved lowest error rates of False Acceptance and False Rejection, compared to previous results of using other models on the same dataset. The research outcome with equal error rate **0.071**.

(Syed, Z. A., 2014)

This study provides contributions to advances two types of behavioral biometrics applicable to desktop and mobile computers: KD and touch dynamics. KD relies upon the manner of typing rather than what is typed to authenticate users. Similarly, a continual touch based authentication that actively authenticates the user is a more natural alternative for mobile devices.

This study shows the significant impact of habituation on user behavior, within the KD domain; habituation refers to the evolution of user typing pattern over time. It offers empirical evidence of the significant impact on authentication systems attempting to identify a genuine user affected by habituation, and the effect of habituation on similarities between genuine and impostors. It also proposes a novel effective feature for the KD domain called event sequences. To provides a unique advantage in distinguishing between users when typing complex should showing empirically that

mismatch features from traditional KD literature, event sequences are independent of typing speed.

(Idrus, S. Z., 2015)

This study illustrates several approaches on how soft biometric information can be combined into KD user authentication systems. It is divided into two parts: *(i)* the development of KD baseline system i.e. verification method (classical); and *(ii)* defining how soft criteria can be combined with classical KD to obtain a better performance than the baseline system i.e. this study assume combination method. Similarly to any other biometric authentication applications, the performance specie cations of the system is evaluated by measuring the number of correct and false verifications (false match rate (FMR) and false non match rate (FNMR)), which then is reported in the form of Equal Error Rate values. For the baseline system, the researcher performed user authentication with computations in order to obtain the verification performance scores from all 5 known passwords i.e. raw scores. It is considered as the foundation of their KD authentication system and its performance is decided by the equal error rate EER values.

2.3 KD Classifiers

This thesis discusses the research results and compares it with past models that used different types of classifiers to evaluate the digestive classifiers in the proposed multi-model anomaly detector. This section presents a literature overview for some classifiers that used by the past studies.

2.3.1 Median-Median Algorithm

This anomaly detector was created by (AL-Rahmani, 2014). The study aimed to enhanced MED-STD model of the AL-Jarrah study in using a different measure of distance to median (DTM), as a metric of anomaly from the normal typing behavior which is centered around the median as a point-of center. The assumption here is that the standard deviation is derived from the mean, which can be affected by extreme or outlier values, this classifier depending on training with user typing rhythm for 31 features and test the password 31 features with the training features and score the matching features with 1 otherwise 0 and calculate the features that scored with 1 to determine this login attempt genuine if the (total scores) 1's pass a threshold called (pass mark). Therefore the proposed model is based on the following criteria:

a) The median of timing values of each typed character, obtained during the training session, is considered as a reference center-point to measure acceptance or rejection against.

b) The DTM value, measured for each character of the password individually, during the training session, used for detecting the genuine/ impostor user at the testing session.

c) The DTM is calculated as a function of the median rather than the mean. As below:

DTM = C x M

Where

M = median of timing values of the key

C = multiplying constant which (0.7).

d) During the training phase, a template vector is created, which is a vector of Median and DTM values for the password.

e) During the testing phase, the timing value of a password character is considered acceptable if it lies within the upper and lower limits around the median of that character.

Upper limit = median + (DTM), as the DTM defined before.

Lower limit = the minimum value for all character individually.

2.3.2 Median-STD (Median Vector Proximity)

This anomaly detector was created by (Al-Jarrah, 2012), that use the median vector with STD standard deviation the DTM = STD.

The classifier work as the below carried out in two levels:

Training level Steps:

Step1: Calculate medians vector for a set of Features Timings of a group of password typing entries.

Step2: Calculate standard deviations vector for the set of features timings in step 1.

Testing (Classifier) level Steps

Step1: Get the features timings vector for the test-typing of the password.

Step2: For each feature element, mark the Feature-Score as 1 if the feature timing is within the proximity Distance from the median of the feature element timing, otherwise mark it as 0.

Step 3: Calculate the Test-Score (TS) as the sum of Feature-Score of vector elements.

Step 4: Mark the test-typing of the password as genuine if the Test-Score is >= Pass-Score, and as impostor otherwise.

From the levels and its steps above, login attempt will accept as genuine or reject as impostor after the training that collect the information about a certain user that enter the password for N times to extract the (template) by calculate the (Median, Standard Deviation, Minimum, Maximum) after building a template for each the feature set will compare the login attempt with its feature set template individually to determine how much this attempt get a scour of true matching features to determine is it genuine or impostor just like the previous classifier, So the median and STD the major vectors for this algorithm.

2.3.3 Manhattan

This classifier has the advantages of simplicity in computation and easy decomposition into contributions made by each variable. Most importantly, it is more robust to the influence of outliers compared to higher order distance metrics including Euclidean distance and Mahalanobis distance.

This resembles the Euclidean detector except that the distance measure is Manhattan (or city block) distance. There are two phases in this classifier (training and testing). In the training phase, the mean vector of the timing vectors is calculated.

In the test phase, the anomaly score is calculated as the Manhattan distance between the mean vector and the test vector (TV). Manhattan distance is used to find the distance between referring keystroke feature vector and the feature vector to be classified. As a result, Manhattan distance is more robust than Mahalanobis distance in the presence of outliers. The Manhattan distance also has a statistical interpretation as the Mahalanobis distance. It is in fact related to the log likelihood of the multivariate Laplace distribution with an identity covariance matrix (Zhao, Y., 2006).

2.3.4 Manhattan (Filtered)

This detector was described by (Joyce, R., & Gupta, G., 1990). It is similar to the Manhattan detector except outliers in the training data are filtered. In the training phase, the mean vector of the timing vectors is calculated, and also calculated the standard deviation for each feature.

2.3.5 Manhattan (scaled)

This detector was described by (Araújo, L. C., Sucupira, L. H., Lizarraga, M. G., Ling, L. L., & Yabu-Uti, J. B. T., 2005). This classifier has also two phases.

In the training phase, calculated the mean vector of the timing vectors, and the mean absolute deviation of each feature is calculated as well.

In the testing phase, the calculation is similar to the Manhattan distance, but with a small change is the anomaly score is calculated as $\sum_{i=1}^{p} |xi - yi| / ai$ where xi and yi are the i-th features of the test and mean vectors respectively, and ai is the average absolute deviation from the training phase.

The score resembles a Manhattan-distance calculation, except each dimension is scaled by ai (Maxion, 2009).

2.3.6 Nearest Neighbor

Classification based verification approach is always deployed for the problem of biometric authentication within a huge database where the input is unknown (Cho, S., Han, C., Han, D. H., & Kim, H. I., 2000). The main goal of classification is that it can significantly increase the matching efficiency. Nearest neighbor model is a simple classification method based on distance measurement. It works by applying a distance measurement between two data sets and then calculates the new value.

All data will be considered as a neighbor if the distance value is within a selected value (k), then. There is no general optimum value for (k) and it is usually found by using trial and error approach (Hu, J., Gingrich, D., & Sentosa, A., 2008).

2.3.7 Nearest Neighbor + Outlier Removal

This classifiers presented by (Zhong, Y., Deng, Y., & Jain, A. K., 2012) they used the Nearest Neighbor classifier with the new distance metric defined in to either ascertain a KD feature as originating from the genuine user when the distance to its nearest neighbor in the training data is below a threshold value, or reject it as an imposter, otherwise. The adoption of the new distance metric helps suppresses the adverse effects of outliers during the classification stage. However, outliers could still corrupt the training data and deteriorate the authentication performance. They employed an outlier removal process during the training phase. For the ith feature variable, they sorted the measurements from the training data and compute the median and standard deviation using all training measurements excluding those in the upper and lower percentiles. Only the training feature vectors with their ith variable falling in the interval are retained and those falling outside of the interval are discarded from the training data.

Once the outliers are removed from the training data, they used the Nearest Neighbor classifier with the new distance metric to classify the test keystroke feature vectors. So, they essentially had two different new metric based nearest neighbor classification algorithms: one without outlier removal and one with outlier removal.

2.3.8 Disorder Classifier

The disorder classifier works on n-graphs rather than discrete words. The disorder classifier is interesting because of its usefulness on free text and its dissimilarity to the statistical classifiers discussed so far.

The training used samples of free or transcribed text. The most common n-graphs are obtained from the samples; there will be few n-graphs longer than characters. The n-graphs are sorted by timing features.

For more clarifying, if the digraphs “en”, “th”, and “er” are all well-represented in the sample, and they take the inter-key light time as the metric, they might find that the “th” has the shortest flight time, followed by “en” and then “er” ([“th”, “en”, “er”]). As the user types, the n-graphs in the sample are sorted under the same principle. Using n-graphs that occur in both the training text and the sample, the “disorder” of the sample is computed. Essentially, they tallied up the total distance of “swaps” that would be required to put all of a sample’s n-graphs into their rightful places in the sorted training array. If their sample was ordered [“th”, “er”, “en”], they would have to make two swaps (“th” moves 0, “er” moves 1, “en” moves 1). If the total number of moves required is sufficiently below the total number of possible moves, then the sample is accepted. This method can also work on single key hold times under the same principle. (Ryan, S. A., 2014) (Killourhy, K. S., & Maxion, R. A., 2012).

Chapter 3

Anomaly Detector Models and the Multi-Model System

3.1 Introduction

The issue of combining several models in a multi-model system requires in the beginning an investigation of which models to be selected and the criteria for the selection. For the proposed multi-model, the criteria for the selection of the component anomaly detector models was the model's equal error rate EER, by choosing the best performing models in terms of EER, in other words models which have the lowest EER.

In addition to using existing models, this work is proposing two new models with good EER performance based on the benchmark data.

The proposed single models were evaluated individually along the lines of similar studies such as (AL-Rahmani, 2014) and (Al-Jarrah, 2012) in which they created new KD anomaly detector models.

First anomaly detector model to be included in the proposed multi-model system was the Enhanced Med-Med model (EMM), which is the outcome of our research in enhancing an existing model (AL-Rahmani, 2014), and was evaluated using the same approach and benchmark that were used in the original Med-Med model.

Second anomaly detector model in the proposed Multi-Model system is the new KD model (Abs-Min) that uses the minimum vector after converting the negative values to positive in the training data for each feature with the median as a center point, to be the DTM for all password character, this model was also similarly evaluated using the same approach and benchmark that were used with the first model.

Third and last anomaly detector model included the proposed multi-model system is the (Med-Std) which proposed in (Al-Jarrah, 2012), it used the standard deviation as a DTM and also was evaluated using the same approach and benchmark that were used with the first model.

Those three component models of the multi-model have a similar way in reducing the effect of outlier data by choosing the median of a set of feature values as the point of center of that feature since the outliers' values do not distort the median, while the mean can be influenced by outliers.

The multi-model uses the outcome of the three component anomaly detector models to voting the final outcome of the login attempt, which is either “Impostor” or “Genuine”. A comparison will be made between the outcomes of the multi-model against that of the three single models working individually for the login attempts.

3.2 The Single Anomaly Detectors

3.2.1 The Enhanced Median-Median Model (EMM), (model #1)

The proposed enhancement of the Median-Median model uses the same approach in measuring the DTM), in which the median is considered as the central point between upper and lower limits. A feature value is considered genuine if it is within the upper and lower limits around the median of that feature. According to the experimental analysis that was carried out using the benchmark, this research observed that reducing the DTM value by decreasing the constant (C) from (0.7) to (0.42) has resulted in lower EER error in comparison with the Median-Median model which means that there will be less false acceptance of impostors using the new model. There are many of analyses made by this research to reduce the DTM and this research find out if the DTM decreases or increased will give bad result according to the dataset that tested, so this research stop the analyses immediately when outcome better EER that discusses in chapter 4.

