UNIVERSIDAD
POLITÉCNICA DE MADRID

FACULTAD DE INFORMÁTICA

MÁSTER UNIVERSITARIO EN SOFTWARE Y SISTEMAS

Enhancing Online Banking Authentication Using Keystroke Dynamics

Master Thesis

Author: María del Carmen Sánchez Medrano
Directors: Manuel Carro
Juan Caballero

July 2017

Master Thesis

MUSS Master thesis from UPM done by:

Author's Surname and name: Sánchez Medrano, María del Carmen
Title: Enhancing Online Banking Authentication Using Keystroke Dynamics
Date: July 2017
Directors: Manuel Carro
Juan Caballero

Abstract

The most common method for banks to authenticate users is through a user identifier and password. Unfortunately, this is a method of easy impersonation, because although many banks control brute force attacks by blocking the account after a maximum number of failed login attempts, credentials may be stolen. A big challenge for banks is to identify whether the user is or not the one he is supposed to be. The security measures based on biometrics are the ones that have given better results against this type of attacks. But as a drawback, most of these methods are very expensive to implement and their usability is low since they require special hardware. There are other types of biometric methods based on patterns, like keystroke dynamics. Each user has their own typing pattern which is very difficult to replicate. The different speeds between typing keys or the number of uses of a control key can be the difference between one person to the other.

In this thesis, we study how to improve bank authentication using keystroke dynamics. In order to achieve this objective, we had to perform data acquisition, data preprocessing and feature extraction processes. In the data acquisition process, a webpage and a Chrome Extension were developed to help with retrieve data for the data collection process. We had to carry out a study about the different kinds of authentication processes that banks have and then we had to identify the several cases of use in terms of keystroke dynamics in order to know which events we were going to give support in our development. The dataset we got, was made by 50 users who took the test 450 times spread over two weeks. After getting the datasets, we proceeded to make the data preprocessing and feature extraction processes. Before calculating the features, we had to separate the correct characters and the deletions and ignored those events that were not supported. After that, we built the feature vector files, having into account special events like the usage of Shift key.

Finally, an study using machine learning techniques was done. The tool used was WEKA with which we run some very well known classification methods such as C4.5 tree, Random Forest, SVM or K nearest neighbor. The accuracy has been measured using false acceptance rate (FAR), i.e, the ratio of incorrect accepted users, and in false rejection rate (FRR), i.e, the ratio of incorrect rejection user. The results have been

satisfactory using most of the methods. The FAR is below 1%, while the FRR could be reduced to 3% in some cases.

Resumen

El método más usado por los bancos para autentificar a los usuarios es el uso de un identificador de usuario y una contraseña. Lamentablemente, un usuario puede ser fácilmente suplantado ya que aunque muchos bancos controlan los ataques de fuerza bruta bloqueando al usuario que se equivoque un número máximo de veces, las credenciales de un usuario pueden ser robadas igualmente. Un desafío para las entidades bancarias es identificar si el usuario que se identifica en sus sistemas es quien dice ser. Las medidas de seguridad basadas en biometría son las que han dado mejores resultados en contra de este tipo de ataques. Pero como desventaja, la mayoría de estos métodos son muy caros de implementar y la usabilidad de las mismas se ve cuestionada ya que requiere un hardware específico. Existe otro tipo de métodos biométricos basados en patrones. Cuando una persona escribe tiene un patrón característico difícil de replicar. La diferencia de velocidades entre dos teclas consecutivas o el número de usos de teclas de control pueden utilizarse para diferenciar usuarios.

En esta tesis, nos vamos a centrar en estudiar cuan útil es el uso de los patrones de tecleo para identificar al usuario en sistemas bancarios. Este objetivo conlleva, por una parte, realizar una recolección de datos, posteriormente prcesarlos y someterlos a un proceso de extracción de características. Para la recolección de datos se ha desarrollado una serie de herramientas: una plataforma web y una extensión para Chrome. También se ha realizado un estudio preliminar acerca de los diferentes procesos de autenticación que utilizan las entidades bancarias, para luego identificar los casos de uso que contemplaremos en el reconocimiento de patrones de escritura. El dataset obtenido está compuesto por un total de 50 usuarios que han tenido que realizar la prueba 450 veces repartidas en dos semanas. Ya con la posesión de los datos, se ha procedido a la limpieza de los mismos y a la extracción de características. La limpieza de los datos consistió en filtrar los caracteres correctos y tener en cuenta los borrados, así como ignorar eventos no soportados. Tras este proceso, construimos los vectores de características, que además tienen en cuenta eventos como e uso de teclas Shift.

Por último, se ha realizado un estudio utilizando técnicas de machine learning. La herramienta utilizada ha sido WEKA, con métodos de clasificación muy conocidos como el árbol C4.5, Random Forest, SVM o los K vecinos más próximos. La extactitud de los datos se han medido en false acceptance rate (FAR) es decir, el ratio de usuarios

aceptados érroneamente y false rejection rate (FRR), es decir el ratio de usuarios rechazados erróneamente. Los resultados han sido satisfactorios puesto que en la mayoría de los métodos, el FAR está por debajo del 1%. Así mismo, el FRR pudo ser reducido al 3% en algunos casos.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Banks are always looking for ways of enhancing security, especially in a time where they are mostly focused on e-banking processes. One of their major problems is user authentication and how to detect identity theft. The most common method that banks use for identifying their users are knowledge-based authentication methods, such as passwords or PINs. But what happens if the client is attacked by a phishing attack? How could the bank know that someone else is supplanting their client? Knowledge-based authentication methods are vulnerable to a number security attacks like brute force, shoulder surfing and smudge attacks.

The usage of biometrics authentication to try to enhance security is one of the most common alternatives that banks are trying to incorporate in their authentication processes. Unfortunately, most of these methods are quite expensive and hard to implement and include in daily life. Biometrics authentication tries to identify a person based on physiological,(e.g. fingerprint, iris pattern) or behavioral characteristics (e.g. signature, voice or keystroke dynamics). In order to find a balance between usability and security, the use of keystroke dynamics methods in conjunction with password authentication method could help enhance security without much effort in an almost transparent process for the user. Keystroke dynamics is the process of analyzing the way users type at a terminal by monitoring the keyboard inputs and attempts to identify them based on habitual rhythm patterns in the way they type' (Monrose, 1999). Although keystroke dynamics is seen as a relatively a new technology, it was used by

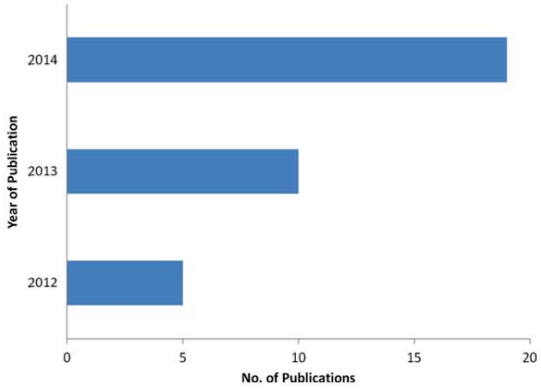


Figure 1.1: The increasing trend of research. Published on paper [27]

the US military to distinguish the ally from the enemy for Morse code messages during World War II. It is one of the cheapest methods and quite easy to implement in almost all the systems. Owing to the potential of touch dynamics biometrics, there have been increasing research efforts in this topic area, as shown in Fig. 1.1.

As explained before, enhancing security in bank authentication process is a critical issue. The main objective of this Master Thesis is to try to build a solution based in machine learning techniques using timing features (keystroke dynamics) and contextual features. As it is shown in Figure 1.2, we have to developed all the phases that happen in an authentication process.

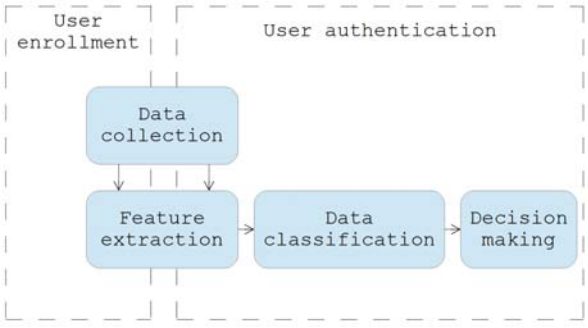


Figure 1.2: General keystroke dynamic process

1.0.1 Authentication Process

The authentication process can be split in two phases: user enrollment where data is acquired and stored, and user authentication where user login data is compared against the models. In the user enrollment phase we had to develop tools for data acquisition, so in the whole project we built a website, a Chrome extension and an Android mobile application. In order to get the data collection we had to face some challenges such us how to store the data, identify important key events and deletion events. After the data collection we had to clean the dataset and construct the features vector. We had to manage deletion keys and separate the events depending on their sequence. Once we had our feature vectors built, we proceeded to make an study using the raw data, the feature vectors and WEKA[24]. We run popular machine learning classification algorithm to compare their results.

With the work done in this master thesis we are able to demonstrate the benefit of using keystroke dynamics features to improve bank user's authentication. to authenticate the users. The reminder of this project is structured as follows. In Chapter 2, we explain what was the state of the art before starting the project and the results obtained. In Chapter 3, we explain the bank study done in order to know what kind of authentication processes banks have. We also summarize the results that we obtained. After this we proceed to explain the processes of user enrollment phase. In Chapter 4, we explain the data collection process. It is here where we talk about how to collect the data, how to store it and the tools developed for achieve it. In Chapter 5, we talk about the features and how to extract them. In Chapter 6 we proceed to present the dataset used and the results obtained by studying the raw data and by executing machine learning classification algorithms. Finally, in Chapter 7 we finish with the conclusions and future work.

Chapter 2

Related Work

Digital signatures are generated every time a human interacts with a keyboard or a mobile phone. These signatures must be unique for the individual. As it can be observed in Figure 2.1, keystroke dynamics have evolved during the years.

