Keystroke Dynamics Applied to Authentication of Network Users

Luis Adrián Lizama Pérez[1], José Guadalupe Aguilar[2]

Universidad Juárez Autónoma de Tabasco
División de Informática y Sistemas
[1] luis.lizama@dais.ujat.mx
[2] jose.aguilar@ujat.mx

Abstract

A method to authenticate users based on statistics of user's biometric keystroke dynamics is described in this paper. The authentication model is based on the comparison of biometrical templates. Each template is constructed using the time intervals that a specific user employs for the events press-release key and release-press key. These events are measured with a precision of four digits in order to compare the similarity of templates through statistical functions of dispersion. The acceptance rate and similarity rate were compared to accept or reject a user. The false acceptance rate (FAR) and the false rejection rate (FRR) were computed during the tests, on both, the local authentication and the network authentication using keyboards in different computers connected to the authentication server. A value of 0.0% for the first was obtained and an average of 35% for the second. A simple adaptation mechanism was incorporated. The viability of its implementation on smaller keyboards is discussed. Also, the inclusion of the biometric identity of each user in a portable USB is suggested.

1. Introduction

Biometric techniques have progressed since the 1990s, including the evaluation of two user characteristics: the release-press key and the press-release key time intervals. The press-press time has also been added, and this is the time elapsed between pressing one key and pressing the next key, that is to say, the time between releasing one key and pressing the next [1][2]. The techniques that have been used to classify users vary from diffuse models and radial based function (RBF) network classifiers, to classifiers based on statistical models [3]. Different studies [4],[5],[6],[7] have classified users using a model based on measuring typing time with a precision of milliseconds, or hundredths of seconds [1][2].

2. Methodology

The basic part of the authentication is an interface capable of collecting the typing times of each user at the moment of authentication, as well as for creating templates for the first time at the moment or registration. Keyboard events detection in high level programming language is not a difficult task as they incorporate routines capable of handling keyboard events such as pressing or releasing a key. The user's behavior on the keyboard is measured considering the following characteristics:

a. The time elapsed between pressing a key and releasing it. This is called the press–release event (Figure 1.a).
b. The time elapsed between releasing the key and pressing the next. This is called release – press event (Figure 1.b).

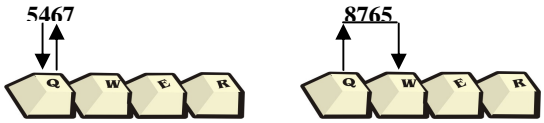


Figure 1.
a) Press– Release Event **b) Release – Press Event**

The interface must provide a set of times matched to a sequence of written characters. The necessary elements for the development of this interface and of the biometric application are routines for the detection of typing-keyboard events, a timer with a precision of four digits to differentiate the times of each user, and the normalization of these times to authenticate over the network.

The management and implementation of the timers strongly depends on the operating system used. Microsoft Windows was used for this study. Among the different programming options that may use a timer, are:

1. Timer tools provided by high-level languages such as Delphi, Visual Basic or Java. These tools work in milliseconds.
2. Windows has an API function called GetTickcount that, when called, provides the time in milliseconds that Windows has been active.
3. Java has the System.currentTimeMillis() function that provides time in milliseconds, this is taken from the system.
4. QueryPerformanceCounter is a Windows API function that returns the processor cycles that have passed since Windows was activated, with a precision of eleven digits.

Table 1 shows a comparison between these four timing options and the number of digits provided for the measurement of each event.

Table I. Time functions comparison

Time functions	Press - Release	Release - Press
Timer	2 digits	2 digits
GetTickcount	2 digits	2 a 3 digits
System.currentTimeMillis()	2 digits	2 a 3 digits
QueryPerformanceCounter	5 digits	4 a 5 digits

Since the timer depends directly on the hardware, the speed at which it increases varies from one computer to another, and as the basis for the keystroke dynamics is the typing speed of the user, the time should be similar each time he is authenticated. In order to normalize the times obtained for each user, the min/max and the mean normalization techniques are proposed, the second being the better one to solve this problem [7].

To show the normalization process, a simple experiment was carried out on two computers, the first equipped with a 2.79 GHz Pentium® IV processor and the second with a 1.8 GHz Pentium® IV processor. The string used was the word BIOMETRIA (Figure 2).

	B-I	I-O	O-M	M-E	E-T	T-R	R-I	I-A	Media
2.79 GHz.	654	753	970	563	614	418	770	559	662.63
1.8 GHz.	264	454	408	409	285	275	420	261	347

Figure 2. Time samples in different computers for the word BIOMETRIA.