Figures 3.1 shows a sample of a template for 31 features, which consists of upper and lower limits (UL and LL) for subject number 57 in the CMU benchmark, calculated using the original med-med model. For comparison, Figure 3.2 shows the new upper limit (UL) using the EMM model, after reducing the constant (C) value, which resulted in a reduced range between the upper and lower limits.

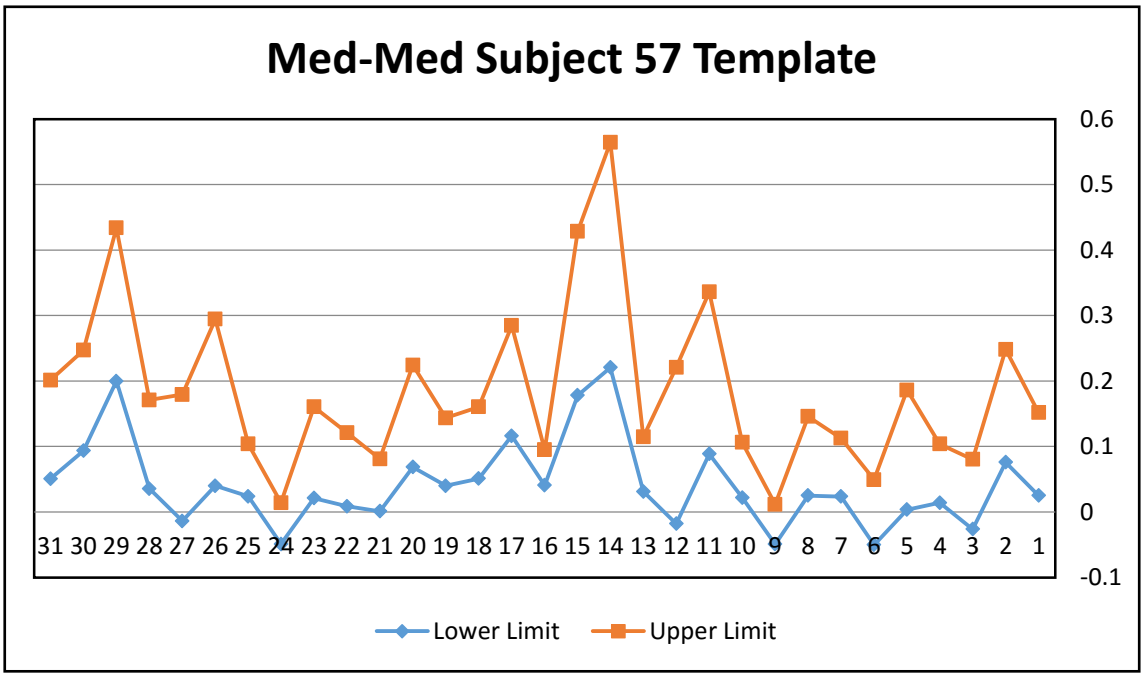

Figure 3.1Med-Med Template Sample of subject No.57 in CMU benchmark

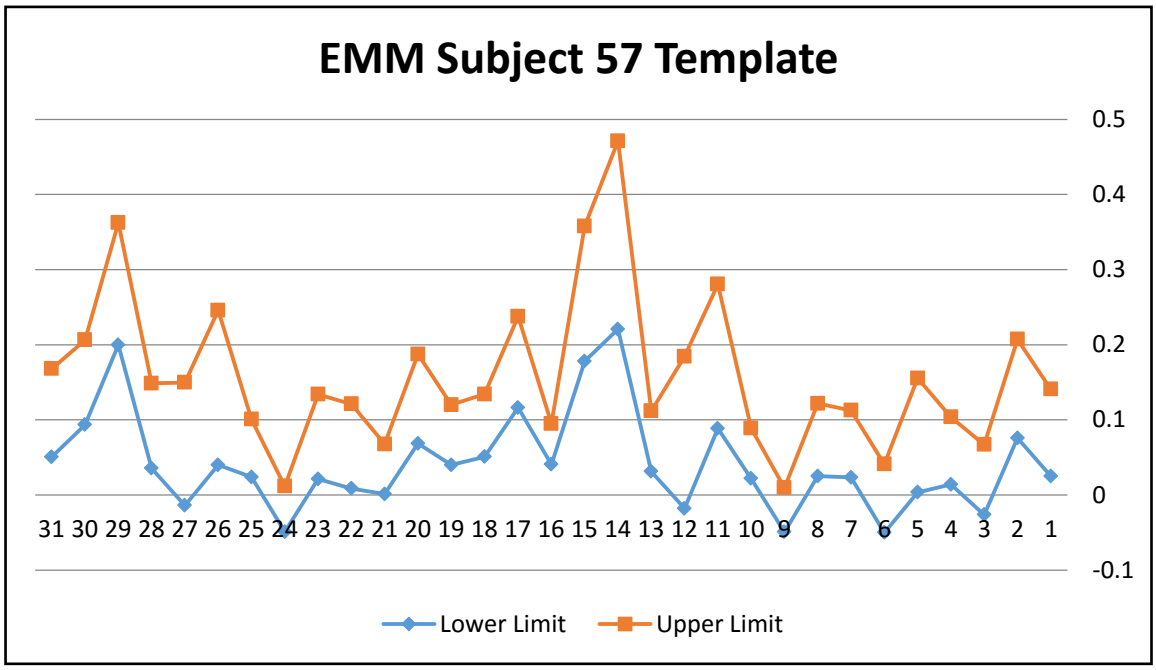

Figure 3.2 EMM Template Sample of subject No.57 in CMU benchmark

The proposed EMM model has two phases, training and testing, as in the following:

- **Training phase:**

During this phase a user or experimental subject types a password a certain number of times, and the keystroke timing raw data are used to generate template vectors, as in step below:

A. Converting the keystrokes to time periods in milliseconds and extracting the features vectors for all training repetitions of each subject.

B. The DTM and the median are calculated for each feature vector individually.

C. A template is generated for each user which consists of two vectors, the lower limit (LL) and the upper limit (UL), where each vector consists of 31 values representing the 31 features of the 10-character password. The template is stored in a database for later use during testing. Calculation of the template limits are stated below:

Lower Limit (LL) = minimum value for each training data feature individually.

Upper Limit (UL) = median (M) + DTM for each training data feature individually.

M = Median of the values of a feature element

DTM = C x M

C = 0.42 (the constant factor of 0.42 was calculated empirically).

- **Testing phase:**

During this phase a user or an experimental subjects types the same password as in training. For a user being authenticated, the password is entered once, while in an experiment the subject enters the password a certain number of times. In this phase the timing raw data of the keystrokes are used to generate a test vector which consists of 31

test elements representing the 31 features of the 10-character password. The following steps summarize the work of the testing phase:

A. Converting keystroke to time period in milliseconds for the test typing of the login attempt and store it as test record.

B. Extracting the test vector of features (Hold, DD, and UD) from the test timing raw data. The vector consists of 31 test feature elements.

C. In user authentication, gets the training template of the user from the database using his user id or user name.

D. Matching test vector elements with the corresponding UL and LL of the template elements and calculating the test score of the user.

A test element is considered genuine if it is within upper and lower limits of the particular feature element of the same user.

E. Counting the number of test elements which are considered as genuine.

F. Classifying the login attempt as genuine if the count of the test vector elements which are classified as genuine equals or exceeds a pre-determined Pass Mark (PM).

The pre-determined Pass Mark is calculated from experimental analysis of the benchmark dataset; first of all the pass mark set with initial value for example the 31 features analyses assuming M = 25 of 31, this number of PM will affected 200 tests of genuine and 250 tests of impostor abstracted to error rates FAR and FRR to calculate the average of them to found the EER. Hence the pass mark is adjusted to bring the FAR and FRR to equality.

3.2.2 The Proposed Absolute Minimum (Abs-Min) model (model#2)

The proposed absolute-minimum model is a new anomaly detector model formulated in this research according to empirical investigation of the dataset. This model is also based on measuring the distance from the median of a set of training feature values, in order to classify a test feature value as either impostor or genuine. The new DTM measure was selected by the researcher from several alternative functions that were investigated empirically to find out as to which measure gives lower EER error rate.

The reason of using the absolute value of the minimum of a set of feature values rather than just the minimum is that the latency features of UD and DD can have negative values, as in the CMU benchmark. Such negative values occur when the typist uses fast two hands touch typing on the keyboard, where sometimes the second key of a pair of characters is pressed before release of the first key, which results in a negative UD value because the down time of key2 is smaller than the up time of key1.

The negative value of a latencies (UD and DD) are unreal because time span cannot be negative (may be possible in science fiction !), and also a negative latency will distort the actual minimum value, therefore, the absolute value of UD and DD was the answer to eliminating the negative effect of negative values.

The choice of the minimum value of a feature set, after convert the negative signal to the positive for all training data as the DTM, is a result of the “learning from data” approach in which empirical comparison of alternative measures was made, as noted earlier.

The proposed absolute-minimum anomaly detector is applied in two phases, training and testing, as in model#1, and follows the same steps. The lower and upper limits are calculated as below:

Lower Limit (LL) = minimum value for each training data feature individually.

Upper Limit (UL) = median (M) + DTM for each training data feature individually.

DTM = the minimum of the absolute values for each training data latencies features individually.

3.2.3 The Standard-Deviation (Med-Std) model, (model#3)

Third model that is combined in the multi-model anomaly detector is the Med-Std model which was proposed in (Al-Jarrah, 2012). The selected anomaly detector model was formulated using the median of a set of feature values and the standard deviation of the same set of feature values. This anomaly detector is applied in two phases, training and testing, and follows the same steps as in the previous two models.

The lower and upper limits are calculated for each training data feature individually as follows:

Lower Limit (LL) = median (M) – standard deviation (Std).

Upper Limit (UL) = median (M) + standard deviation (Std).

3.3 The Proposed Multi-Model Anomaly Detector System (MMD)

The multi-modal anomaly detector model is aimed to reduce the classification errors for the login attempts in the KD system by combining more than one model together and taking a vote for the final decision on the classification. The outcome of the individual single models is taken into the vote, and the final decision is based on majority of votes, not on absolute majority.

The pass mark PM in the multi-model unlike the PM in the previous single models, in this proposed model PM has fixed number for all subjects because it determined before according to the analyses that calculated its average EER and the average pass mark that that resulted from these analyses.

The proposed multi-model anomaly system (MMD) is implemented as a KD data collection system to be used for data collection to support further analysis of the multi-model anomaly detection concept. At the same time the MMD system is aimed to be a live KD authentication prototype tool for actual experimental work on KD authentication. The MMD system is designed to allow single model or multi-model modes of authentication.

The proposed multi-modal anomaly detector system consists of two phases, as stated below:

- **Training phase:**

This phase starts with a new user registration step, and ends when this registration is completed and the typing profile templates are stored in the database. The templates of the typing profile of each user in the multi-model system consist of three single model templates, where each individual model template consists of two vectors, the lower and upper limits (thresholds).

In this phase the typing rhythm of a particular user is learned through a number of repetitions of typing the same password. The number of repetitions for this experimental work was chosen to be 30 repetitions in a KD authentication tool.

A static version of this MMD was implemented in Excel to analyze benchmark datasets such as the case of the CMU benchmark which had 200 typing rows for training,

collected for 51 subjects and the static analyses aimed to analyze the recorded rows that will generated by this phase of MMD, which aimed to record 30 typing rows for training collected for 20 subjects.