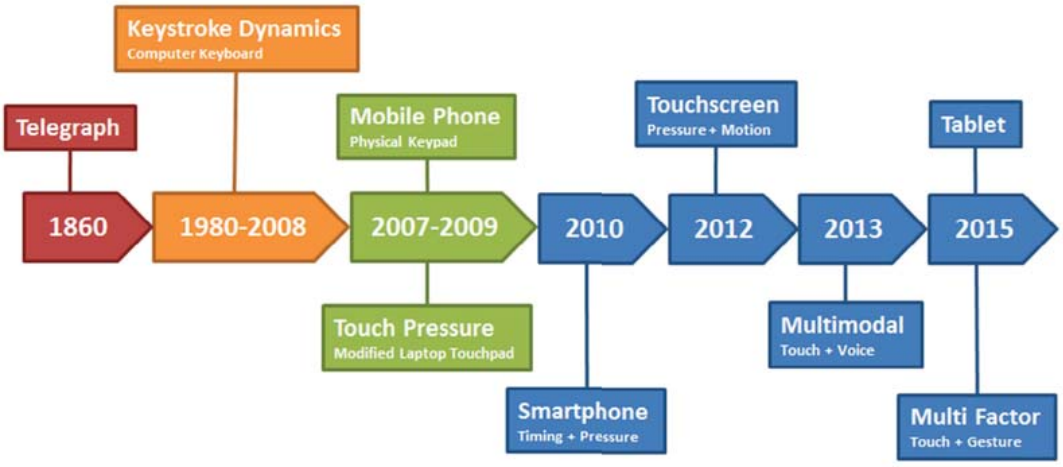


Figure 2.1: Evolution of keystroke dynamics. Published on paper [27]

2.1 Characteristics

Reading through the different papers, we can resume that the characteristics that motivated the study of keystroke dynamics were:

- **Distinctiveness**. Touch dynamics patterns are able to generate multidimensional features, such as timing, spatial or motion features. This features are hard to replicate so they can be used for authentication.
- **Security improvement**. Including touch dynamics methods could increase the security level of the most widely accepted authentication mode: passwords.
- **Continuous Monitoring**. The use of this kind of biometric methods allows to continuously monitor the user allowing user's reauthentication.
- **Revocability**. When a a touch dynamic template is compromised, a new touch dynamic template can be easily generated.
- **Non-dependency**.The feature acquisition of keystroke biometrics is less sensitive to environments factors such us noise or light.
- **Transparency**. Touch dynamics authentication system require little or no additional interventions from the user.
- **Cost-Effectiveness**. In contrast of the other biometrics methods that sometimes requite the use of specialized hardware, keystroke dynamics and touch dynamics only require a program that records them.

2.2 Related Work

There are a number of papers that talks about keystroke dynamics and contextual features to enhance security. We are going to distinguish with the one that focus on mobile authentication and the one that focus on web.

2.2.1 Web Authentication Studies

For web authentication methods we can find studies that built neural networks models and others not. In the first paper we are talking about a method based on a Support Vector Machine (SVM) is presented. This method outperforms all methods in the literature up until its release (2011) in deployment conditions, even though the computation time for enrollment remains higher. . In order to compare the performance of the proposed method with that of others, a database was created with more than 100 users with at least 5 sessions for the acquisition phase. One of the conclusions this team reached is that using individual thresholds could improve the performance of the system. Good robustness was shown for the tried algorithms for different keyboards. The benefit of supervised template update mechanisms of the biometric reference was also demonstrated. However, they noted that several factors have to be tested in the keystroke dynamics domain. This often implies creating a new database especially designed for the corresponding tests. [1].

Related to the previous paper, we find another that also uses SVM for the application of keystroke dynamics to collaborative systems (collaborative systems need to authenticate their users). This study states that, even with quite simple methods from the state of the art, the obtained results are almost correct (less than 5% of error) but need to be improved. One-class Support Vector Machines with only five vectors per users for the training seems to give better results. In the tested systems, only well typed passwords are taken into account. This is a problem with static password based authentication methods, because the genuine user can correct his own typing errors and being correctly authenticated. On the contrary, with the keystroke dynamic implementation described on this work, the system will force the user to type again the password in order to have a correct sized vector. One flaw of this system is that it has to be modified to allow the use of backspace key to correct the password. [7].

However, SVM is not the only model used for keystroke dynamics. There is one that makes an analysis is made on the performance of different methods for keystroke analysis. At the release of this paper (2009) there is a lack of available databases to test different issues and improve systems' performances. The performance of methods in the state of the art varies a lot in function of the size of the database in term of number of users (and impostors' data) and number of captures per user (and deviations

in the templates). The performance of the SVM-based method presented is lower than in the previous published studies, as phrases are used instead of free passwords and only 5 captures for the enrollment step. For this context, the method presented on this work outperforms all other tested methods from the state of the art. One clear conclusion is that individual thresholds improve the performance of systems. However, it is noted that the performance of the method is highly dependent on our database on the distribution of the individuals, and the password itself, which shows one use case for this method [8].

A different approach, using not single strokes but groups of them, is presented on in other study. This work collects data in the form of digraphs from a series of users entering a specific login ID. The goal was to determine if there were any particular patterns in the typing styles that would indicate whether a login attempt was legitimate or not using rough sets. The analysis produced a sensitivity of 96%, specificity of 93% and an overall accuracy of 95%. The results of this study indicate that typing speed and the first few and the last few characters of the login ID were the most important indicators of whether the login attempt was legitimate or not. The most interesting result from this study indicates that the digraph times were most critical for determining whether a user was legitimate. The decision class of the legitimated owner took the least amount of time in entering the characters of their login ID compared with that of an illegitimate owner. In addition, for the legitimate owner of the login ID, the first few and last digraphs were sufficient to make a correct classification[4].

Uses of other technologies such as neural networks for analysis of keystroke dynamics are also presented in the state of art. In order to analyse this process, the authors focused on the commonly used MLP/BP model as well as the multi technique employing Neural Network. In this study more than 100 students and staffs were involved. They were required to type the same word “Thurs1day” a number of times they wish. Ten users were selected to be authentic users and attempts collected after profile creation amounted to 5440. Some of these were genuine attempts while others impostor attempts. Some problems raised when using a Neural Network in this scenario: given the inherent characteristic of Neural Network it will produce an output using the current inputs and the models learn, therefore it will always match an impostor to one of the authentic users; also, a careful selection of system parameters is very important, even more so for Neural Networks than for other models. The results show that an intruder

detection unit placed before the Neutral Neural network is primordial to enhance its usability and acceptability. By removing the intruder and presenting only authentic users to the neutral network an ideal system can be achieved even with learning sample consisting of fewer attempts. However such an implementation may also aim at modifying the neural network to achieve less computation [5].

Another way to analyse data obtained from keystrokes is the fusion of characteristics. A keystroke dynamic recognition system is presented by using a fusion method. Firstly, the dwell time and the flight time are recorded as the feature data. Their mean and standard deviation values are also calculated and stored. The test feature data will be transformed into the scores via Gaussian probability density function. On the other hand, a new technique known as Direction Similarity Measure (DSM) is proposed to measure the differential of sign among each coupled characters in a phrase. Lastly, a weighted sum rule is applied by fusing the Gaussian scores and the DSM to enhance the final result. The best result of equal error rate 6.36% is obtained by using a home-made dataset. Results show that a combination of dwell time and flight time yields better results than using them and other measures individually. Also, combining them with the DSM method, the result is improved[6].

The authors of these paper, instead of fusing characteristics, tried to give more value to some of them, deeming them more important for keystroke dynamics. They tried to answer two questions. First, what influence do each of six factors—algorithm, training amount, feature set, updating, impostor practice, and typist-to-typist variation—have on keystroke dynamics error rates? Second, what methodology should be used to establish the effects of these various factors? In answer to the first question, the detection algorithm, training amount, and updating were found to strongly influence the error rates. No difference was found among the three feature sets, and impostor practice had only a minor effect. Typist-to-typist differences were found to introduce substantial variation; some subjects were much easier to distinguish than others. In answer to the second question, a methodology was proposed with roots in the scientific method: experimentation, statistical analysis, and validation. This methodology produced a useful, predictive, explanatory model of anomaly-detector error rates. Consequently, the authors reached to the conclusion that the proposed methodology would add valuable predictive and explanatory power to future anomaly-detection studies [11].

Lastly, the study focuses on the integration of keystroke dynamics along with other additional instruments. Based on a data set of 1254 participants who typed the same user ID and password, 20 times each, the authors developed a test statistic and obtain the power of this test. The results show that keystroke dynamics can be a reliable security instrument for authentication, if used together with other instruments. It seems more suitable for authentication (verification) than for identification. Dwell times (how long a key is held pressed) are more discriminatory and therefore more powerful than flight times (time between consecutive press times) [12].

2.2.2 Mobile Authentication Studies

Applying keystroke dynamics on mobile keyboards requires of a different set of measures than on physical keyboards, and can also use more innovative measures such as pressure depending on the touchscreen used to collect data. Some of these innovative measures are explored on paper [13], which couples keystroke dynamics with a knowledge factor to add as a two-factor authentication without carrying more hardware, such as a smartcard or a one-time password generator. Keystroke adds to the smartphone and tablet authentication process an additional factor. This makes the authentication process more secure than a one-factor authentication with a password. Today, smartphones and tablets have additional sensors which can be used to extract features. These features give us the possibility to increase the usability without lowering the security level. The user does not need to write a whole sentence to authenticate himself. His "normal" password can be used which is basically only a one-factor authentication. During typing this password biometric features are extracted and compared to the database. So it does not need more time to verify each user. All together the features which could be extracted with the capacitive display are leading to more analyzable data which will lower the FAR and FRR. The authors suggest to develop a mixture of a keystroke-based and a handwriting-based authentication method using capacitive displays. The authors also discuss limitations of existing approaches and why it is believed that keystroke and handwriting authentication is a possible way for improving the security on mobile devices. First experiments demonstrate the effectivity of this new approach with error rates under 2%. Some limitations of this approach could be the learning process of the user. In addition, the behavior varies daily and also over years and because of injuries. In the future, touch pads maybe could provide more

information like the fingerprint or the temperature of the finger. This data could be added as well to the authentication process and would have the advantages of a fusion with existing methods. Other methods like authentication via Swype input could be analyzed.