The speed difference between both processors was 0.99 GHz. So, greater times were obtained with the 2.79 GHz processor, as it is shown in Figure 4. By analyzing those times, one could say that the samples are not from the same user. The mean normalization process is then applied. The mean for each sample was 662.63 for the 2.79 GHz processor and 347 for the 1.8 GHz processor. Next, each tn time was divided by its corresponding mean (Figure 3).

	B-I	I-O	O-M	M-E	E-T	T-R	R-I	I-A
2.79 GHz.	0.986983588	1.13638936	1.463874741	0.849651009	0.926617619	0.630824373	1.162044897	0.843614412
1.8 GHz.	0.760806916	1.308357349	1.175792507	1.178674352	0.821325648	0.792507205	1.21037464	0.752161383

Figure 3. Normalisation process results.

Figures 4.a and 4.b show the normalization process of the samples. In Figure 6.a, the lines generated by the tn not normalized times are separated, although a certain similarity with respect to the curves may be observed. When the normalization process is applied, the value for the times changes for both samples. When normalized by the mean, the shape of the curves is not affected so that each generated line is similar to the original, but with normalized times (Figure 4.b). These are now intercepted within the same range of values and a similarity between the two lines is observed.

2.1. Keystroke dynamics comparison model

This model receives a list of times that may include those of the press-release and release-press events, as well as the template with which the new times will be compared.

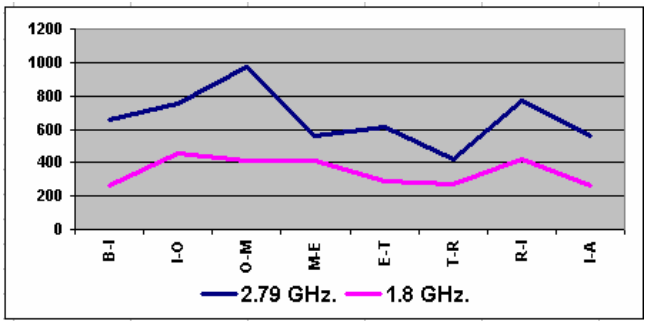
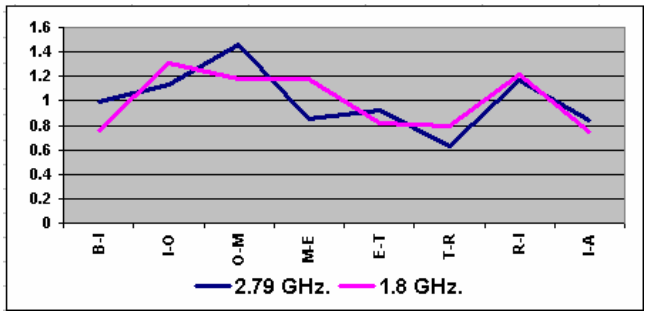


Figure 4. a) Times without normalisation.



b) Normalised times.

The return parameter that the model will send is a percentage that is called Similarity Percentage (SP). This will indicate the percentage at which the new times are similar to those in the template. Another percentage called Acceptation Rate (AR) will indicate the minimum percentage required from the user to be accepted on the

system. The templates of the press-release and release-press events are structured by columns that contain the times of each event, and by lines that contain the number of samples taken from the user (Figure 5).

	W-O	O-O	O-D	D-Y	Y-S	S-A	A-R	R-G	G-E
M-1	1.08644	1.58053	0.66606	1.14923	0.54049	1.22839	1.30755	0.67698	0.76433
M-2	1.16605	1.48887	0.67625	1.21336	0.42301	1.10761	1.45826	0.86827	0.59833
M-3	2.21196	1.34217	0.76533	0.91067	0.58592	0.84481	1.22407	0.68584	0.42922
M-4	0.9659	1.0431	1.69644	0.79079	0.37469	1.54017	1.21067	0.91506	0.46318
M-5	1.28182	1.46455	0.57	1.28182	0.47455	1.10182	1.46727	0.84273	0.51545
M-6	1.09151	1.462	0.44459	1.37935	0.38759	1.29386	1.519	0.77517	0.64693
M-7	1.56026	1.33489	0.85938	0.91882	0.47056	1.22097	1.4191	0.70088	0.51513
M-8	1.1543	1.46178	0.54439	1.46934	0.4965	1.17446	1.35844	0.63512	0.70568
M-9	1.36951	1.56331	0.55039	1.19121	0.65891	0.86305	1.49096	0.64341	0.66925
M-10	1.36457	1.41101	0.76612	0.88994	0.69647	1.17885	1.37231	0.69132	0.62941

Figure 5. Template for the release-press event, sampled from the word WOODYSARGE.