Figure 3.3 shows flowchart of the training phase process, in which three templates (TMP1, TMP2, and TPM3) are generated for the training entry of 30 password typing attempts of a particular user. A mistyped password is rejected and re-entry is allowed. After completion of 30 correct entries, the templates are stored in the database with a user-id.

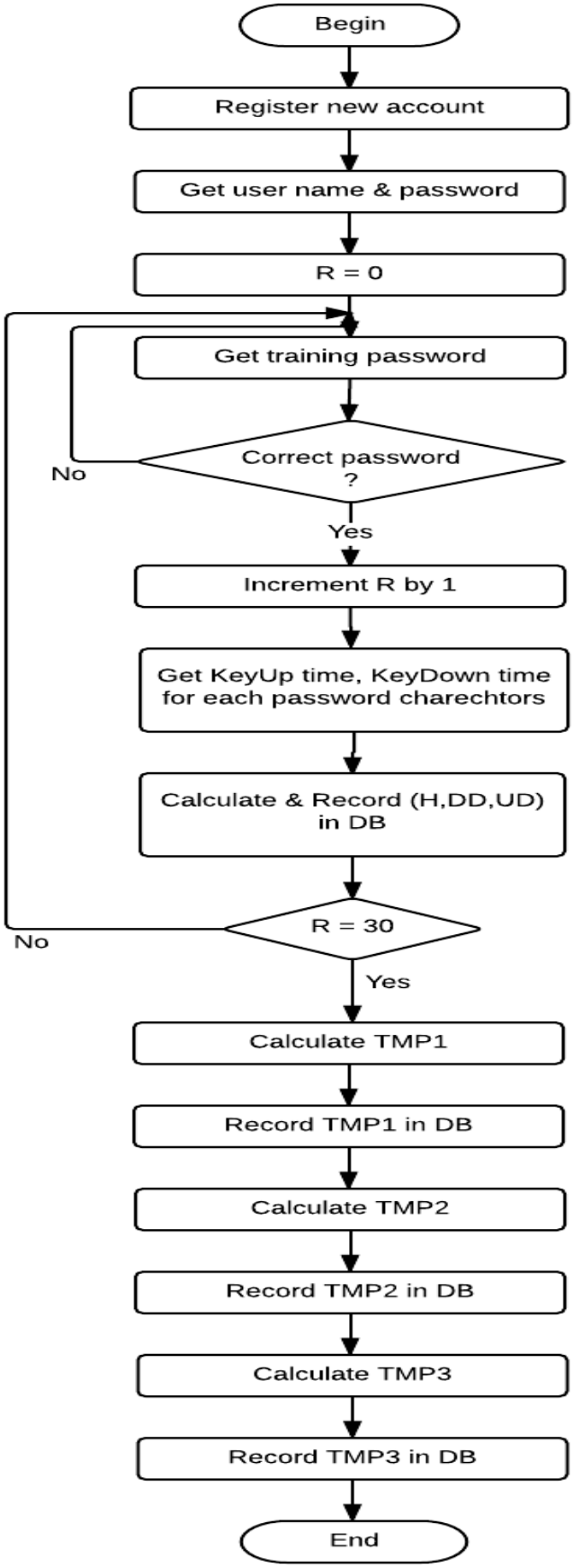

Figure 3.3 Training Phase Flowcharts

- **Testing phase:**

This phase aims to test the login attempt of a particular user who has already registered and entered his/she's training data. In this phase a matching is done between the testing features vector of the password and the corresponding features vectors in the templates that are retrieved from the database.

Test score for each model is calculated based on the matching results between the test vector and the templates vectors (31 test features with 31 training features in case of 10-character password), then for each model the classification is made as “genuine” if the test score is equal to or higher than the pass mark of that model, otherwise the result is “impostor”. A final classification decision is made in the multi-model by taking a vote of “genuine” or “imposter” of the three models, and a test attempt is given a multi-model “genuine” result if it gets two or three “genuine” results by the single models.

Figure 3.3 shows a flowchart of the testing phase process. The sequence of operations starts with retrieving the threshold templates (TMP1, TMP2, TMP3) from the database for comparison with password features that will be extracted from the test typing of the password and stored in the Test Vector (TV).

There are two limits for each template, upper and lower thresholds, each consisting of 31 values for the 31 features (Hold, DD and UD) of the 10-character password. Each of the 31 password test features in the test vector will get a score of 1 if it is within the upper and lower thresholds of that feature, otherwise it gets a score of 0. The comparison between the test vector values and the templates threshold values will be repeated for the three models individually, and then a final classification result of “genuine” or “impostor” is given based on the majority vote of the three single models.

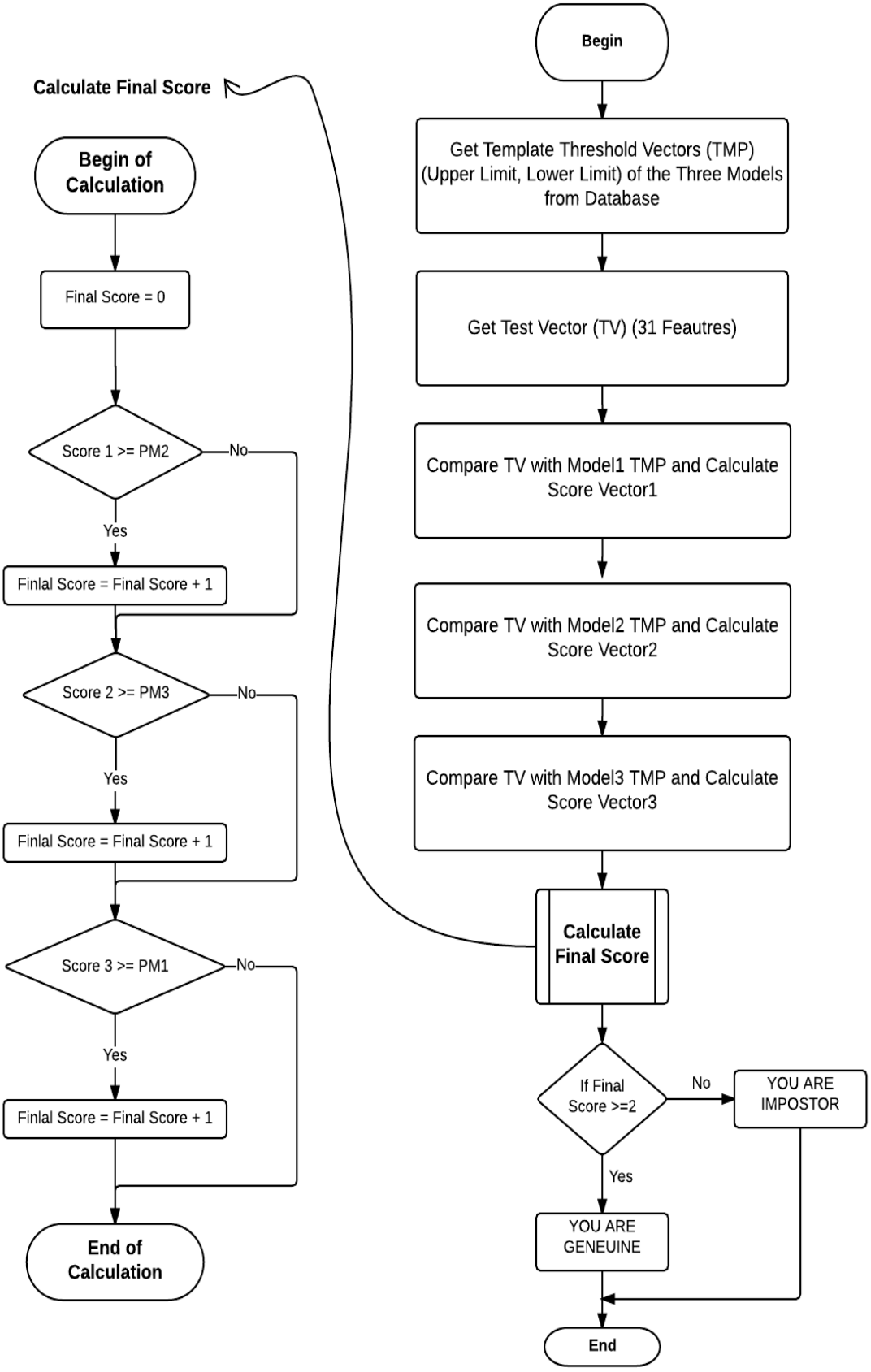

Figure 3.4 Testing phase flowchart

3.4 Modules of the Multi-Model Anomaly Detector (MMD) Tool

This research supported with this tool that collect a data by recording the training data to extracted later and to test the data as a live static test by the multi-model as the following.

3.4.1 The Purpose of the MMD Software is Two Folds:

A. To be a KD data collection tool for further experimental data acquisition of typing data, to help in the creation of new dataset. This part is a training part that does not relate to any anomaly detector model.

B. To be a live KD authentication tool, providing training and testing phases, that can be used to verify the actual authentication performance of the proposed models.

3.4.2 Register New User

The purpose of this module is to register a new user, collect training keystrokes timing data and generate the template features vectors for the 30 training repetitions. The initial step gets and store username and password. Figure 3.5 shows the interface for creating new account in the MMD tool.

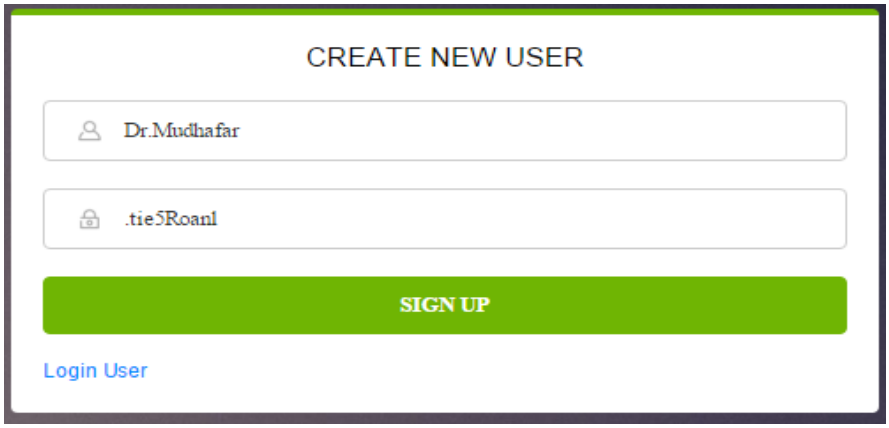

Figure 3.5: Registering new account

3.4.2.1 Data Collection Enrollment

To enroll a new user in the database, the password is typed 30 times as shown in figure 3.6. A mistyped password is rejected and the user is allowed a re-entry.

A console window is provided to display features values (Hold, UD and DD) data of the correctly entered passwords and to show an error message if the password is mistyped, as shown in figure 3.7.