A more statistical approach to keystroke dynamics is described on this paper. Here, a statistical median-based classifier model is presented to serve as an anomaly detector in keystroke dynamics authentication on mobile devices. The proposed classifier uses the timing features of keystroke dynamics as well as touch features of mobile devices. Formulation and evaluation of proposed classifier have been influenced by an empirical analysis of a public dataset of keystroke dynamics on Android platform, The classifier considers the distance from the median of a feature element as an indicator of whether it relates to a genuine user or an impostor, based on previous training data of genuine users. The formula for measuring the acceptable distance to the median is derived from the distance between the minimum value of a feature element and its median. The model was evaluated through comparison with previous detectors, using the same dataset, and the results have shown a significant reduction in the equal-error-rate. The inclusion of touch features of mobile devices in user authentication has resulted in lower Equal-Error-Rate (EER), and confirmed in the present work using the Med-Min-Diff classifier. This suggests that other mobile devices' related features can further reduce the error rate, leading to more accurate authentication. The differences in the reported coefficient of variation among features can be a useful guide in determining which features to be given more weight in the authentication process. Also, the EER analysis results of the proposed classier have shown a significantly lower EER value compared with the three verification models, which should lead to further investigation, to enhance the median-based model as a classifier in keystroke dynamics authentication[9].

More extensive work in classifiers and distance metrics is exposed on this study. It is demonstrated experimentally that touchscreen based features improve keystroke dynamics based identification and verification. A dataset was collected using Android devices with touchscreens. Both time and touchscreen based features were studied. Identification measurements were performed using several machine learning algorithms, of which the best performers were Random forests, Bayesian nets and SVM, in this order. Not only identification, but also verification measurements were performed on the same datasets. In this case EER were computed using three different distance

metrics: Euclidean, Manhattan and Mahalanobis. Manhattan was the best performing distance function. In case of identification mesasurements, the addition of touchscreen based features to the default feature set induced an increas of over 10% in accuracy for each classifier. This improvement is harder to notice in the case of verification measurements where the equal error rate was reduced by 2.4% (Manhattan metric). In the data preprocessing stage, it was observed that several typing patterns contained deletions and these were eliminated from the dataset. However, these errors can be considered a separate feature of the user and can be studied in the future [10].

Finally, with a more conservative approach to keystroke dynamics but on Android, this paper proposes a biometric authentication system which use password based and behavioural traits (typing behaviours) authentication technology to establish user's identity on a mobile phone. The presented system builds a multi-level mobile authentication model, and also combines with the keystroke analysis technique which can effectively prevent the potential attacks from criminals. The typing behaviour recognition enhances user ID and password based authentication with keystroke analysis that periodically asks the user to re-verify their identity. Experimental results indicate that the developed authentication system is highly reliable and very secure with an equal error rate is below 7.5% [3].

2.2.3 Others keystroke dynamics studies

Although keystroke dynamics was firstly developed to replace computer password, it is being used also for other fields of study. Some focus in medicine and psychology fields.

For instance, there is an on work investigation called "Monitor Parkinson's symptoms" in which the researchers set out to investigate whether keystroke dynamics, could be used to monitor the motor effects of Parkinson's disease in the home. In previous work the researchers had demonstrated that the technique can be used to spot signs of sleep inertia, or the decline in motor dexterity caused by grogginess on being suddenly woken. Another work that focus on a different field is "Identifying emotion by keystroke dynamics and text pattern analysis" developed by MIT in 2014. They suggested that if computer systems were capable of recognizing users‘ emotions they would be able to make more intelligent decisions, so they provided a methodology that can be used for

creating emotional state models based on their keystroke timings that had a classification rate of 79.5%.
The paper titled "A Case-Based Approach Using Behavioral Biometrics to Determine a Users Stress Level" try to identify individuals' medical health status- By applying biometrics in combination with case based reasoning and personification, the potential of more effective diagnosis, prevention and treatment is emerging. Their approach was to detect muscle tension in user's keyboard usage. Their results were much higher working on non Stress detection, depending on the user, the accuracy of the results varied from 65% to 100%.

Chapter 3

Bank Study

The data collection process is a critical point because the reliability of results achieved depends on the quality of raw data. In order to know what data must be collected and how to handle it, a study was made to check the different identification methods that banks use.

3.1 Bank Login Methods

In order to identify the most used method for login process, we studied a total of 58 banks from Europe. Table 3.1 shown the information we gathered. The following characteristics were studied:

- For user identifier:
  - **Personal Identification Number**. National Identity Number (e.q. DNI), passport.
  - **User Name**. This refers to an unique word that the bank gives us.
  - **Account Number** The account number that a user has.
  - **Credit Card Number** The credit card number a user has.
  - **Birthdate**. The user's birthdate.

  - **Postal Code**. The user's postal code.
  - **Paste enabled**. If pasting the field content is allowed.

- We got these kind of passwords:
  - **Password**. Hidden characters to enter.
  - **Virtual keyboard**. The password has to be written using a virtual keyboard.
  - **SMS**. You have to enter a code sent to your mobile phone.
  - **App code**. You have to have installed the bank application to be able to make login.
  - **Paste enabled**. If pasting the field content is allowed.

Table 3.1 Heading

- The columns names of Table 3.1 correspond with: CC is Bank's Country Code,Bank name. For User field, we have the following columns: ID can be DNI or Passport number, ACC# refers to Account Number, CCN to Credit Card Number, Birth is birthdate, PC: Postal Code, C&P: Tells you if paste is allowed; For password field we have these columns: Pwd that indicates if a password is required and them Minimum length, Max length, type, VirKey refers to virtual keyboard.
- For row we have four values: M: Mandatory. It means that you always have to enter it. O: Optional. It means that you can choose the field to fill because there are more than one option but at least one must be filled. T: True. Pasting is allowed and F: False. Pasting is not allowed.

After the study, we concluded that:

- Almost 100% of the banks use some kind of login process that includes these two kind of inputs; one for username and another one for password.
- Only 12% of the banks studied, add the extra security characteristic of using virtual keyboard instead of physical keyboard.

- Almost all banks have the security restriction of not allowing copy&paste in password inputs. This issue will allows us to ensure that the user will be at least forced to type the password.

		User Fields							Password Fields				
CC	**Bank Name**	**ID**	**User**	**ACC#**	**CCN**	**Birth**	**PC**	**C&P**	**Pwd**	**VirtKey**	**SMS**	**APP**	**C&P**
ES	Santander	O	O					T	M...	O			F
ES	Sabadell	O			O			T	M,,8,				F
ES	BBVA	O	O					T	M...				F
ES	ING Direct	M				M		T	M...	x			F
ES	La Caixa	M						T	M...	x			F
ES	Bankia	M						T	M,,8,				F
ES	Banco Popular	O	O		O			T	M...	x	x		F
DE	Deutsche Bank			O	O			T	M,5,5,				F
DE	Commerzbank	O			O			T	M,,8,				F
DE	Kfw Bankengruppe	M						T	M,,,				F
DE	HypoVereisbank			O	O			T	M,,10,				F
FR	Crédit Agricole	M					M	T	M,,,	M			F
FR	BNP Paribas	M						T	M,,,	M			F
FR	Société Générale	M						T	M,,,				F
FR	Caisse D'Epargne	M						T	M,,,				F
IT	UniCredit	M						T	M,,,				F
IT	Intesa Sanpaolo	M						T	M,,,				F
IT	Cassa Depositi e Prestiti	M						T	M,,,				F
IT	Banca Monte dei Paschi di Siena	M						T	M,,,				F
IT	Banco Popolare Società Cooperativa	M						T	M,,,			M	F
IT	UBI Banca	M						T	M,,,				F
IT	Banca Nazionale del Lavoro	M						T	M,,,				F
IT	BPER Banca	M						T	M,,,				T
IT	CheBanca!	M				M		T	M,,5,	M			T
IT	Crédit Agricole Cariparma	M				M		T	M,,6,	M			F
GB	HSBC	M						T	M,,,				T
GB	Royal Bank of Scotland	M						T	M,,,				F
GB	Lloyds TSB	M						T	M,,,				T
GB	Barclays	O			O			T	M,,,				T
CH	UBS Group	M						T	M,,,				F
CH	Credit Suisse	M						T	M,,,				F
CH	Raiffeisen Switzerland	M						T	M,,,				T
CH	Zurich Cantonal Bank	M						T	M,,,				T
BE	BNP Paribas Fortis	O			O			T	M,,,				F
BE	ING	O			O			T	M,,,				F
BE	Hello Bank!	O			O			T	M,,,				F
AT	UniCredit Bank Austria	M						T	M,,,				T
AT	Volksbank Verbund	M						T	M,,,				F
AT	Raiffeisenlandesbank Oberösterreich	O			O			T	M,,,				T
AT	BAWAG P.S.K.	M						T	M,,8,				F
FI	Nordea Bank Finland PLC	O	O		O			T	M,,4,			M	T
FI	OP-Pohjola Group	M						T	M,,4,				F
FI	Dankse Bank PLC	M						T	M,,,				F
FI	Aktia Savings Bank PLC	M						T	M,,,				T
PT	Banco BiG	O	O					T	M,,,				T
PT	Banco de Portugal	O	O					T	M,,,				F
IE	KBC Bank Ireland	O	O					T	M,,,			M	F
IE	Ulster Bank	O	O					T	M,,,				F
IE	Permanent TSB	O			O			T	M,,,				F
IE	Wells Fargo Bank	M						T	M,,14,				F
UA	VS Bank	M						T	M,,,				F
UA	Pravex Bank	M						T	M,,,				F
UA	Platinum Bank		O		O			T	M,,,		M		T
UA	Piraeus Bank	M						T	M,,,				F
MC	Banque Havilland		M					T	M,,,	M			T
MC	Monte paschi Monaco		M					T	M,,,				T
MC	Barclays	O			O			T	M,,,				T
MC	UBS Group	M						T	M,,,				F

Table 3.1: Bank user and password fields results

Chapter 4

Data Collection

This process is critical for the final results of our study because collecting high quality data guarantee the reliability of the results. During this process we had to find ways to extract as much information as we can from key down and key up events. A visual representation about what the key down and key up timestamps are can be observed at Figure 4.1.