The times in each column in Figure 5 have relatively similar values, as they are samples taken from one user. The problem is to determine whether a new time sample (see Figure 6) has a significant SP with respect to the times on the template, in such a way that it may be concluded that it is the same user.

	W-O	O-O	O-D	D-Y	Y-S	S-A	A-R	R-G	G-E
Nueva	0.83535	1.45678	1.78329	1.4352	0.6273	1.2167	1.51923	0.91506	0.51545

Figure 6. New time sample.

What is needed is a function that indicates whether a new time is similar to a template column, as well as to determine the similarity of all the times in each column in the template in order to be able to compare a new time. This function should indicate, in a numerical way, the spread degree of the times of each sample stored in the template, and the function that does this is the standard deviation (S). S indicates the distance between the average and the results.

If S of each column is calculated, then a number that indicates the degree of deviation of the user is obtained at the moment of recording the samples for the template on a key. Once calculated S of each column of the template, is necessary to know how much the new sample has deviated. The mean of each column of the template is taken as a reference. A new S, called S', is calculated based on the mean of each column of the template and the new time of the corresponding column. If the user is the right one, most of the S columns should be greater than S'.

Another statistical function used in this model is the variation coefficient (VC)[8][9] that indicates the spread of a set of data in terms of percentages. As it was mentioned before, what is required is a percentage that indicates the degree of similarity between new time samples and the time samples recorded on the template. Since the comparison is between the times on the template and the new times, the same formula is applied as for S, although the new times are estimated with respect to the mean. The mean of the new sample will be the sum of the new time and the mean of the template divided by two. This will be called mean' and is applied to each column (Figure 7).

	W-O	O-O	O-D	D-Y	Y-S	S-A	A-R	R-G	G-E
M-1	1.08644	1.58053	0.66606	1.14923	0.54049	1.22839	1.30755	0.67698	0.76433
M-2	1.16605	1.48887	0.67625	1.21336	0.42301	1.10761	1.45826	0.86827	0.59833
M-3	2.21196	1.34217	0.76533	0.91067	0.58592	0.84481	1.22407	0.68584	0.42922
M-4	0.9659	1.0431	1.69644	0.79079	0.37469	1.54017	1.21067	0.91506	0.46318
M-5	1.28182	1.46455	0.57	1.28182	0.47455	1.10182	1.46727	0.84273	0.51545
M-6	1.09151	1.462	0.44459	1.37935	0.38759	1.29386	1.519	0.77517	0.64693
M-7	1.56026	1.33489	0.85938	0.91882	0.47056	1.22097	1.4191	0.70088	0.51513
M-8	1.1543	1.46178	0.54439	1.46934	0.4965	1.17446	1.35844	0.63512	0.70568
M-9	1.36951	1.56331	0.55039	1.19121	0.65891	0.86305	1.49096	0.64341	0.66925
M-10	1.36457	1.41101	0.76612	0.88994	0.69647	1.17885	1.37231	0.69132	0.62941
S	0.35625	0.15339	0.35408	0.22991	0.10929	0.20134	0.10831	0.09995	0.10948
Media	1.32523	1.41522	0.75389	1.11945	0.51087	1.1554	1.38276	0.74348	0.59369
C.Var	26.88%	10.84%	46.97%	20.54%	21.39%	17.43%	7.83%	13.44%	18.44%
Nueva	0.83535	1.45678	1.78329	1.4352	0.6273	1.2167	1.51923	0.91506	0.51545
S'	0.3464	0.02939	0.72789	0.22327	0.08233	0.04335	0.0965	0.12133	0.05532
Media'	1.08029	1.436	1.26859	1.27733	0.56908	1.18605	1.451	0.82927	0.55457
C.Var'	32.07%	2.05%	57.38%	17.48%	14.47%	3.65%	6.65%	14.63%	9.98%

Figure 7. Calculation applied to the template.

Thus far calculations with percentages have been carried out to see how clustered or dispersed the data on the template are when a user types, as well as the percentage of deviation at the moment of authentication with respect to the template mean. A bigger percentage takes a greater possibility of variation in a particular column. In the previous figure, the A-R column has the least variation percentage. This means that when the user releases the A key and presses the R key, his timing is regular. However, the O-D column which has the greatest percentage indicates there is no regularity when releasing the O key and pressing the D key, as this is sometimes done fast and sometimes very slowly.