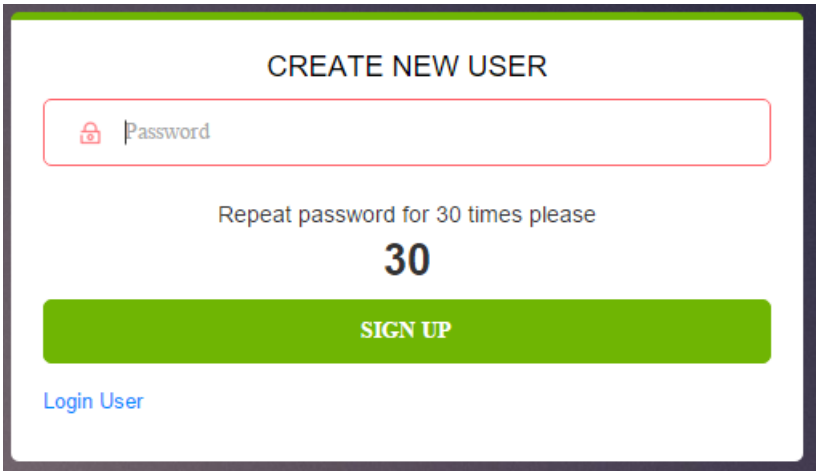

Figure 3.6: Repetitions enrolment

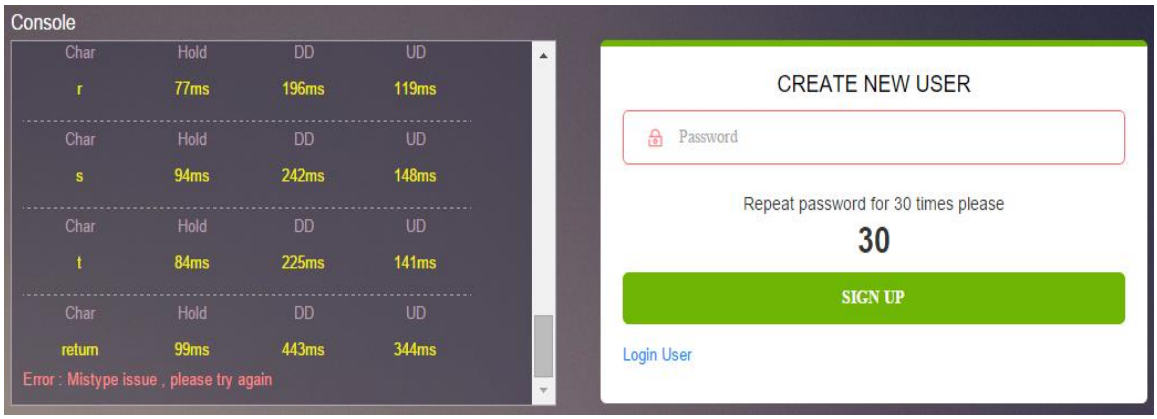

Figure 3.7: Console-mistyping error messages

After completion of the 30 correct password entries, this module will generate the template vectors and store them in the database.

3.4.3 Login-User

The purpose of this module is to provide the testing phase of the MMD system. This module utilizes the three single models to make a decision on the login attempt by comparing the login password features with the models template threshold vectors individually.

The input to the module is username, to be used for retrieving his template thresholds vectors, and the password.

The output of this module is a message stating genuine or impostor according to the voting of the three single modules. Figures 3.8, 3.9 show the authentication result output, which includes votes of the individual models.

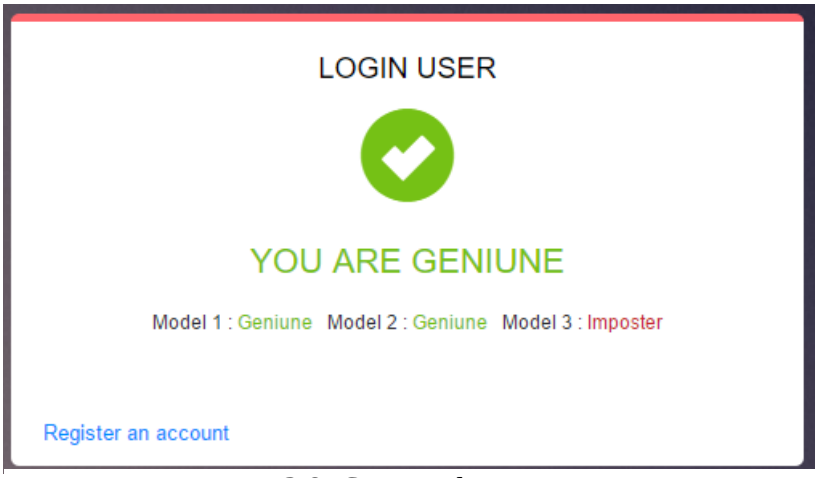

Figure 3.8: Genuine login attempt

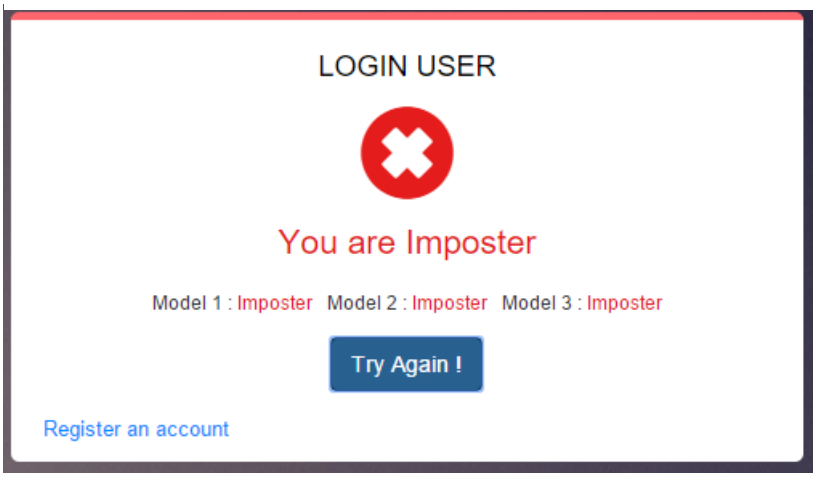

Figure 3.9: Impostor login attempt

3.4.4 The MMD Tool Implementation

The MMD tool was implemented using several software platforms, as noted below:

A. Getting the time for each event (KeyUp and KeyDown) during password typing. JavaSecrept was used for this task.

B. Template generation and comparison. PHP is used for this task.

C. Database storage of ussername, password and template vectors. The task was implemented in SQL DB.

D. Tool style was created by using CSS, bootstrap.

3.4.5 The Static Multi-Model Anomaly Detector

This module is an Excel based version of the system for the static analysis of exsiting KD datasets. The CMU dataset was analyzed using this module.

The module has the flexibility to change the anomaly detection model.

Firgure 3.10 shows a sample of the module main page, which summarizes the error rates of each the 51 subjects based on the analysis, including FAR, FRR, and EER for each user, as well as the average of EER for the entire population of the dataset.

These analyses tested 22,950 rows of data for each model to voting by the multi-model to output with 22,950 results abstracted as a multi-model FAR and FRR to find the average of them as a EER, finally compare the single model result with multi-model result to determined mesure the assuming EER value after voting.

Multi-Model Results

Subject	Model1 MM0.42 FRR	Model1 MM0.42 FAR	Model2 Abs(Min) FRR	Model2 Abs(Min) FAR	Model3 Std FRR	Model3 Std FAR	Multi-Model FRR	Multi-Model FAR	Genuine Accept Rate	Imposter Hit Rate
2	11.50%	16.80%	10.00%	10.00%	29.50%	8.80%	**13.00%**	**12.00%**	87.00%	88.00%
3	8.50%	5.20%	23.50%	7.20%	28.50%	3.60%	**17.50%**	**5.60%**	82.50%	94.40%
4	2.00%	9.60%	1.50%	10.00%	11.50%	9.20%	**1.50%**	**8.40%**	98.50%	91.60%
5	0.50%	15.60%	2.50%	12.00%	2.50%	3.60%	**1.00%**	**10.80%**	99.00%	89.20%
7	6.00%	5.60%	10.00%	4.40%	21.00%	3.60%	**7.50%**	**3.60%**	92.50%	96.40%
8	2.50%	4.80%	3.00%	4.00%	6.00%	4.40%	**2.50%**	**4.00%**	97.50%	96.00%
10	1.00%	1.20%	1.50%	0.80%	1.00%	1.60%	**1.00%**	**1.20%**	99.00%	98.80%
11	5.00%	1.60%	25.00%	1.20%	7.50%	2.00%	**7.50%**	**1.20%**	92.50%	98.80%
12	4.50%	0.80%	4.00%	6.40%	1.00%	2.00%	**3.00%**	**1.20%**	97.00%	98.80%
13	1.50%	0.80%	2.00%	1.60%	6.50%	2.40%	**1.50%**	**1.20%**	98.50%	98.80%
15	3.00%	2.80%	12.50%	3.20%	2.50%	11.20%	**4.00%**	**3.60%**	96.00%	96.40%
16	19.00%	24.00%	28.00%	19.60%	42.00%	8.00%	**25.00%**	**19.60%**	75.00%	80.40%
17	2.00%	2.80%	5.50%	1.20%	10.00%	0.40%	**5.00%**	**0.80%**	95.00%	99.20%
18	8.00%	8.40%	9.50%	8.00%	2.00%	11.60%	**6.50%**	**8.40%**	93.50%	91.60%
19	0.50%	4.40%	1.00%	4.80%	3.00%	2.80%	**0.50%**	**2.80%**	99.50%	97.20%
20	28.00%	2.00%	23.50%	4.40%	7.50%	18.80%	**21.00%**	**4.00%**	79.00%	96.00%
21	5.00%	8.00%	8.50%	7.20%	5.00%	12.80%	**5.00%**	**7.60%**	95.00%	92.40%
22	4.00%	0.80%	7.00%	1.20%	13.50%	0.00%	**4.50%**	**0.80%**	95.50%	99.20%
24	11.50%	16.80%	10.00%	10.00%	29.50%	8.80%	**13.00%**	**12.00%**	87.00%	88.00%
25	6.00%	8.80%	8.00%	6.00%	5.50%	7.20%	**6.50%**	**7.60%**	93.50%	92.40%
26	5.00%	6.00%	4.00%	5.20%	5.00%	4.00%	**4.00%**	**4.80%**	96.00%	95.20%
27	3.00%	8.00%	4.00%	9.20%	4.00%	7.20%	**2.50%**	**7.20%**	97.50%	92.80%
28	4.50%	2.40%	5.00%	2.80%	20.00%	2.00%	**5.50%**	**2.40%**	94.50%	97.60%
29	3.00%	4.40%	6.00%	4.00%	7.50%	3.20%	**4.50%**	**4.00%**	95.50%	96.00%

Pass Mark1,MED.42	**23**
Pass Mark2,ABS(min)	**24**
Pass Mark3,STD	**21**
Features	**31**
PM Ratio1	**74.19%**
PM Ratio2	**77.42%**
PM Ratio3	**67.74%**
Avg Hit Rate	**92.01%**
Avg Accept Rate	**93.25%**

sajjad alrobay3i: This is best Pass-mark for model1 !

Evalution (Single Model VS Multi-Model)

Model#1 FAR	Model#1 FRR	Model#2 FAR	Model#2 FRR	Model#3 FAR	Model#3 FRR	Multi-Model FAR	Multi-Model FRR
6.27%	8.25%	10.09%	7.52%	8.74%	11.80%	6.75%	7.99%
7.26%		**8.80%**		**10.27%**		**7.37%**	

Figure 3.10: Sample of the analyses main page

Chapter 4

Analysis and Discussion of Results

4.1 Evaluation of the Proposed Models

This research evaluated the proposed anomaly detector models, EMM and AbsMin, using the CMU and MEU dataset, and the Multi-Model using only the CMU dataset.

To understand the anomaly detector models behavior on the same environment this research made different types of evaluation according to the past studies (AL-Rahmani, 2014), (Al-Jarrah, 2012), and (Killourhy, K. S., 2012) (Killourhy, K. S., & Maxion, R., 2009) as the following:

A. EER evaluation, measures the error detection performance of the anomaly detector model at the point of equality of false acceptance rate FAR and false rejection rate FRR. It is used for comparison of the detection performance of various models. This evolution occurred on all the proposed models (single model and multi-model) to compare its results with the past models.