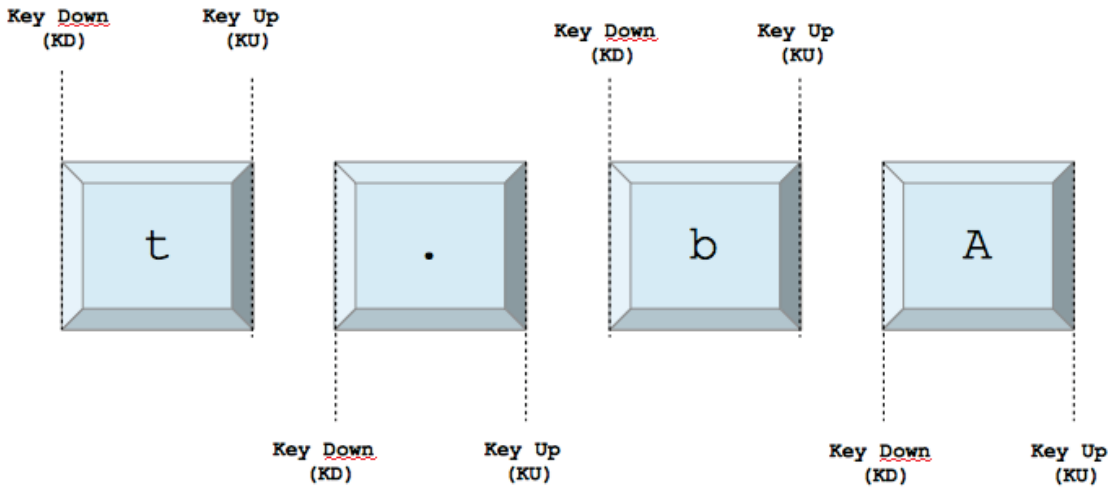


Figure 4.1: Sequence of key down and key up events

4.1 Defining Data Collection Output

The first step we had to manage is defining the structure of the output and how many fields we were going to record.

In relation to keystroke dynamics, we agreed that the user id and password keystrokes dynamics must be separated. For each key press event we must record:

- **Key code**. The key code of the key pressed. This information must be removed in production phase.
- **Key down timestamp**. Time when the key was pressed.
- **Key up timestamp**. Time when the key was released.
- **Backspace**. True if the press key is a backspace key.
- **Delete**. True if the press key corresponds with delete key.
- **Special**. True if the press key is not alphanumeric.
- **Meta**. True if mac OS cmd key has been pressed.
- **Ctrl**. True if ctrl key has been pressed.
- **Right Shift**. True if right shift key has been pressed. (Only for web features)
- **Left Shift**. True if left shift key has been pressed. (Only for web features)
- **Shift**. True if shift key has been pressed. This feature is used in replacement of the two previous fields because mobile platforms do not have right and left shift. (Only for mobile features)

The same process of data collection is also made for contextual features. Some of the features are:

- **Operative System**.
- **Platform**.

- **Browser language**

The whole list of all the contextual and timing features can be found json file whose schemes are in the appendixes .1 and .2.

4.1.1 Defining Delete Methods

There are 5 main methods that the user can use for deleting a character:

1. **Backspace key**. Deletes the previous character o a set of them if the hold time key is bigger.
2. **Ctrl-a command + any other character**. Selects the code and removes all selecting code.
3. **Delete key**. Delete the events or events from cursor position until the end.
4. **Double click + any other character**. Selects the code and removes all selecting code.
5. **Mouse selection**. Selects the code and removes all selecting code.

We currently support the first 2 approaches in this first iteration of the project. We decided not to support the other three because of the challenge of knowing where the cursor is.

4.2 Development Platform

To collect keystroke dynamics data, we needed to develop the toolkit. For this issue, we had to choose which development platforms we were going to use. Despite the project covering both, mobile and web platforms, we are going to focus on the web development. Another partner's project is incharged of mobile development.
The language chosen for developing the data acquisition was Javascript. We chose it, because is one of the most common used languages in web and mobile and their cost and performance are not high. For the backend part, we chose Python as language and Flask as framework[25]. We explain more about the backend in Section 4.2.2

4.2.1 Web Platforms

One of the most important web characteristic is the fact that it must work on any browser, independently of the operative system. Bearing in mind this, we built two tools related to web, a web page and a Chrome extension.

Website

The development of the web page allows us to collect data in a controlled environment. This control give us the opportunity to set and record as much information as we want and being prepared to web changes. The web was developed using AngularJS 4. AngularJS is a TypeScript-based open-source front-end web application framework mainly maintained by Google.
As we explain in section 3.1, the inputs that users will have to type consist of the typical knowledge based authentication method, the pair of user ID/password which are the same for all users.

How to use. Each user that wants to participate in our study has to register. It is mandatory to enter his/her name, surname, email address and username. Once the server gets the register submission, an email indicating the access password will be sent. The first image this new user will be able to see are two web inputs in which he/she will have to enter the user ID and password. The process is not completed until it is repeated 10 times per session.

Chrome Extension

Chrome is currently the most popular browser[26]. This is the reason why we chose Chrome as our first option. One of our proposals is not to disturb the user too much and make the process of data collection as transparent as possible.
Although no experiments have been done with the extension yet, it was developed to get ready if some banks denies to give us real data and being able to get data from real scenarios. The Chrome extension injects Javascript code in input boxes that allows to get information about keystroke dynamics.

How to use. The only process a user has to do is to install the Chrome extension on the Google Chrome browser. The process of data collection is transparent for him. He just have to log in the banks' websites as he usually does. The data collected will be sent to our server in the same way that the webpage does. The user ID will be hashed for avoiding privacy issues.

4.2.2 Backend Platforms

In order to be able to manage and store all the raw data collected, a backend was built. We opted for building an REST API in Python. The REST API was made with the use of Flask. Flask is a micro web framework written in Python and is considered more Pythonic than Django because Flask web application code is in most cases more explicit.
The functionalities that this REST API can manage are:

- **Register process**. Once the registration petition is completed, this REST API is the one that have to manage if the username and email are unique. If the information is correct, an email with a randomized password will be sent. Otherwise if any problem arises, it will be shown on the webpage to the user. This process is used by the web and the mobile app.
- **Login process**. The REST API is the one that control the session ID and the user ID allowing the concurrency of users. This process is used by the web and the mobile app.
- **Data collection process**. Once a person enrolled types the user ID and password and submits, a json is built and sent to this server. You can choose if saving the json in a file or in Mongo DB.This process is used for the two tools developed.

The server is running under an Ubuntu OS server.

Chapter 5

Features Extraction

Once we had our raw data, we could proceed to define and extract he features that will be used in machine learning algorithms.

5.1 Features

For these first approach, we focused on timing features. A brief summary of the total features used is shown at Table 5.1. The features selected are:

- **Dwell time**. It is also known as hold or press time. It refers to the time duration of a touch event with the same key. The interval between key press and key release.
- **Flight time**(Latency). The time interval between releasing one key an pressing the next one. This timing could be negative. It may happen due to the difference in physical and geometrical size of virtual keys against physical keys.[14]
- **Down down time**. It refers to the time interval between two key down events.
- **Up up time**. It refers to the time interval between two key up events.
- **Total time length**. This is the total time that a user uses to write both username and password before submitting.

- **Right shift**. Number of right shift keys pressed.
- **Left shift**. Number of left shift keys pressed.
- **CapsLock**. Boolean. Indicate if capslock key have been pressed.
- **Shift**. Number of shift keys pressed in mobile platforms, where there is no right/left shifts.

Username				Password							
Dwell	Flight	Down Down	Up Up	Dwell	Flight	Down Down	Up Up	Total (user + pwd)	Right Shift	Left Shift	CapsLock
Float	Float	Float	Float	Float	Float	Float	Float	Float	Integer	Integer	String

Table 5.1: Features

5.2 Feature Extraction

To be able to calculate the features, we had to clean the datasets. As we explained in Chapter 4, the keystroke raw data is just a sequence of events. Events that can be supported for our software or not.

5.2.1 Timing Features

Timing features are calculated separately for both fields, user ID and password. In Figure 5.1 we can observed how we calculate them.

5.2.2 Copy and Paste

The functions copy and paste are just a sequence of key events, Ctrl+c or Ctrl+v respectively. Although in our web development, these functionalities are disabled, the events are recorded in any case. As these sequences of events are not desirable and do not give valuable information for keystroke dynamics, they are removed.