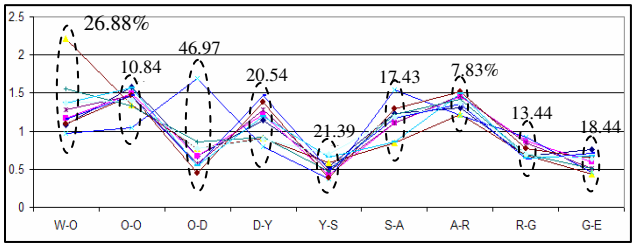


Figure 8. Times for the template and its respective variation percentage.

The Figure 8 shows that the greatest VC values (26.88 and 46.97%) were due to only one time, respectively, that of 2.29644, as the other times are between 1.00 and 1.52. In the smallest VC (7.83%), it can be seen that the points are near and uniformly distributed with times that vary

between 1.20 and 1.52. No time lies outside this range and this produces a very small VC. Generally, the VCs that tend to be bigger are originated by one or two points outside the range. Columns Y-S, A-R, R-G and G-E have points that are grouped within a small range, and these results in smaller percentages.

So far we have a model that indicates the percentages of variation on the points of the template, and the percentage of variation of the new times with respect to the mean of the template. The area of acceptation of the template that has been used as an example is shown in Figure 9.

In order for a user to be authentic, the new times must lie below the limit line, in contrast with the case of the maximum and minimum values that had two limiting lines. If the variation coefficient of the new sample were plotted, one would find that, as it belongs to the same user, it would lie below the limit shown in the last figure, with the exception of three points that lie above it. The question here is, are these three points enough to reject the user? Or, are the seven points that lie below the limit enough to decide that the user will be accepted? Initially 60% is established as the limit to accept a user.

A general average called Similarity Percentage (SP) was calculated for this:

$$SP = \frac{\#VC > VC'}{\#Columns} \qquad (1)$$

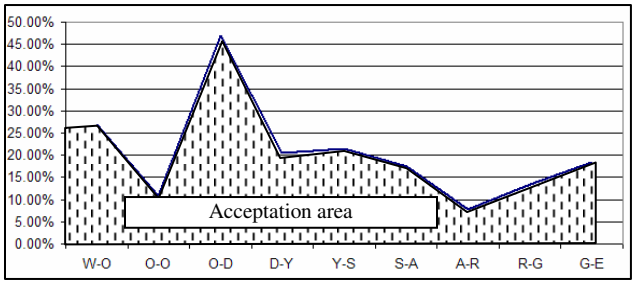


Figure 9. Variation coefficient of the template showing the limit of acceptation of the user.

Where:

#VC>VC' is the number of percentages in which the VC of the template was greater than the VC' of the new time sample with respect to the template mean.

#Columns is the number of columns on the template (in the case of a press-release event template, the number of columns is equal to the length of the typed string, and in the case of a release-press event it is equal to the length of the typed string minus one).

The calculation of this percentage is the final result of this model that provides a SP for the samples. The acceptance or rejection of a user depends on this percentage and on the manner in which it is interpreted and compared. As it was previously stated, the SP value must be greater than 60% in order for a user to be accepted, as observations and tests of the model established that most of the non-authentic users recorded values below 50%, although some authentic users did not reach 60%.

It was decided to compare the percentage in a dynamic way and that it should depend directly on the behavior of the user at the moment of creating his template. The VC average of each column of the template was used as a reference, but why this value? As previously mentioned this percentage indicates the deviations on the template and for each column. This percentage varies from greater in some cases to smaller in others. In the example case in Figure 10, the lines can be seen to follow a pattern where the points do not deviate much, as on average they only deviate one to two points for each column. This means that when this user authenticates himself again, he must generate a line similar to the template to be accepted, as the lines on the template have a high degree of similarity. The level of acceptance might be lowered for this user as it is easy for him to make a mistake and originate a line that is more different than expected. Assuming that no other user types as he does and as his record shows he has a well established keystroke dynamics, there should be no problem in lowering the acceptance level for him.

If the VC is large, the lines of the template do not have a considerable degree of similarity. Thus, when the user authenticates, he would have a great range in which his times could fall, and the AR could increase to avoid FAR. Similarity could be considered as the inverse of variability. So, AR is calculated as the complement of the mean VC, AR=1-VC. If the AR is calculated for the user of the example, a value of 79.58% is obtained, and this means that for this user to be accepted he must obtain a SP value of 79.58% with respect to the template mean.