B. Miss-Rate evaluation, measures the Miss-Rate which is FAR at an acceptable level of FRR, as proposed in (Killourhy, K. S., 2012).

   The FRR rate is fixed at a practical limit of 5%, i.e. a rejection of 1 in 20 login attempts, which is tolerated in a normal login situation. This metric gives a better measure of comparison between models, because it demonstrates in a practical way that one model is better than another when the first model allows less impostors at the same level of rejecting genuine users of the two models. This type of evolution occurred only on the proposed single models to compare it results with the past models.

4.2 EER Comparison on MEU and CMU Datasets

4.2.1 EER Evaluation of Single Model on CMU Benchmark

Table 4.1 shows the EER of the proposed EMM model.

Table 4.2 shows the EER of the proposed AbsMin model.

Tables 4.1 and Table 4.2 show the error metrics of FAR, FRR and EER, as well as the pass-mark for the each of the 51 subjects in the published CMU benchmark and detailed as in the following:

- **Subject:** Subject number according to the CMU benchmark.
- **Pass mark (<= 31):** The pass mark value at which a typing test score is considered as genuine or imposter. Calculated as the adjustment that give best equality between FAR and FRR when find that EER.
- **Genuine Test:** It is the test when a genuine user's login data from the testing phase is evaluated against the same user's training template. In the CMU benchmark, there are 200 genuine user's testing attempts against a template generated by 200 training attempts of the same user.
- **Impostor Test**: It is the test when an impostor's login data from the testing phase is evaluated against the training template of a particular genuine user's template. In the CMU benchmark, 250 impostors' login data (5 from each other user) are evaluated against the training templates of each genuine user (for each of the 51 users, the other 50 users are considered impostors).
- **TA** or true negative: The number of true acceptances when a genuine login is classified as such.

- **FR** or false positive: The number of false rejections when a genuine login is classified as impostor.

- **FA** or false negative: The number of false acceptances when an impostor login is classified as genuine.

- **TR** or true positive: The number of true rejections when an impostor is classified as such.
- **FAR:** False acceptance rate = FA / 250.
- **FRR:** False rejection rate = FR / 200.
- **EER:** Equal error rate are the average of FAR and FRR.

Comparison of the results in the Table 4.1 and Table 4.2 with past models is presented in Table 4.5

Table 4.1 EER analysis of EMM Model on CMU benchmark (31 features)

Subject No.	Pass Mark (<= 31)	Genuine Test (200)		Impostor Test (250)		FAR	FRR	EER
		TA	FR	TR	FA			
2	23	177	23	208	42	0.168	0.115	0.142
3	22	191	9	233	17	0.068	0.045	0.057
4	24	194	6	237	13	0.052	0.030	0.041
5	25	193	7	239	11	0.044	0.035	0.040
7	23	188	12	236	14	0.056	0.060	0.058
8	24	190	10	242	8	0.032	0.050	0.041
10	23	198	2	247	3	0.012	0.010	0.011
11	22	193	7	244	6	0.024	0.035	0.030
12	22	195	5	243	7	0.028	0.025	0.027
13	23	197	3	248	2	0.008	0.015	0.012
15	23	194	6	243	7	0.028	0.030	0.029
16	23	162	38	190	60	0.240	0.190	0.215
17	23	196	4	243	7	0.028	0.020	0.024
18	23	184	16	229	21	0.084	0.080	0.082
19	24	198	2	245	5	0.020	0.010	0.015
20	20	176	24	230	20	0.080	0.120	0.100
21	24	184	16	236	14	0.056	0.080	0.068
22	22	195	5	248	2	0.008	0.025	0.017
24	24	194	6	244	6	0.024	0.030	0.027
25	24	184	16	232	18	0.072	0.080	0.076
26	23	190	10	235	15	0.060	0.050	0.055
27	24	187	13	239	11	0.044	0.065	0.055
28	22	194	6	241	9	0.036	0.030	0.033
29	23	194	6	239	11	0.044	0.030	0.037
30	25	172	28	222	28	0.112	0.140	0.126
31	24	168	32	203	47	0.188	0.160	0.174
32	21	166	34	223	27	0.108	0.170	0.139
33	23	180	20	216	34	0.136	0.100	0.118
34	22	179	21	232	18	0.072	0.105	0.089
35	22	173	27	214	36	0.144	0.135	0.140
36	22	198	2	248	2	0.008	0.010	0.009
37	23	188	12	231	19	0.076	0.060	0.068
38	25	190	10	244	6	0.024	0.050	0.037
39	24	188	12	240	10	0.040	0.060	0.050
40	23	166	34	196	54	0.216	0.170	0.193
41	23	187	13	236	14	0.056	0.065	0.061
42	25	197	3	247	3	0.012	0.015	0.014
43	23	197	3	247	3	0.012	0.015	0.014
44	24	190	10	243	7	0.028	0.050	0.039
46	26	183	17	236	14	0.056	0.085	0.071
47	24	158	42	201	49	0.196	0.210	0.203
48	24	196	4	241	9	0.036	0.020	0.028
49	25	193	7	237	13	0.052	0.035	0.044
50	25	184	16	239	11	0.044	0.080	0.062
51	25	188	12	241	9	0.036	0.060	0.048
52	23	195	5	244	6	0.024	0.025	0.025
53	22	198	2	246	4	0.016	0.010	0.013
54	25	186	14	245	5	0.020	0.070	0.045
55	21	198	2	249	1	0.004	0.010	0.007
56	22	194	6	238	12	0.048	0.030	0.039
57	22	191	9	235	15	0.060	0.045	0.053
AVG	23	187	13	235	15	0.062	0.064	**0.063**

Table 4.2: EER analysis of Abs-Min model on CMU dataset (31 features)

Subject No.	**Pass Mark (<= 31)**	**Genuine Test (200)**		**Impostor Test (250)**		**FAR**	**FRR**	**EER**
		TA	**FR**	**TR**	**FA**			
2	24	177	23	230	20	0.080	0.115	0.1
3	21	189	11	231	19	0.076	0.055	0.120
4	23	196	4	239	11	0.044	0.020	0.034
5	25	191	9	231	19	0.076	0.045	0.060
7	23	184	16	238	12	0.048	0.080	0.066
8	23	190	10	241	9	0.036	0.050	0.035
10	24	197	3	248	2	0.008	0.015	0.011
11	20	191	9	240	10	0.040	0.045	0.052
12	22	190	10	240	10	0.040	0.050	0.052
13	23	198	2	249	1	0.004	0.010	0.018
15	21	195	5	238	12	0.048	0.025	0.045
16	23	161	39	185	65	0.260	0.195	0.227
17	22	197	3	243	7	0.028	0.015	0.024
18	23	185	15	234	16	0.064	0.075	0.087
19	25	193	7	245	5	0.020	0.035	0.027
20	19	174	26	228	22	0.088	0.130	0.104
21	24	182	18	233	17	0.068	0.090	0.078
22	22	195	5	245	5	0.020	0.025	0.022
24	24	190	10	242	8	0.032	0.050	0.040
25	24	181	19	235	15	0.060	0.095	0.07
26	24	191	9	241	9	0.036	0.045	0.046
27	23	193	7	233	17	0.068	0.035	0.057
28	23	194	6	243	7	0.028	0.030	0.025
29	23	189	11	239	11	0.044	0.055	0.050
30	25	166	34	218	32	0.128	0.170	0.145
31	24	163	37	200	50	0.200	0.185	0.201
32	19	172	28	205	45	0.180	0.140	0.159
33	23	173	27	218	32	0.128	0.135	0.154
34	20	173	27	219	31	0.124	0.135	0.135
35	22	174	26	215	35	0.140	0.130	0.139
36	22	198	2	247	3	0.012	0.010	0.011
37	23	183	17	231	19	0.076	0.085	0.072
38	23	192	8	234	16	0.064	0.040	0.059
39	24	188	12	239	11	0.044	0.060	0.054
40	22	167	33	198	52	0.208	0.165	0.204
41	22	183	17	225	25	0.100	0.085	0.153
42	25	198	2	245	5	0.020	0.010	0.009
43	23	198	2	246	4	0.016	0.010	0.011
44	23	191	9	235	15	0.060	0.045	0.059
46	25	192	8	229	21	0.084	0.040	0.063
47	23	158	42	186	64	0.256	0.210	0.236
48	24	191	9	243	7	0.028	0.045	0.028
49	25	193	7	235	15	0.060	0.035	0.047
50	24	183	17	232	18	0.072	0.085	0.081
51	22	186	14	239	11	0.044	0.070	0.066
52	23	197	3	246	4	0.016	0.015	0.016
53	20	199	1	245	5	0.020	0.005	0.015
54	23	190	10	234	16	0.064	0.050	0.057
55	19	198	2	247	3	0.012	0.010	0.016
56	23	190	10	238	12	0.048	0.050	0.047
57	21	189	11	234	16	0.064	0.055	0.070
AVG	**23**	186	14	232	18	0.070	0.068	**0.069**

4.3 EER Evolution of Multi-Model on CMU Benchmark

Table 4.3 shows the average of error metrics FAR, FRR and EER for each individual model and multi-model.

The multi-model accepts or rejects the login attempts after taking votes of the three models. In this case the error metrics FAR, FRR and ERR for the multi-model will obtained. After this investigating the research can comparison the EER for each single model before the assuming combine with multi-model EER to get following observation.

1- The EER results for single models are slightly different than when they were tested individually because the pass mark changed to generate the multi-model equal error rate when they are part of the multi-model.
2- The error metric EER for the multi-model is 0.062 that shows multi-model EER lowest than model1 EER 0.063, which had the lower EER among the single models when they combined.
3- The two other single models (Abs-Min and Med-Std) have a higher EER than the multi-model, this clarifying that the multi-model voting enhanced the final decision on some of the 22,950 tested login attempts.

.