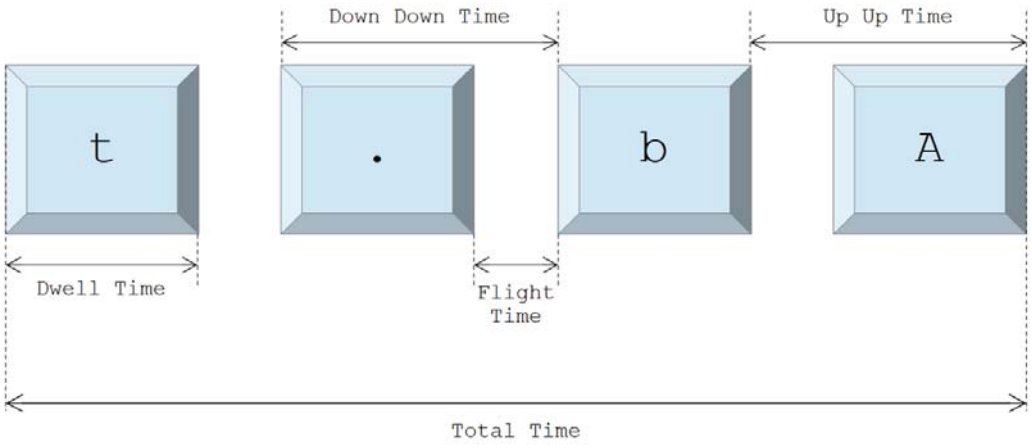


Figure 5.1: Timing features

5.2.3 Correct and Delete Sequences

As we have mentioned before, any key event that happened during a session is recorded and thats also includes mistakes. If the user is not aware of them and presses submit, the session result will be failure, but if he is aware of and wants to delete their incorrect characters, we need to manage how to differentiate final character versus deleted ones. In Section 4.1.1, we explained that there are five ways of deleting (backspace, ctrl-a, delete, double click, mouse selection). We also explained that we only support backspace and ctrl-a because the rest require to identify the cursor position. Both correct and incorrect events, gives valuable information, so we had to divide which characters have been removed. In order to solve this we had to face with some challenges:

- When the backspace key comes after the sequence of events represented in Figure 5.3. Although you can treat them as unique event in terms of calculating the features or not, you must move two or three events to the deletion field.
- You have to check when backspace is pressed if the previous event is a valid event or it is just an event without meaning like an alone shift key. In that case, the event that has to be removed is the one before the shift.
- We had to define the period of time to know whether the backspace must delete one or all characters.

There are other events that are not taken into account for introducing them into correct or delete sequences. Usually we just ignore the delete key but we also ignore sequences that have no sense such us the one shown in Figure 5.2. In this Figure we just see that the user pressed three times shift before pressing the A character. We just record the last two.

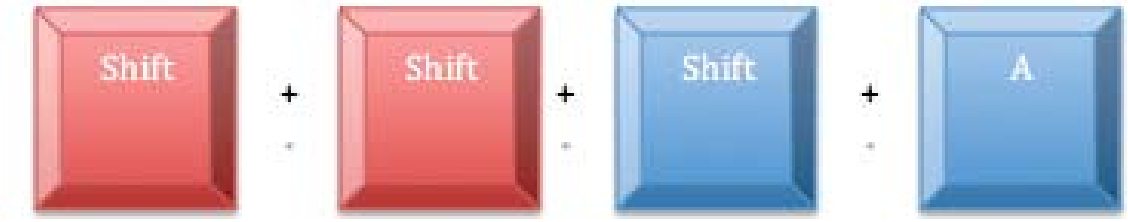
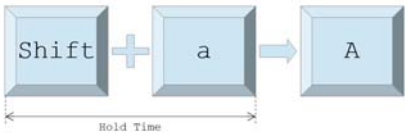


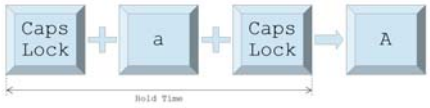
Figure 5.2: Unusual sequence of events

5.2.4 Shift and CapsLocks events

In order to calculate the timing of certain sequences of events, an agreement was needed. These sequence events were those used to write capital letters; shift (5.3a) or capslock key (5.3b). It can be treated as both unique events as it is presented in Figure 5.3 or individual events. Our script is able to calculate in both ways. In addition, if the



(a) Shift event



(b) Capslock event

Figure 5.3: Shift and capslock events

capital letter is treat as two individual events, some features like flight time will be always negative. If we look at Figure 5.4, flight time between shift and a is equal to KD3-KU2 being KD3 minus than KU2.

As additional information, we realized that when a person writes, he/she uses in the same shift/capslock key in 99% . For instance, if a person is used to write capital

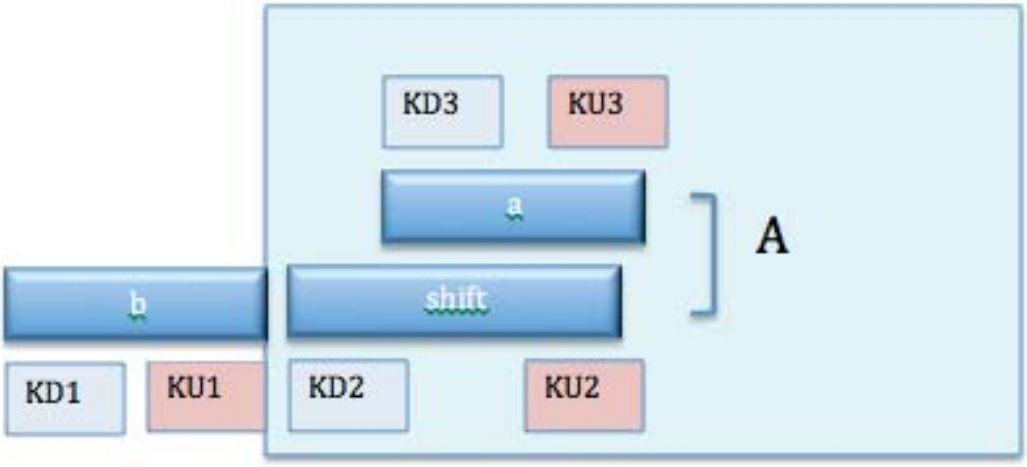


Figure 5.4: Capital letter as two individuals events

letters using right shift, it is most likely this user is always going to write capital letters in the same way. Thats is teh reason why we also included the features right Shift, left Shift and CapsLock to the feature list.

The feature extraction process is done by executing a Python script that turn the raw data json files into feature json files.The json format is defined by their schema that can be found in Appendix .3.

Chapter 6

Evaluation

6.1 Dataset

In order to get the validation results, we used two datasets, one with web characteristics and the other one with app characteristics. Both datasets were collected by one of the companies in the project. The characteristics of these data collection processes are:

1. **Number of users**. We will have a total of 50 users, all acting as authentic users.
2. **Session length**. The number of repetitions that a user has to do before finishing the session is 10.
3. **Number of sessions**. The number of sessions per user is 45.
4. **Inputs**. The inputs will consist of user ID and password. Both will be the same for all users.The user ID to enter is "16295538" (8 characters). The password to enter is "t.biAs198" (9 characters).
5. **Correct and Incorrect session**. The session in which the user ID or password did not match with the established ones, will be also saved.

In general terms, every user had to complete 45 sessions of 10 tries each one. This inputs could not be all done at the same day and had two weeks for the collection

data. For the experiments we are going to carry out we only focused on sessions that finished in correct way, and we only took into account the correct characters, removing the deleted ones.

6.2 Studying the Raw Data

Once we got our raw data, we processed it in order to get a general idea of the diverse kinds of cases that we have and check if we were able to .

6.2.1 Login Attempts

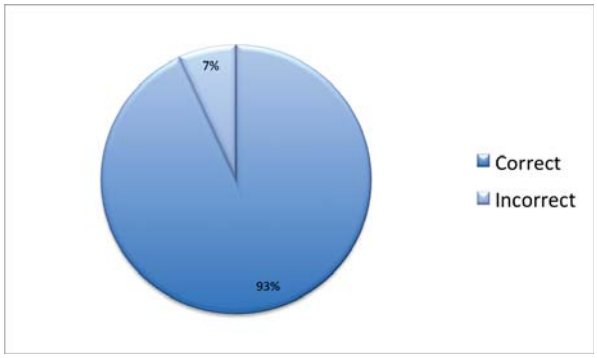


Figure 6.1: Percentage of correct versus incorrect attemps

In Figure 6.1, we observed that more than 93% of the cases registered finished in a correct login. This is when user ID and password are correct when submitting.

6.2.2 Usage of Delete Keys

In our feature selection, we are ignoring the delete key because it carries out that we need to know th cursor position. In figure 6.2, we can see that around 32% of the users use delete key in our dataset.

Also we have realized that the 86% of the mistakes done, are in password field. That probably happens because the inputs in this field are hidden so you can not see what

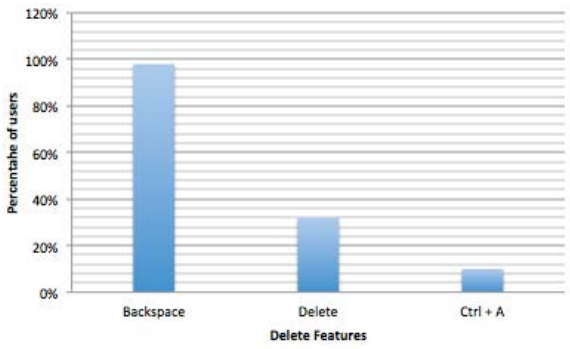


Figure 6.2: Percentage of users that uses the delete keys

you have written. Another curious fact is that when users make mistakes in password, 73,5% of them removed all written characters and started again from the beginning. But, if we compare user ID field, only 20% remove everything.

6.2.3 Capital Letters

In the web dataset, we could study the way that our users wrote capital letters. In Figure 6.3, we observe that the most commonly used is Left Shift (56%) followed by Right Shift (30%) and Capslock(14%).

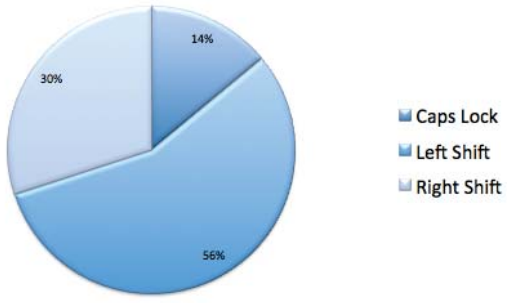


Figure 6.3: Percentage of left shift, right shift and Capslock

6.3 Feature Selection

During the several discussions about what fields to include in a data collection, we thought that the characteristic of shift usage and Capslock could give us valuable information. To check whether our intuition was correct or not, we processed our data with an evaluator of gain ratio attribute.
In Table 6.2, we can observe that these three features are the ones that give us more information. In fact, the difference between these three features and the next one is notable.

6.4 Measurements

All the measurements were performed using WEKA. Different kinds of classifiers had been used previously in keystroke dynamics. For this thesis we are going to cover:

- Decision trees. Popular methods based on tree like graphs. We used Weka's JK48 method which is an implementation of C4.5 algorithm (confidence factor 0.2, minimum 2 instances per leaf).
- Naive Bayes. A probabilistic classifier based on Bayes theorem.
- Bayesian Networks. A probabilistic model that represents a set of random variables and their conditional dependencies using a directed acyclic graph.
- K nearest neighbors. Known as IBk in Weka. It is an instance based classification algorithm where new instance label is chosen by the K closest neighbors.
- Support Vector Machines. They build a linear discriminant function that separates the instances of cases.

6.5 Classification Results

We got the datasets processed by Weka with a cross validation fold of 10. The results are shown in the following table:

	Web		**Mobile**	
	FRR	FAR	FRR	FAR
C4.5(JK48)	13,90%	0,30%	13,10%	0,30%
Naive Bayes	38,10%	0,90%	54,80%	1,20%
Network Bayes	6,30%	0,10%	9,00%	0,20%
k-NN	13,70%	0,30%	21,20%	0,50%
SVM	27,30%	0,70%	27,70%	0,70%
Random Forest	3,03%	0,10%	4,40%	0,10%

Table 6.1: Validation results in terms of FRR and FAR

The table presents the results in terms of False Rejection Rate (FRR) and False Acceptance Rate (FAR). We can observe that in both cases web and mobile platforms, the FAR values are quite low. This will mean if we talk about bank login that less that 1% of the users that supplant other person will be able to enter. There are other cases where the FRR is higher, which means that a great percentage of the users that are going to enter their own credentials, will be rejected. It happens when we use some classification algorithms such as Naive Bayes or K-Nearest Neighbors methods.
In general terms, the results are better in web rather than mobile but in a small percentage. The reason why is always higher could be because in webs you have more information about right shift, left shift and CapsLocks.

Feature Name	Information Gain	Feature Name	Information Gain
left_shift	**0.898**	pwd_flight_5	0.271
right_shift	**0.875**	pwd_flight_3	0.27
capsLock	**0.672**	pwd_upup_3	0.269
user_flight_0	0.361	user_downdown_5	0.268
pwd_flight_7	0.349	pwd_upup_0	0.266
pwd_upup_7	0.34	user_flight_5	0.265
user_flight_2	0.334	user_upup_1	0.262
user_upup_6	0.332	pwd_downdown_5	0.261
user_downdown_0	0.33	pwd_downdown_3	0.259
user_flight_6	0.328	pwd_upup_5	0.259
pwd_flight_2	0.327	pwd_upup_1	0.257
user_downdown_6	0.325	user_dwell_0	0.257
user_upup_0	0.324	pwd_dwell_2	0.256
pwd_dwell_4	0.323	pwd_dwell_3	0.254
user_downdown_2	0.32	pwd_dwell_8	0.252
pwd_downdown_2	0.316	user_dwell_1	0.251
pwd_upup_2	0.312	pwd_flight_1	0.249
pwd_downdown_7	0.309	user_upup_5	0.249
user_upup_2	0.303	pwd_dwell_6	0.248
pwd_downdown_4	0.303	user_flight_4	0.246
pwd_flight_6	0.3	user_dwell_3	0.245
user_flight_3	0.298	pwd_dwell_5	0.24
pwd_downdown_6	0.298	user_dwell_6	0.235
user_flight_1	0.291	user_dwell_7	0.235
total_time	0.291	pwd_downdown_1	0.234
pwd_flight_4	0.287	user_upup_4	0.234
pwd_dwell_7	0.283	user_downdown_1	0.234
pwd_downdown_0	0.281	pwd_dwell_1	0.234
pwd_upup_4	0.275	user_dwell_2	0.229
user_downdown_3	0.275	user_dwell_5	0.219
user_upup_3	0.273	pwd_dwell_0	0.217
pwd_flight_0	0.273	user_dwell_4	0.212
pwd_upup_6	0.272	user_downdown_4	0.192

Table 6.2: Features gain ratio

Chapter 7

Conclusion and Future Work

7.1 Conclusion

Keystroke dynamics has a lot of advantages. There is no need to add extra hardware to deploy it or have minimum cost.
Looking at the raw data, we have just realized that there are some events that are been ignored that must be taken into account such as delete key that is used by more than 3O% of the user. There are also, other cases in which the mouse selection is been done. We have also realizes that a 10% of the users have used some kind of mouse selection deletion.
Using Keystroke dynamics to the authentication process add an additional factor that makes them more secure.In our approach, we study timing features of both inputs username and password which allows us to get more information about the user. No many papers read take information about the username too. In addition, no study mix Shift/CapsLock information with the username information. This may be the reason why we obtained better results .
The results we got from Weka in terms of FAR and FRR are quite low comparing with other papers results though having an FRR value of 3% is high if we are talking about a million of users. This is the case of using Random Forest. On the other hand we have the naive bayes method whose FRR result is quite high. It varies from 38% too 55% in mobile platforms

Although to make sure that it always happen, we must prove with bigger datasets to check if our model validates all users in so perfect way.

7.2 Future Work

In this master thesis, we have not evaluated neural networks. We leave their evaluation as future work. Bearing in mind that all the results presented are made from the first raw data collected, there are some scenarios that we did not take into account right now but we suspected they could happen as we mentioned in Section 7.1. So we must remake the data collection process for enabling the recording of mouse interaction and deleting key.

One of these scenarios that we are talking about, was when a user selects the text for deleting but with mouse instead of using backspace key or ctrl-a command. As we are not controlling clicks on these inputs, it is not possible for us to filter between the corrects and deleting character. As a very first step, we must remake the data collection code for enabling the recording of mouse interaction.

Appendices

.1 Web Data Collection Schema


```
{
"$schema": "http://json-schema.org/draft-04/schema#",
"definitions": {},
"id": "http://example.com/example.json",
"properties": {
"data": {
"id": "/properties/data",
"properties": {
"_id": {
"id": "/properties/data/properties/_id",
"type": "string"
},
"correct": {
"id": "/properties/data/properties/correct",
"type": "boolean"
},
"datetime": {
"id": "/properties/data/properties/datetime",
"type": "string"
},
"fingerprint": {
"id": "/properties/data/properties/fingerprint",
"properties": {
"adblock": {
"id": "/properties/data/properties/fingerprint/properties/adblock",
"type": "boolean"
},
"available_resolution": {
"id": "/properties/data/properties/fingerprint/properties/available_resolution",
"items": {
"id": "/properties/data/properties/fingerprint/properties/available_resolution/items",
"type": "integer"
},
"type": "array"
},
"color_depth": {
"id": "/properties/data/properties/fingerprint/properties/color_depth",
"type": "integer"
},
"cpu_class": {
"id": "/properties/data/properties/fingerprint/properties/cpu_class",
"type": "string"
},
"do_not_track": {
"id": "/properties/data/properties/fingerprint/properties/do_not_track",
"type": "string"
},
"hardware_concurrency": {
"id": "/properties/data/properties/fingerprint/properties/hardware_concurrency",
```

```
"type": "string"
},
"has_lied_browser": {
"id": "/properties/data/properties/fingerprint/properties/has_lied_browser",
"type": "boolean"
},
"has_lied_languages": {
"id": "/properties/data/properties/fingerprint/properties/has_lied_languages",
"type": "boolean"
},
"has_lied_os": {
"id": "/properties/data/properties/fingerprint/properties/has_lied_os",
"type": "boolean"
},
"has_lied_resolution": {
"id": "/properties/data/properties/fingerprint/properties/has_lied_resolution",
"type": "boolean"
},
"hash": {
"id": "/properties/data/properties/fingerprint/properties/hash",
"type": "string"
},
"indexed_db": {
"id": "/properties/data/properties/fingerprint/properties/indexed_db",
"type": "integer"
},
"ip": {
"id": "/properties/data/properties/fingerprint/properties/ip",
"type": "string"
},
"language": {
"id": "/properties/data/properties/fingerprint/properties/language",
"type": "string"
},
"local_storage": {
"id": "/properties/data/properties/fingerprint/properties/local_storage",
"type": "integer"
},
"navigator_platform": {
"id": "/properties/data/properties/fingerprint/properties/navigator_platform",
"type": "string"
},
"pixel_ratio": {
"id": "/properties/data/properties/fingerprint/properties/pixel_ratio",
"type": "integer"
},
"regular_plugins": {
"id": "/properties/data/properties/fingerprint/properties/regular_plugins",
"items": {
"id": "/properties/data/properties/fingerprint/properties/regular_plugins/items",
"type": "string"
},
```

```
"type": "array"
},
"resolution": {
"id": "/properties/data/properties/fingerprint/properties/resolution",
"items": {
"id": "/properties/data/properties/fingerprint/properties/resolution/items",
"type": "integer"
},
"type": "array"
},
"session_storage": {
"id": "/properties/data/properties/fingerprint/properties/session_storage",
"type": "integer"
},
"timezone_offset": {
"id": "/properties/data/properties/fingerprint/properties/timezone_offset",
"type": "integer"
},
"touch_support": {
"id": "/properties/fingerprint/properties/touch_support",
"items": {
"id": "/properties/fingerprint/properties/touch_support/items",
"anyOf": [
{
"type": "integer"
},
{
"type": "boolean"
}
]
},
"type": "array"
},
"user_agent": {
"id": "/properties/data/properties/fingerprint/properties/user_agent",
"type": "string"
}
},
"required": [
"available_resolution",
"ip",
"has_lied_browser",
"regular_plugins",
"hardware_concurrency",
"has_lied_languages",
"color_depth",
"session_storage",
"local_storage",
"do_not_track",
"hash",
"pixel_ratio",
"timezone_offset",
```

```
"adblock",
"has_lied_resolution",
"language",
"indexed_db",
"navigator_platform",
"cpu_class",
"user_agent",
"touch_support",
"resolution",
"has_lied_os"
],
"type": "object"
},
"keystroke": {
"id": "/properties/data/properties/keystroke",
"items": {
"id": "/properties/data/properties/keystroke/items",
"properties": {
"events": {
"id": "/properties/data/properties/keystroke/items/properties/events",
"items": {
"id": "/properties/data/properties/keystroke/items/properties/events/items",
"properties": {
"char_count": {
"id": "/properties/data/properties/keystroke/items/properties/events/items/properties/char_count",
"type": "integer"
},
"ctrl": {
"id": "/properties/data/properties/keystroke/items/properties/events/items/properties/ctrl",
"type": "boolean"
},
"delete": {
"id": "/properties/data/properties/keystroke/items/properties/events/items/properties/delete",
"type": "boolean"
},
"backspace": {
"id": "/properties/data/properties/keystroke/items/properties/events/items/properties/backspace",
"type": "boolean"
},
"kc": {
"id": "/properties/data/properties/keystroke/items/properties/events/items/properties/kc",
"type": "integer"
},
"kd_ts": {
"id": "/properties/data/properties/keystroke/items/properties/events/items/properties/kd_ts",
"type": "integer"
},
"ku_ts": {
"id": "/properties/data/properties/keystroke/items/properties/events/items/properties/ku_ts",
"type": "integer"
},
"left_shift": {
```

```
"id": "/properties/data/properties/keystroke/items/properties/events/items/properties/left_shift",
"type": "boolean"
},
"right_shift": {
"id": "/properties/data/properties/keystroke/items/properties/events/items/properties/right_shift",
"type": "boolean"
},
"special": {
"id": "/properties/data/properties/keystroke/items/properties/events/items/properties/special",
"type": "boolean"
},
"meta": {
"id": "/properties/data/properties/keystroke/items/properties/events/items/properties/meta",
"type": "boolean"
}
},
"required": [
"kc",
"ku_ts",
"ctrl",
"meta",
"left_shift",
"backspace",
"delete",
"right_shift",
"kd_ts",
"char_count",
"special"
],
"type": "object"
},
"type": "array"
},
"input_id": {
"id": "/properties/data/properties/keystroke/items/properties/input_id",
"type": "string"
}
},
"required": [
"events"
],
"type": "object"
},
"type": "array"
},
"repetition": {
"id": "/properties/data/properties/repetition",
"type": "integer"
},
"testing": {
"id": "/properties/data/properties/testing",
"properties": {
```

```
"requested_pwd": {
"id": "/properties/data/properties/testing/properties/requested_pwd",
"type": "string"
},
"requested_userid": {
"id": "/properties/data/properties/testing/properties/requested_userid",
"type": "string"
},
"submitted_pwd": {
"id": "/properties/data/properties/testing/properties/submitted_pwd",
"type": "string"
},
"submitted_userid": {
"id": "/properties/data/properties/testing/properties/submitted_userid",
"type": "string"
}
},
"type": "object"
},
"user": {
"id": "/properties/data/properties/user",
"type": "string"
}
},
"required": [
"_id",
"datetime",
"user",
"fingerprint",
"repetition",
"keystroke",
"correct"
],
"type": "object"
},
"password": {
"id": "/properties/password",
"type": "string"
}
},
"required": [
"password",
"data"
],
"type": "object"
}
```