Table 4.3: Multi-model evaluation

Subject	Pass Mark	Model1 EMM		Model2 Abs-Min		Model3 Med-Std		Model4 Multi-Model	
		FRR1	FAR1	FRR2	FAR2	FRR3	FAR3	FRR	FAR
2	23	0.115	0.168	0.060	0.108	0.395	0.028	0.130	0.104
3	21	0.025	0.092	0.055	0.076	0.285	0.036	0.055	0.076
4	23	0.020	0.096	0.020	0.044	0.390	0.032	0.030	0.056
5	25	0.035	0.044	0.045	0.076	0.240	0.000	0.045	0.044
7	22	0.040	0.084	0.045	0.080	0.350	0.020	0.045	0.060
8	23	0.025	0.048	0.050	0.036	0.105	0.012	0.040	0.040
10	23	0.010	0.012	0.010	0.024	0.075	0.012	0.010	0.016
11	21	0.030	0.048	0.095	0.032	0.075	0.020	0.035	0.032
12	22	0.025	0.028	0.050	0.040	0.020	0.008	0.025	0.028
13	22	0.010	0.016	0.005	0.020	0.130	0.020	0.010	0.012
15	22	0.010	0.052	0.065	0.028	0.060	0.084	0.035	0.044
16	23	0.190	0.240	0.195	0.260	0.635	0.028	0.200	0.236
17	22	0.010	0.044	0.015	0.028	0.175	0.004	0.020	0.024
18	23	0.080	0.084	0.075	0.064	0.065	0.056	0.055	0.060
19	24	0.010	0.020	0.010	0.036	0.160	0.000	0.015	0.016
20	19	0.100	0.120	0.130	0.088	0.030	0.296	0.085	0.108
21	23	0.050	0.080	0.055	0.112	0.160	0.064	0.060	0.080
22	21	0.010	0.028	0.010	0.032	0.135	0.000	0.010	0.024
24	24	0.030	0.024	0.050	0.032	0.115	0.028	0.040	0.028
25	23	0.060	0.088	0.070	0.112	0.130	0.044	0.070	0.084
26	23	0.050	0.060	0.025	0.076	0.150	0.020	0.050	0.052
27	23	0.030	0.080	0.035	0.068	0.115	0.016	0.030	0.064
28	23	0.045	0.024	0.030	0.028	0.345	0.004	0.045	0.024
29	22	0.010	0.064	0.040	0.080	0.130	0.016	0.040	0.060
30	24	0.090	0.164	0.100	0.180	0.155	0.060	0.100	0.152
31	24	0.160	0.188	0.185	0.200	0.270	0.160	0.165	0.180
32	20	0.100	0.180	0.185	0.112	0.165	0.188	0.140	0.152
33	23	0.100	0.136	0.135	0.128	0.260	0.132	0.135	0.108
34	21	0.060	0.100	0.195	0.084	0.130	0.156	0.125	0.096
35	22	0.135	0.144	0.130	0.140	0.120	0.164	0.120	0.152
36	21	0.005	0.012	0.000	0.020	0.045	0.000	0.005	0.012
37	22	0.035	0.124	0.060	0.104	0.300	0.032	0.055	0.088
38	24	0.020	0.032	0.085	0.044	0.025	0.140	0.020	0.040
39	23	0.035	0.060	0.040	0.076	0.095	0.012	0.045	0.056
40	23	0.170	0.216	0.250	0.112	0.185	0.136	0.195	0.144
41	23	0.065	0.056	0.150	0.040	0.055	0.024	0.080	0.040
42	25	0.015	0.012	0.010	0.020	0.125	0.000	0.010	0.004
43	22	0.005	0.020	0.000	0.028	0.075	0.004	0.005	0.020
44	23	0.015	0.056	0.045	0.060	0.125	0.124	0.040	0.052
46	25	0.030	0.120	0.040	0.084	0.155	0.096	0.035	0.088
47	23	0.140	0.304	0.210	0.256	0.280	0.212	0.170	0.256
48	24	0.020	0.036	0.045	0.028	0.165	0.016	0.030	0.032
49	25	0.035	0.052	0.035	0.060	0.040	0.152	0.035	0.056
50	24	0.045	0.080	0.085	0.072	0.180	0.020	0.065	0.060
51	22	0.000	0.112	0.070	0.044	0.225	0.040	0.050	0.060
52	23	0.025	0.024	0.015	0.016	0.095	0.000	0.030	0.016
53	22	0.010	0.016	0.040	0.008	0.100	0.000	0.020	0.008
54	24	0.015	0.048	0.130	0.024	0.210	0.032	0.060	0.016
55	20	0.010	0.004	0.010	0.008	0.080	0.000	0.010	0.004
56	22	0.030	0.048	0.030	0.072	0.075	0.036	0.030	0.048
57	21	0.010	0.084	0.055	0.064	0.045	0.076	0.030	0.072
AVG	**23**								**0.062**

4.3.1 EER Comparison (Proposed Models with Past Models)

Table 4.4 shows the average EER of the proposed models compared with published studies such as (AL-Rahmani, 2014), (Al-Jarrah, 2012) and (Killourhy, K. S., & Maxion, R., 2009) all used same CMU benchmark.

Table 4.4: Models EER comparison on CMU benchmark

No.	Algorithm	EER Average	EER Std. Dev.
1	The proposed Multi-Model	***0.062***	***0.052***
2	The proposed EMM	***0.063***	***0.052***
3	The proposed Abs-Min	***0.069***	***0.056***
4	Median-Median (AL-Rahmani, 2014)	0.071	0.049
5	Medians Vector Proximity (Med-Std) (Al-Jarrah, 2012)	0.080	0.060
6	Manhattan (scaled) (Bleha, S., Slivinsky, C., & Hussien, B., 1990)	0.096	0.069
7	Nearest Neighbor (Mahalanobis) (Cho, S., Han, C., Han, D. H., & Kim, H. I., 2000)	0.100	0.064
8	Outlier Count (z-score) (Haider, S., Abbas, A., & Zaidi, A. K., A. K, 2000)	0.102	0.077
9	SVM (one-class) (Yu, E., & Cho, S., 2003)	0.102	0.065
10	Mahalanobis (Bleha, S., Slivinsky, C., & Hussien, B., 1990)	0.110	0.065
11	Mahalanobis (normed) (S Bleha, S., Slivinsky, C., & Hussien, B., 1990)	0.110	0.065
12	Manhattan (filter) (Joyce, R., & Gupta, G., 1990)	0.136	0.083
13	Manhattan (Bleha, S., Slivinsky, C., & Hussien, B., 1990)	0.153	0.092
14	Neural Network (auto-assoc.) (Cho, S., Han, C., Han, D. H., & Kim, H. I., 2000)	0.161	0.080
15	Euclidean (Duda, R. O., Hart, P. E., & Stork, D. G., 2001)	0.171	0.095
16	Euclidean (normed) (Bleha, S., Slivinsky, C., & Hussien, B., 1990)	0.215	0.119
17	Fuzzy Logic (Haider, S., Abbas, A., & Zaidi, A. K., 2000)	0.221	0.105
18	K Means (Kang, P., Hwang, S. S., & Cho, S., 2007)	0.372	0.139
19	Neural Network (standard) (Haider, S., Abbas, A., & Zaidi, A. K., 2000)	0.828	0.148

Table 4.4 shows ERR of the proposed multi-model, EMM, Abs-Min models and past models sorted from best to worst according to the EER value. In comparison with the

past models with considering the unified CMU benchmark for all models in the Table 4.4 this comparison outcome with the following:

1. The EER for the proposed Multi-model has 0.062 compared with past models, it is the best performing model based on the CMU benchmark.
2. The EER for the EMM model has 0.063 compared with the past models, it best than past models' EER and take the second place in the comparison table according to it EER.
3. The EER for the proposed Abs-Min is 0.069 compared with the past models; it best than past models' EER and take the third place in the comparison table according to it EER.
4. The EER of Multi-Model has 0.062 compared with proposed two single models before it combined, the lower EER was 0.063 for EMM model and the AbsMin model had EER 0.069, which make it on the top of past and current models according to the error metric EER.
5. The result of comparison in Table 4.5 leads this study to choose EMM, Abs-Min and Med-Std for the proposed multi-model.

Model No.4 in Table 4.4 (Median-Median) ignored because the EMM model is an enhancement of this, and the approach that used to find the DTM used in the multi-model by the EMM.

4.3.2 EER Evaluation on MEU Dataset

Table 4.5 and table 4.6 show the error metrics of FAR, FRR and ERR, as well as the pass-mark for the each of the 20 subjects in the collected MEU dataset and detailed as the following:

- **Genuine Test:** In the MEU dataset, there are 30 genuine user's testing attempts against a template generated by 30 training attempts of the same user.
- **Impostor Test**: In the MEU dataset, impostors' login data (5 from each other user) are evaluated against the training templates of each genuine user (for each of the 20 users, the other 19 users are considered impostors) that generate 95 impostor's test for each subject in MEU dataset.

Table 4.5: EER analysis of EMM model on MEU dataset (31 features)

Subject	**Pass Mark (<= 31)**	**Genuine Test (30)**		**Impostor Test (95)**		**FAR**	**FRR**	**EER**
		TA	**FR**	**TR**	**FA**			
1	17	28	2	85	10	0.105	0.067	0.086
2	19	27	3	84	11	0.116	0.100	0.108
3	19	28	2	84	11	0.116	0.067	0.091
4	23	26	4	82	13	0.137	0.133	0.135
5	24	29	1	93	2	0.021	0.033	0.027
6	22	27	3	89	6	0.063	0.100	0.082
7	21	29	1	93	2	0.021	0.033	0.027
8	24	29	1	94	1	0.011	0.033	0.022
10	20	30	0	94	1	0.011	0.000	0.005
11	21	27	3	83	12	0.126	0.100	0.113
12	21	29	1	91	4	0.042	0.033	0.038
13	19	28	2	91	4	0.042	0.067	0.054
14	23	26	4	80	15	0.158	0.133	0.146
16	23	29	1	87	8	0.084	0.033	0.059
24	22	28	2	88	7	0.074	0.067	0.070
18	19	26	4	78	17	0.179	0.133	0.156
20	24	30	0	95	0	0.000	0.000	0.000
21	21	29	1	94	1	0.011	0.033	0.022
22	21	29	1	93	2	0.021	0.033	0.027
23	22	29	1	91	4	0.042	0.033	0.038
AVG	**21**	28	2	88	7	0.069	0.062	**0.065**

Table 4.5 shows results of the EER metric of the proposed EMM model. In comparison with results the CMU benchmark for the same proposed EMM model, and taking into

consideration the different dataset sizes, the comparison is summarized as in the following:

1. The EER metric in the MEU dataset has an average of 0.065, compared to 0.063 in the CMU benchmark.

2. The Pass-Mark in the MEU dataset has an average of 22, compared to 23 in the CMU benchmark.

3. The difference in dataset sizes and number of repetitions for training and testing did not have an effect on the results.

Table 4.6: EER analysis of Abs-Min model on MEU Dataset (31 features)

Subject	**Pass Mark (<= 31)**	**Genuine Test (30)**		**Impostor Test (95)**		**FAR**	**FRR**	**EER**
		TA	FR	TR	FA			
1	18	26	4	88	7	0.074	0.133	0.104
2	20	26	4	81	14	0.147	0.133	0.140
3	20	26	4	86	9	0.095	0.133	0.114
4	24	27	3	85	10	0.105	0.100	0.103
5	25	29	1	93	2	0.021	0.033	0.027
6	22	27	3	84	11	0.116	0.100	0.108
7	22	29	1	92	3	0.032	0.033	0.032
8	25	30	0	92	3	0.032	0.000	0.016
10	21	29	1	94	1	0.011	0.033	0.022
11	22	26	4	86	9	0.095	0.133	0.114
12	22	29	1	90	5	0.053	0.033	0.043
13	20	26	4	85	10	0.105	0.133	0.119
14	24	26	4	81	14	0.147	0.133	0.140
16	24	28	2	88	7	0.074	0.067	0.070
24	23	29	1	91	4	0.042	0.033	0.038
18	20	25	5	80	15	0.158	0.167	0.162
20	24	30	0	95	0	0.000	0.000	0.000
21	22	29	1	94	1	0.011	0.033	0.022
22	21	29	1	93	2	0.021	0.033	0.027
23	24	29	1	93	2	0.021	0.033	0.027
AVG	**22**	28	2	89	6	0.068	0.075	**0.071**

Table 4.6 shows the EER results of the proposed Abs-Min model. In comparison with the CMU benchmark for the same proposed Abs-Min model, and taking into consideration the difference in the two dataset sizes, the comparison is summarized as in the following:

1. The EER metric has an average of 0.071, compared to 0.069 in the CMU benchmark.

2. The pass-mark has an average of 22, compared to 23 in the same Abs-Min anomaly detector model on CMU.

3. The difference in datasets size and the number of repetitions of training and testing did not have an effect on the results.