.2 Mobile Phone Data Collection Schema

```
{
"$schema": "http://json-schema.org/draft-04/schema#",
"additionalProperties": false,
"definitions": {},
"id": "http://example.com/example.json",
"properties": {
"_id": {
"id": "/properties/_id",
"type": "string"
},
"correct": {
"id": "/properties/correct",
"type": "boolean"
},
"datetime": {
"id": "/properties/datetime",
"type": "string"
},
"fingerprint": {
"additionalProperties": false,
"id": "/properties/fingerprint",
"properties": {
"deviceID": {
"id": "/properties/fingerprint/properties/deviceID",
"type": "string"
},
"device_info": {
"id": "/properties/fingerprint/properties/device_info",
"type": "string"
},
"manufacturer": {
"id": "/properties/fingerprint/properties/manufacturer",
"type": "string"
},
"model": {
"id": "/properties/fingerprint/properties/model",
"type": "string"
},
"network_operator": {
"id": "/properties/fingerprint/properties/network_operator",
"type": "string"
},
"res_height": {
"id": "/properties/fingerprint/properties/res_height",
"type": "integer"
},
"res_width": {
"id": "/properties/fingerprint/properties/res_width",
"type": "integer"
},
```

```
"rooted_dev": {
"id": "/properties/fingerprint/properties/rooted_dev",
"type": "boolean"
},
"sim_operator": {
"id": "/properties/fingerprint/properties/sim_operator",
"type": "string"
}
},
"required": [
"res_width",
"network_operator",
"res_height",
"device_info",
"sim_operator",
"deviceID",
"rooted_dev",
"model",
"manufacturer"
],
"type": "object"
},
"keystroke": {
"id": "/properties/keystroke",
"items": {
"id": "/properties/keystroke/items",
"properties": {
"events": {
"id": "/properties/keystroke/items/properties/events",
"items": {
"id": "/properties/keystroke/items/properties/events/items",
"properties": {
"char_count": {
"id": "/properties/keystroke/items/properties/events/items/properties/char_count",
"type": "integer"
},
"del": {
"id": "/properties/keystroke/items/properties/events/items/properties/del",
"type": "boolean"
},
"kd_ts": {
"id": "/properties/keystroke/items/properties/events/items/properties/kd_ts",
"type": "integer"
},
"keycode": {
"id": "/properties/keystroke/items/properties/events/items/properties/keycode",
"type": "integer"
},
"ku_ts": {
"id": "/properties/keystroke/items/properties/events/items/properties/ku_ts",
"type": "integer"
},
```

```
"special": {
"id": "/properties/keystroke/items/properties/events/items/properties/special",
"type": "boolean"
}
},
"required": [
"ku_ts",
"char_count",
"del",
"kd_ts",
"keycode",
"special"
],
"type": "object"
},
"type": "array"
},
"input_id": {
"id": "/properties/keystroke/items/properties/input_id",
"type": "string"
}
},
"required": [
"events"
],
"type": "object"
},
"type": "array"
},
"latitude": {
"id": "/properties/latitude",
"type": "integer"
},
"longitude": {
"id": "/properties/longitude",
"type": "integer"
},
"open_app_ts": {
"id": "/properties/open_app_ts",
"type": "integer"
},
"repetition": {
"id": "/properties/repetition",
"type": "integer"
},
"sensors": {
"id": "/properties/sensors",
"properties": {
"acc_val": {
"id": "/properties/sensors/properties/acc_val",
"items": {
"id": "/properties/sensors/properties/acc_val/items",
```

```
"properties": {
"t": {
"id": "/properties/sensors/properties/acc_val/items/properties/t",
"type": "integer"
},
"x": {
"id": "/properties/sensors/properties/acc_val/items/properties/x",
"type": "number"
},
"y": {
"id": "/properties/sensors/properties/acc_val/items/properties/y",
"type": "number"
},
"z": {
"id": "/properties/sensors/properties/acc_val/items/properties/z",
"type": "number"
}
},
"required": [
"y",
"x",
"z",
"t"
],
"type": "object"
},
"type": "array"
},
"gyr_val": {
"id": "/properties/sensors/properties/gyr_val",
"items": {
"id": "/properties/sensors/properties/gyr_val/items",
"properties": {
"t": {
"id": "/properties/sensors/properties/gyr_val/items/properties/t",
"type": "integer"
},
"x": {
"id": "/properties/sensors/properties/gyr_val/items/properties/x",
"type": "number"
},
"y": {
"id": "/properties/sensors/properties/gyr_val/items/properties/y",
"type": "number"
},
"z": {
"id": "/properties/sensors/properties/gyr_val/items/properties/z",
"type": "number"
}
},
"required": [
"y",
```

```
"x",
"z",
"t"
],
"type": "object"
},
"type": "array"
},
"press_val": {
"id": "/properties/sensors/properties/press_val",
"items": {
"id": "/properties/sensors/properties/press_val/items",
"type": "number"
},
"type": "array"
}
},
"required": [
"acc_val",
"press_val",
"gyr_val"
],
"type": "object"
},
"start_test_ts": {
"id": "/properties/start_test_ts",
"type": "integer"
},
"submit_ts": {
"id": "/properties/submit_ts",
"type": "integer"
},
"testing": {
"additionalProperties": true,
"id": "/properties/testing",
"properties": {
"requested_password": {
"id": "/properties/testing/properties/requested_password",
"type": "string"
},
"requested_userid": {
"id": "/properties/testing/properties/requested_userid",
"type": "string"
},
"submitted_password": {
"id": "/properties/testing/properties/submitted_password",
"type": "string"
},
"submitted_userid": {
"id": "/properties/testing/properties/submitted_userid",
"type": "string"
}
```

```
},
"type": "object"
},
"user": {
"id": "/properties/user",
"type": "string"
}
},
"required": [
"repetition",
"start_test_ts",
"longitude",
"datetime",
"user",
"fingerprint",
"latitude",
"submit_ts",
"sensors",
"_id",
"keystroke",
"correct",
"open_app_ts"
],
"type": "object"
}
```