4.4 Miss-Rate Evaluation and Comparison on MEU and CMU Datasets

The Miss-Rate analysis uses Hold and DD features only, by excluding the UD feature,

In order to be consistent with this type of analysis that was carried out on the CMU dataset (Killourhy, K. S., 2012), which compared 12 models using 21 features of Hold and DD, for the same password (Killourhy, K. S., 2012).

This analysis fixes the False-Alarm (FRR) at 5% and measures the Miss-Rate at that point.

4.4.1 Miss-Rate Evaluation on CMU Benchmark

Table 4.7 shows the Miss-Rate of the proposed EMM model.

Table 4.8 shows the Miss-Rate of the proposed Abs-Min model. Tables 4.8 and 4.9 show the error metrics of FAR and FRR, as well as the Pass-Mark for the each of the 51 subjects in the published CMU benchmark and detailed as the following:

Subject: Subject number of the subject according to the CMU benchmark.

- **Pass Mark (<= 21):** Pass mark value at which the FRR (false-alarm) is 5% (or as close to 5% as possible), which is the fixed FRR for evaluation of the FAR metric.
- **Genuine Test:** It is the test when a genuine user's login data from the testing phase is evaluated against the same user's training template. In the CMU

benchmark, there are 200 genuine user's testing attempts against a template generated by 200 training attempts of the same user.

- **Impostor Test**: It is the test when an impostor's login data from the testing phase is evaluated against the training template of a particular genuine user's template. In the CMU benchmark, 250 impostors' login data (5 from each other user) are evaluated against the training templates of each genuine user (for each of the 51 users, the other 50 users are considered impostors).
- **FAR (Miss-Rate):** False acceptance rate = FA / 250, at the 5% point of FRR.
- **FRR (False-Alarm):** False rejection rate = FR / 200, which is fixed around 5% by tuning the pass-mark

Comparison of the results in the Table 4.7 and Table 4.8 with past studies is presented in Table 4.9.

Table 4.7: Miss-Rate analysis of EMM model on CMU benchmark (21 features)

Subject No.	Pass Mark (<=21)	Genuine Test (200)		Impostor Test (250)		FAR (Miss-Rate)	FRR (False-Alarm)
		TA	FR	TR	FA		
2	13	193	7	98	152	60.8%	3.5%
3	16	186	14	181	69	27.6%	7.0%
4	18	185	15	225	25	10.0%	7.5%
5	18	197	3	238	12	4.8%	1.5%
7	17	186	14	224	26	10.4%	7.0%
8	18	189	11	235	15	6.0%	5.5%
10	18	193	7	248	2	0.8%	3.5%
11	17	192	8	234	16	6.4%	4.0%
12	18	187	13	236	14	5.6%	6.5%
13	17	189	11	242	8	3.2%	5.5%
15	17	195	5	231	19	7.6%	2.5%
16	16	186	14	163	87	34.8%	7.0%
17	18	192	8	248	2	0.8%	4.0%
18	17	190	10	204	46	18.4%	5.0%
19	18	196	4	246	4	1.6%	2.0%
20	14	191	9	140	110	44.0%	4.5%
21	17	187	13	221	29	11.6%	6.5%
22	18	185	15	250	0	0.0%	7.5%
24	18	189	11	244	6	2.4%	5.5%
25	17	190	10	226	24	9.6%	5.0%
26	18	188	12	242	8	3.2%	6.0%
27	18	187	13	236	14	5.6%	6.5%
28	17	192	8	243	7	2.8%	4.0%
29	17	193	7	236	14	5.6%	3.5%
30	17	188	12	178	72	28.8%	6.0%
31	15	193	7	105	145	58.0%	3.5%
32	13	193	7	118	132	52.8%	3.5%
33	17	188	12	218	32	12.8%	6.0%
34	15	193	7	172	78	31.2%	3.5%
35	14	187	13	158	92	36.8%	6.5%
36	18	187	13	250	0	0.0%	6.5%
37	17	190	10	226	24	9.6%	5.0%
38	18	194	6	237	13	5.2%	3.0%
39	17	192	8	231	19	7.6%	4.0%
40	16	186	14	137	113	45.2%	7.0%
41	17	188	12	193	57	22.8%	6.0%
42	19	192	8	248	2	0.8%	4.0%
43	18	192	8	250	0	0.0%	4.0%
44	18	185	15	243	7	2.8%	7.5%
46	18	192	8	197	53	21.2%	4.0%
47	15	188	12	108	142	56.8%	6.0%
48	19	191	9	241	9	3.6%	4.5%
49	18	194	6	241	9	3.6%	3.0%
50	18	188	12	224	26	10.4%	6.0%
51	18	186	14	231	19	7.6%	7.0%
52	18	196	4	247	3	1.2%	2.0%
53	18	193	7	248	2	0.8%	3.5%
54	18	194	6	223	27	10.8%	3.0%
55	17	192	8	248	2	0.8%	4.0%
56	17	190	10	224	26	10.4%	5.0%
57	17	187	13	227	23	9.2%	6.5%
AVG	17	190	10	214	36	**14.4%**	**4.9%**

Table 4.8: Miss-Rate analysis of Abs-Min model on CMU benchmark (21 features)

Subject No.	Pass Mark (<= 21)	Genuine Test (200)		Impostor (250)		Miss-rate FAR	False-alarm FRR
		TA	FR	TR	FA		
2	15	192	8	132	118	47.2%	4.0%
3	14	194	6	124	126	50.4%	3.0%
4	18	189	11	228	22	8.8%	5.5%
5	18	188	12	225	25	10.0%	6.0%
7	16	192	8	184	66	26.4%	4.0%
8	17	193	7	228	22	8.8%	3.5%
10	18	194	6	242	8	3.2%	3.0%
11	15	189	11	198	52	20.8%	5.5%
12	17	188	12	225	25	10.0%	6.0%
13	17	189	11	241	9	3.6%	5.5%
15	15	195	5	218	32	12.8%	2.5%
16	16	190	10	151	99	39.6%	5.0%
17	17	187	13	244	6	2.4%	6.5%
18	16	188	12	166	84	33.6%	6.0%
19	19	185	15	249	1	0.4%	7.5%
20	14	189	11	148	102	40.8%	5.5%
21	16	193	7	158	92	36.8%	3.5%
22	17	193	7	247	3	1.2%	3.5%
24	17	195	5	239	11	4.4%	2.5%
25	17	193	7	209	41	16.4%	3.5%
26	19	190	10	245	5	2.0%	5.0%
27	18	190	10	226	24	9.6%	5.0%
28	18	189	11	244	6	2.4%	5.5%
29	17	187	13	233	17	6.8%	6.5%
30	17	193	7	165	85	34.0%	3.5%
31	16	188	12	127	123	49.2%	6.0%
32	12	188	12	97	153	61.2%	6.0%
33	16	187	13	178	72	28.8%	6.5%
34	14	189	11	150	100	40.0%	5.5%
35	14	193	7	143	107	42.8%	3.5%
36	18	188	12	250	0	0.0%	6.0%
37	17	192	8	208	42	16.8%	4.0%
38	16	192	8	178	72	28.8%	4.0%
39	18	190	10	233	17	6.8%	5.0%
40	15	193	7	108	142	56.8%	3.5%
41	15	193	7	107	143	57.2%	3.5%
42	19	193	7	246	4	1.6%	3.5%
43	18	195	5	248	2	0.8%	2.5%
44	16	193	7	224	26	10.4%	3.5%
46	18	194	6	195	55	22.0%	3.0%
47	14	190	10	101	149	59.6%	5.0%
48	19	189	11	232	18	7.2%	5.5%
49	18	194	6	235	15	6.0%	3.0%
50	17	190	10	196	54	21.6%	5.0%
51	14	194	6	190	60	24.0%	3.0%
52	18	189	11	249	1	0.4%	5.5%
53	16	194	6	245	5	2.0%	3.0%
54	16	194	6	185	65	26.0%	3.0%
55	16	189	11	238	12	4.8%	5.5%
56	17	192	8	202	48	19.2%	4.0%
57	16	188	12	217	33	13.2%	6.0%
AVG	16.5	191	9	199	51	**20.4%**	**4.5%**

4.5 Miss-Rate Comparison (Proposed Models with Past Models)

Table 4.9 shows the Miss-Rate when the false-alarm is fixed at 5% (or as close as possible) for the proposed EMM and Abs-Min models, and the other models that were studied by CMU, and sorted in ascending order according to its Miss-Rate values.

Table 4.9: Miss-Rate models comparison on CMU benchmark

No	Classifier	False-Alarm Rate(FRR)	Miss Rate(FAR)
1	EMM	***4.9%***	***14.4%***
2	Abs-Min	***4.5%***	***20.4%***
3	Scaled Manhattan	5.0%	23.6%
4	KNN	5.0%	29.8%
5	SVM	5.0%	30.2%
6	Outlier Count	2.9%	31.7%
7	Mahalanobis KNN	5.0%	33.7%
8	K-Means	5.0%	35.0%
9	Mahalanobis	5.0%	39.1%
10	Manhattan	5.0%	41.8%
11	Auto AssocNNet	5.0%	56.3%
12	Euclidean	5.0%	61.0%

The outcome of this comparison is summarized in the following:

1. The Miss-Rate of the EMM model is 14.4% and compared with the other models in the CMU study it is the top performer on the Miss-Rate metric.
2. The Miss-Rate of the Abs-Min model has 20.4%, and in comparison with the other models in the CMU study it is the second best performer on the Miss-Rate metric, followed by the Scaled Manhattan which had 23.6% of Miss-Rate and was the top performer in the CMU study.

4.6 Miss-Rate Evaluation on MEU Benchmark

Table 4.10 and table 4.11 show the error metrics of FAR, FRR and ERR, as well as the pass-mark for the each of the 20 subjects in the collected MEU dataset and detailed as the following:

- **Subject:** Subject number of the subject according to the MEU dataset.
- **Pass Mark (<= 21):** Pass mark value at which the FRR (false-alarm) is 5% (or as close to 5% as possible), which is the fixed FRR for evaluation of the FAR metric.
- **Genuine Test:** It is the test when a genuine user's login data from the testing phase is evaluated against the same user's training template. In the CMU benchmark, there are 30 genuine user's testing attempts against a template generated by 30 training attempts of the same user.

- **Impostor Test**: It is the test when an impostor's login data from the testing phase is evaluated against the training template of a particular genuine user's template. In the MEU dataset, 95 impostors' login data (5 from each other user) are evaluated against the training templates of each genuine user (for each of the 20 users, the other 19 users are considered impostors).
- **FAR (Miss-Rate):** False acceptance rate = FA / 95, at the 5% point of FRR.
- **FRR (False-Alarm):** False rejection rate = FR / 30, which is fixed around 5% by tuning the pass-mark.