.3 Web and Mobile Feature Schema

```
// Json Scheme
{
"$schema": "http://json-schema.org/draft-04/schema#",
"definitions": {},
"id": "http://example.com/example.json",
"properties": {
"user_id": {
"id": "/properties/user_id",
"type": "string"
},
"repetition_id": {
"id": "/properties/repetition_id",
"type": "string"
},
"repetition_index": {
"id": "/properties/repetition_index",
"type": "integer"
},
"result": {
"id": "/properties/result",
"type": "boolean"
},
"session_id": {
"id": "/properties/session_id",
"type": "integer"
},
"total_del_keys": {
"id": "/properties/total_del_keys",
"type": "integer"
},
"total_left_shift": {
"id": "/properties/total_left_shift",
"type": "integer"
},
"total_right_shift": {
"id": "/properties/total_right_shift",
"type": "integer"
},
"total_time": {
"id": "/properties/total_time",
"type": "number"
},

"password": {
"id": "/properties/password",
"properties": {
"correct": {
"id": "/properties/password/properties/correct",
"properties": {
```

```
"down_down_time": {
"id": "/properties/password/properties/correct/properties/down_down_time",
"items": {
"id": "/properties/password/properties/correct/properties/down_down_time/items",
"type": "number"
},
"type": "array"
},
"dwell_time": {
"id": "/properties/password/properties/correct/properties/dwell_time",
"items": {
"id": "/properties/password/properties/correct/properties/dwell_time/items",
"type": "number"
},
"type": "array"
},
"flight_time": {
"id": "/properties/password/properties/correct/properties/flight_time",
"items": {
"id": "/properties/password/properties/correct/properties/flight_time/items",
"type": "number"
},
"type": "array"
},
"up_up_time": {
"id": "/properties/password/properties/correct/properties/up_up_time",
"items": {
"id": "/properties/password/properties/correct/properties/up_up_time/items",
"type": "number"
},
"type": "array"
}
},
"required": [
"dwell_time",
"flight_time",
"up_up_time",
"down_down_time"
],
"type": "object"
},
"deletions": {
"id": "/properties/password/properties/deletions",
"properties": {
"down_down_time": {
"id": "/properties/password/properties/incorrect/properties/down_down_time",
"items": {
"id": "/properties/password/properties/incorrect/properties/down_down_time/items",
"type": "number"
},
"type": "array"
},
```

```
"dwell_time": {
"id": "/properties/password/properties/incorrect/properties/dwell_time",
"items": {
"id": "/properties/password/properties/incorrect/properties/dwell_time/items",
"type": "number"
},
"type": "array"
},
"flight_time": {
"id": "/properties/password/properties/incorrect/properties/flight_time",
"items": {
"id": "/properties/password/properties/incorrect/properties/flight_time/items",
"type": "number"
},
"type": "array"
},
"up_up_time": {
"id": "/properties/password/properties/incorrect/properties/up_up_time",
"items": {
"id": "/properties/password/properties/incorrect/properties/up_up_time/items",
"type": "number"
},
"type": "array"
}
},
"required": [
"dwell_time",
"flight_time",
"up_up_time",
"down_down_time"
],
"type": "object"
}
},
"required": [
"correct"
],
"type": "object"
},
"username": {
"id": "/properties/username",
"properties": {
"correct": {
"id": "/properties/username/properties/correct",
"properties": {
"down_down_time": {
"id": "/properties/username/properties/correct/properties/down_down_time",
"items": {
"id": "/properties/username/properties/correct/properties/down_down_time/items",
"type": "number"
},
"type": "array"
```

```
},
"dwell_time": {
"id": "/properties/username/properties/correct/properties/dwell_time",
"items": {
"id": "/properties/username/properties/correct/properties/dwell_time/items",
"type": "number"
},
"type": "array"
},
"flight_time": {
"id": "/properties/username/properties/correct/properties/flight_time",
"items": {
"id": "/properties/username/properties/correct/properties/flight_time/items",
"type": "number"
},
"type": "array"
},
"up_up_time": {
"id": "/properties/username/properties/correct/properties/up_up_time",
"items": {
"id": "/properties/username/properties/correct/properties/up_up_time/items",
"type": "number"
},
"type": "array"
}
},
"required": [
"dwell_time",
"flight_time",
"up_up_time",
"down_down_time"
],
"type": "object"
},
"deletions": {
"id": "/properties/username/properties/deletions",
"properties": {
"down_down_time": {
"id": "/properties/username/properties/incorrect/properties/down_down_time",
"items": {
"id": "/properties/username/properties/incorrect/properties/down_down_time/items",
"type": "number"
},
"type": "array"
},
"dwell_time": {
"id": "/properties/username/properties/incorrect/properties/dwell_time",
"items": {
"id": "/properties/username/properties/incorrect/properties/dwell_time/items",
"type": "number"
},
"type": "array"
```

```
},
"flight_time": {
"id": "/properties/username/properties/incorrect/properties/flight_time",
"items": {
"id": "/properties/username/properties/incorrect/properties/flight_time/items",
"type": "number"
},
"type": "array"
},
"up_up_time": {
"id": "/properties/username/properties/incorrect/properties/up_up_time",
"items": {
"id": "/properties/username/properties/incorrect/properties/up_up_time/items",
"type": "number"
},
"type": "array"
}
},
"required": [
"dwell_time",
"flight_time",
"up_up_time",
"down_down_time"
],
"type": "object"
}
},
"required": [
"correct"
],
"type": "object"
}
},
"required": [
"user_id",
"result",
"session_id",
"repetition_id",
"repetition_index",
"total_del_keys",
"total_time",
"username",
"password"
],
"type": "object"
}
```

Bibliography

[1] Romain Giot, Mohamad El-Abed, Baptiste Hemery and Christophe Rosenberger. Unconstrained keystroke dynamics authentication with shared secret. Computers & Security 30 (2011), pages 427 to 445.

[2] William Bond and Ahmed Awad E. A. Touch-based Static Authentication Using a Virtual Grid. IH
textbf&MMSec'15, June 17–19, 2015, Potland, OR, USA.

[3] Xuan Huang, Geoffrey Lund, and Andrew Sapeluk. Development of a typing behaviour recognition mechanism on Android. IEEE 2012.

[4] Kenneth Revett, Sérgio Tenreiro de Magalhães, Henrique Santos. Data Mining a Keystroke Dynamics Based Biometrics Database Using Rough Sets. IEEE 2005.

[5] Mr N. Pavaday, and Associate Prof. Dr. K.M.S.Soyjaudah, Investigating performance of Neural Networks in authentication using keystroke dynamics. IEEE 2007.

[6] Pin Shen Teh, Andrew Beng Jin Teoh, Thian Song Ong, Han Foon Neo. Statistical Fusion Approach on Keystroke Dynamics. IEEE 2008.

[7] Romain Giot, Mohamad El-Abed and Christophe Rosenberger. Keystroke Dynamics Authentication For Collaborative Systems. IEEE 2009.

[8] Romain Giot, Mohamad El-Abed and Christophe Rosenberger. Keystroke Dynamics With Low Constraints SVM Based Passphrase Enrollment. IEEE 2009.

[9] Noor Mahmood AI-Obaidi and Mudhafar M. AI-Jarrah. Statistical Median-Based Classifier Model for Keystroke Dynamics on Mobile Devices. IEEE 2016.

[10] Margit Antal, László Zsolt Szabó, Izabella László. Keystroke dynamics on Android platform. Procedia Technology 19 (2015) pp. 820 – 826.

[11] Kevin Killourhy and Roy Maxion. Why Did My Detector Do That?! Predicting Keystroke-Dynamics Error Rates. RAID 2010, LNCS 6307, pp. 256–276.

[12] Salima Douhou and Jan R. Magnus. The reliability of user authentication through keystroke dynamics. Statistica Neerlandica (2009) Vol. 63, nr. 4, pp. 432–449.

[13] Matthias Trojahn and Frank Ortmeier. Toward mobile authentication with keystroke dynamics on mobile phones and tablets. IEEE 2013.

[14] Sheng Y, Phoha VV, Rovnyak SM. *A parallel decision tree-based method for user authentification based on keystroke patterns*. IEEE Trans Syst Man Cybern B Cybern, 2005.

[15] Margit Antal, Laszlo, Izabella. *Keystroke Dynamics on Android Platform*. 2014.

[16] Daniel Busckek, De Luca, Alt. *Improving Accuraccy, Applicability and Usability of Keystroke Biometrics on Mobile Touch screen sevices*. 2015.

[17] Huang, Lund. *Development of typing behaviour recognition mechanism on Android*. IEEE 2012.

[18] Kevin Killourhy and Roy Maxion. *Why Did My Detector Do That?! Predicting Keystroke-Dynamics Error Rates*. RAID 2010, LNCS 6307, pp. 256–276.

[19] William Bond and Ahmed Awad E. A. *Touch-based Static Authentication Using a Virtual Grid*. IH&MMSec'15, June 17–19, 2015, Potland, OR, USA.

[20] Margit Antal, László Zsolt Szabó, Izabella László. Keystroke dynamics on Android platform. Procedia Technology 19 (2015) pp. 820 – 826.

[21] Xuan Huang, Geoffrey Lund, and Andrew Sapeluk. Development of a typing behaviour recognition mechanism on Android. IEEE 2012.

[22] Kenneth Revett, Sérgio Tenreiro de Magalhães, Henrique Santos. Data Mining a Keystroke Dynamics Based Biometrics Database Using Rough Sets. IEEE 2005.

[23] Salima Douhou and Jan R. Magnus. The reliability of user authentication through keystroke dynamics. Statistica Neerlandica (2009) Vol. 63, nr. 4, pp. 432–449.

[24] Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer Peter Reutemann, Ian H. Witten. The WEKA Data Mining Software: An Update.

[25] Armin Ronacher, "Flask Web Deleopment, One Drop At Time. 2015

[26] DigitalGov.org, "Gov Analytics Breakdown #1 – Browsers: Chrome Takes the Cake", 2017

[27] Md Liakat Ali, John V. Monaco, Charles C. Tappert and Meikang Qiu. Keystroke Biometric Systems for User Authentication. Journal of Signal Processing Systems, 2016