Table 4.10: Miss-Rate analysis of EMM model on MEU dataset

Subject	**Pass Mark 31**	**Genuine VS Genuine 30**		**Impostor VS Impostor 95**		**Miss-rate FAR**	**False-alarm FRR**
		TA	FR	TR	FA		
1	12	81	14	29	1	14.7%	3.3%
2	12	77	18	29	1	18.9%	3.3%
3	14	87	8	28	2	8.4%	6.7%
4	16	82	13	29	1	13.7%	3.3%
5	17	94	1	29	1	1.1%	3.3%
6	16	87	8	28	2	8.4%	6.7%
7	15	92	3	29	1	3.2%	3.3%
8	17	93	2	28	2	2.1%	6.7%
10	15	95	0	29	1	0.0%	3.3%
11	15	80	15	29	1	15.8%	3.3%
12	16	94	1	29	1	1.1%	3.3%
13	11	59	36	28	2	37.9%	6.7%
14	14	57	38	28	2	40.0%	6.7%
16	17	88	7	29	1	7.4%	3.3%
24	17	91	4	28	2	4.2%	6.7%
18	11	51	44	28	2	46.3%	6.7%
20	18	95	0	29	1	0.0%	3.3%
21	16	94	1	28	2	1.1%	6.7%
22	16	94	1	29	1	1.1%	3.3%
23	12	81	14	29	1	14.7%	3.3%
AVG	**15**	84	11	29	1	**11.9%**	**4.7%**

Table 4.10 shows results of the Miss-Rate (FAR) metric of the proposed EMM model using the MEU dataset. In comparison with results the CMU benchmark for the same proposed EMM model, and taking into consideration the different dataset sizes, the comparison is summarized as in the following:

1. The Miss-Rate has an average of 11.9% at the FRR 4.7% fixed point, compared to 14.4% in the CMU benchmark that is shown in Table 4.7.

2. The Pass-Mark has an average of 15, compared to 17 in the CMU benchmark results as shown in Table 4.7.

3. The difference in dataset sizes and number of repetitions for training and testing did not have an effect on the results.

Table 4.11: Miss-Rate analysis of Abs-min model on MEU dataset

Subject	**Pass Mark 31**	**Genuine Test (30)**		**Impostor Test (95)**		**Miss-rate FAR**	**False-alarm FRR**
		TA	FR	TR	FA		
1	13	82	13	28	2	13.7%	6.7%
2	12	72	23	29	1	24.2%	3.3%
3	14	86	9	28	2	9.5%	6.7%
4	16	81	14	29	1	14.7%	3.3%
5	18	94	1	27	3	1.1%	10.0%
6	16	78	17	28	2	17.9%	6.7%
7	16	92	3	27	3	3.2%	10.0%
8	18	94	1	29	1	1.1%	3.3%
10	16	95	0	29	1	0.0%	3.3%
11	16	84	11	29	1	11.6%	3.3%
12	16	92	3	29	1	3.2%	3.3%
13	10	46	49	28	2	51.6%	6.7%
14	16	70	25	28	2	26.3%	6.7%
16	17	80	15	29	1	15.8%	3.3%
24	18	93	2	28	2	2.1%	6.7%
18	11	45	50	30	0	52.6%	0.0%
20	18	95	0	29	1	0.0%	3.3%
21	16	93	2	29	1	2.1%	3.3%
22	16	94	1	29	1	1.1%	3.3%
23	13	82	13	28	2	13.7%	6.7%
AVG	**15**	82	13	29	1	**13.2%**	**4.9%**

Table 4.11 shows results of the Miss-Rate (FAR) metric of the proposed EMM model using the MEU dataset. In comparison with results the CMU benchmark for the same proposed EMM model, and taking into consideration the different dataset sizes, the comparison is summarized as in the following:

1. The Miss-Rate has an average of 13.2% at the FRR 4.9% fixed point, compared to 20.4% in the CMU benchmark that is shown in Table 4.8.

2. The Pass-Mark has an average of 15, compared to 16.5 in the CMU benchmark results as shown in Table 4.8.

Chapter 5

Conclusion and Future Work

5.1 Conclusions

The work in this thesis has focused on investigating the enhancing of KD based authentication through an empirical study of a public benchmark dataset.

The following conclusions are made based on the reported work:

1. The proposed multi-model has given the lowest EER, compared to previous models using the same benchmark.
2. The proposed Enhanced Med-Med (EMM) has given a lower EER, compared to previous models using the same benchmark.
3. An alternative anomaly detector model (Abs-Min) was formulated, which showed good EER error, and can be used with the EMM model in a multi-model,
4. The power of anomaly detection can be enhanced through the combining of several good performing authentication models into a multi-mode.

5.2 Future Work

Based on the results and experience obtained during work on this thesis, the following suggestions can be made for future work:

1. Combine the KD method with other authentication modalities for a multi-modal authentication, such as using mixed text and voice input.
2. Encrypt the typing profile templates in the database to prevent attempts to mimic the typing behavior of an important person through a computerized or a robotic attack.
3. Investigate the proposed multi-model in continuous KD authentication.
4. Investigate the enhancement of the two-level authentication approach (password and one-time-password), by adding KD, to strengthen access control when the one-time-password (OTP) is also compromised through a resident Malware.
5. Investigate extending the multi-model system to include more single models that add more anomaly detection power.
6. Investigate other statistical functions to measure the DTM in the median-based anomaly detection models.
7. Do further data collection of typing profiles, to establish datasets that reflect different users typing skills, backgrounds, and education levels.

References

References

Abernethy, M., & Rai, S. (2012). Applying feature selection to reduce variability in keystroke dynamics data for authentication systems.

Al-Jarrah, M. M. (2012). An anomaly detector for keystroke dynamics based on medians vector proximity. *Journal of Emerging Trends in Computing and Information Sciences*, *3*(6), 988-993..

Al-Jarrah, M. M. (2013). A multifactor authentication scheme using keystroke Dynamics and Two-Part Passwords. *International Journal of Academic Research*, *5*(3).

AL-Rahmani, A. O. (2014). *An Enhanced Classifier for Authentication in Keystroke Dynamics Using Experimental Data.* Amman-Jordan: MEU(master thesis).

Antal, M., Szabó, L. Z., & László, I. (2015). Keystroke dynamics on android platform. *Procedia Technology*, *19*, 820-826.

Araújo, L. C., Sucupira, L. H., Lizarraga, M. G., Ling, L. L., & Yabu-Uti, J. B. T. (2005). User authentication through typing biometrics features. *Signal Processing, IEEE Transactions on*, *53*(2), 851-855.

Bleha, S. A., Knopp, J., & Obaidat, M. S. (1992, March). Performance of the perceptron algorithm for the classification of computer users. In *Proceedings of the 1992 ACM/SIGAPP symposium on Applied computing: technological challenges of the 1990's* (pp. 863-866). ACM.

Bleha, S., Slivinsky, C., & Hussien, B. (1990). Computer-access security systems using keystroke dynamics. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, *12*(12), 1217-1222.

Cho, S., Han, C., Han, D. H., & Kim, H. I. (2000). Web-based keystroke dynamics identity verification using neural network. *Journal of organizational computing and electronic commerce*, *10*(4), 295-307.

Duda, R. O., Hart, P. E., & Stork, D. G. (2001). Pattern classification.*International Journal of Computational Intelligence and*

Applications, *1*, 335-339.

e_bias_detail.php?BiasID=3. (2016, 1 2). Retrieved from face-tek: http://www.face-tek.com/e_bias_detail.php?BiasID=3

Forsen, G. E., Nelson, M. R., & Staron Jr, R. J. (1977). *Personal Attributes Authentication Techniques* (No. PAR-77-21). PATTERN ANALYSIS AND RECOGNITION CORP ROME NY.

Gaines, R. S., Lisowski, W., Press, S. J., & Shapiro, N. (1980).*Authentication by keystroke timing: Some preliminary results* (No. RAND-R-2526-NSF). RAND CORP SANTA MONICA CA.

Giot, R., El-Abed, M., & Rosenberger, C. (2009, September). Greyc keystroke: a benchmark for keystroke dynamics biometric systems. In*Biometrics: Theory, Applications, and Systems, 2009. BTAS'09. IEEE 3rd International Conference on* (pp. 1-6). IEEE.

Giot, R., El-Abed, M., Hemery, B., & Rosenberger, C. (2011). Unconstrained keystroke dynamics authentication with shared secret. *computers & security*, *30*(6), 427-445..

Haider, S., Abbas, A., & Zaidi, A. K. (2000). A multi-technique approach for user identification through keystroke dynamics. In *Systems, Man, and Cybernetics, 2000 IEEE International Conference on* (Vol. 2, pp. 1336-1341). IEEE.

Hu, J., Gingrich, D., & Sentosa, A. (2008, May). A k-nearest neighbor approach for user authentication through biometric keystroke dynamics. In *Communications, 2008. ICC'08. IEEE International Conference on* (pp. 1556-1560). IEEE.

Idrus, S. Z. (2015). Soft Biometrics for Keystroke Dynamics. *Computer Vision and Pattern Recognition.Universite de Caen Basse-Normandie*(PhD thesis).

Joyce, R., & Gupta, G. (1990). Identity authentication based on keystroke latencies. *Communications of the ACM*, *33*(2), 168-176..

Kang, P., Hwang, S. S., & Cho, S. (2007). Continual retraining of keystroke dynamics based authenticator. In *Advances in Biometrics* (pp. 1203-1211). Springer Berlin Heidelberg.

Karatzouni, S., & Clarke, N. (2007). Keystroke analysis for thumb-based keyboards on mobile devices. In *New approaches for security, privacy and trust in complex environments* Vol 232(pp. 253-263). Springer US.

Killourhy, K. S. (2012). *A scientific understanding of keystroke dynamics* (No. CMU-CS-12-100). CARNEGIE INST OF TECH PITTSBURGH PA DEPT OF COMPUTER SCIENCE.

Killourhy, K. S., & Maxion, R. (2009, June). Comparing anomaly-detection algorithms for keystroke dynamics. In *Dependable Systems & Networks, 2009. DSN'09. IEEE/IFIP International Conference on* (pp. 125-134). IEEE.

Killourhy, K. S., & Maxion, R. A. (2012, July). Free vs. transcribed text for keystroke-dynamics evaluations. In *Proceedings of the 2012 Workshop on Learning from Authoritative Security Experiment Results* (pp. 1-8). ACM.

Monrose, F., & Rubin, A. (1997, April). Authentication via keystroke dynamics. In *Proceedings of the 4th ACM conference on Computer and communications security* (pp. 48-56). ACM.

Monrose, F., & Rubin, A. D. (2000). Keystroke dynamics as a biometric for authentication. *Future Generation computer systems*, *16*(4), 351-359..

Roy, S., Roy, U., & Sinha, D. D. (2014). Enhanced knowledge-based user authentication technique via keystroke dynamics. *Int. J. Eng. Sci. Invention (IJESI)*, *3*(9), 41-48.

Ryan, S. A. (2014). *mobile keystroke dynamics: assessment and implementation california:* California State University (master thesis)

Sedenka, J., Balagani, K. S., Phoha, V., & Gasti, P. (2014, September). Privacy-preserving population-enhanced biometric key generation from free-text keystroke dynamics. In *Biometrics (IJCB), 2014 IEEE International Joint Conference on* (pp. 1-8). IEEE.

Singh, K., & Kaur, H. (2013). Rule Based Approach for Keystroke Biometrics to identify authenticated user. *International Journal of Computer Science and Information Security*, *11*(7), 6.

Spillane, R. (1975). Keyboard apparatus for personal identification. *IBM Technical Disclosure Bulletin*, *17*(3346), 3346.

Syed, Z. A. (2014). *Keystroke and Touch-dynamics Based Authentication for Desktop and Mobile Devices* (Doctoral dissertation, WEST VIRGINIA UNIVERSITY).

Yu, E., & Cho, S. (2003, July). GA-SVM wrapper approach for feature subset selection in keystroke dynamics identity verification. In *Neural Networks, 2003. Proceedings of the International Joint Conference on* (Vol. 3, pp. 2253-2257). IEEE.

Zhao, Y. (2006, December). Learning user keystroke patterns for authentication. In *Proceeding of World Academy of Science, Engineering and Technology* (Vol. 14, pp. 65-70).

Zhong, Y., Deng, Y., & Jain, A. K. (2012, June). Keystroke dynamics for user authentication. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on* (pp. 117-123). IEEE.