2.2. Adaptation mechanism

The adaptation mechanism implemented in this study is simple and consists of calculating the VC based on the mean for each sample on the template, as if one were dealing with a new sample, obtaining the one with the greatest CV and replacing it with the new authentication sample, when and only when this has a smaller CV. If not, the template remains the same.

2.3. Portable biometric identity

The templates are structured on text files with a maximum size of 5 KB per template. This file is included in the executable of the application that is produced by an algorithm as resource files.

When it is incorporated into the executable, the template may not be modified and, thus, it cannot be altered by an intruder and the authentic user cannot be replaced by another. The size of the executable produced by the

authentication algorithm in the proposed prototype was 1.5 MB, in a window environment. This size may be reduced considerably when changed to text mode.
It is of interest to apply this software to smartcard technology but, due to its broad use and versatility, the application is installed in a USB that acts as a hardware biometric key for the control of access to systems and applications in the web. When the USB is recognized by the operating system, the application blocks it and requires the user to be authenticated. The authentication is based on the keystroke dynamics of the user as the template already recorded in the application.

3. Tests

Tests were carried out on three groups of people:

1. Sample Study Group (SSG). This group of ten persons resulted from a sampling of workers, and included persons from 24 to 35 years of age with positions of programmers, accountants, secretaries and administrators.
2. University Group (UG). This group included 200 accounting, education and communication sciences students.
3. Various Group (VG). This group consisted of 20 persons of different ages, occupations, and typing skills.

Each person that tested the biometric application established two phrases: the first as login, the second as password. It was recommended that, for the first phrase, the user's full name should be used as it is believed that it is easy to write. It is logical to think that people are familiar with their names. For the extraction of the typing characteristics, each person typed the login ten times, and then the password ten times. The purpose of the tests was to establish FAR and FRR on both the local authentication and the network authentication using keyboards in different computers connected to the authentication server.

4. Results

A value of FRR of 5% was obtained for SSG with 10 users and 10 tries each. This indicates that from a total of 100 tries, only five were incorrectly rejected. Five users' passwords were given to these 10 users in order to fake an intruder's attempt to login, and a FAR of 0% was obtained. An FRR of 60% was obtained for this group in the case of the authentication on the network, when considering the evaluation of the press-release and release-press events. This was followed by an evaluation of the time of the release-press event, for which a FRR of 19% was obtained. The FAR on the network was 0%.
In the case of UG, each person was authenticated 10 times. A FRR of 26% was obtained, meaning that only 523 tries out of 2000 were wrong. The FAR for this group was 0%. The authentication of this group on the network was carried out with three types of computers, and a EFR of 36% was obtained. This is to say that 2182 tries out of 6000 were incorrectly rejected. The FAR was 0%.
For VG, twenty users were authenticated 10 times with a FFR of 25%, with 50 out of 200 tries incorrectly rejected. The FAR for this group was 0%.

4.1. Analysis of results

In accordance with the experiments carried out on the three test groups and with the observations that were recorded, the following was determined:
The phrase with which most users are familiarized is their own full name, and the percentage of false rejection was less than 0.2 (20%) for the users that used it as their login.
The most important issue in the authentication of a user is the design of the template for which he must type at a normal speed and avoid unnecessarily pausing between keys. If a user chooses a phrase not familiar to him, a decrease in typing speed was observed. However, particular typing characteristics could still be noticed. If a user chooses a phrase with less than ten characters, the FRR increased considerably. The adequate length for a phrase in this model that decreases the FRR is of 15 to 30 characters.
SSG recorded the lowest percentage of FRR, and one of the causes for this was that the tests were individual, and instructions were given to each user. UG did the test as a group, this resulted in the users being distracted and their keystroke dynamics failed when creating the templates.
For authentication, it is advisable that the press-release and release-press times are recorded together at the moment when the user's template is created and on the same computer where the authentication took place.
In the case of authentication on a different computer, it is advisable to exclude the press-release times from the comparison, as these vary considerably depending on the softness of the keys. However, the release-press event is sufficient to carry out the comparison.
The average of the SP obtained for impostors is 35%, and the general percentage of FAR is 0% as at least 60% of SP is necessary to be accepted by the system. This average of 35% SP for impostors makes it possible to lower the AR of 40% to 50%. This will decrease the FRR obtained in the tests and will guarantee that the FAR will not increase.
The adaptation mechanism replaces samples with bigger VCs when a user improves his typing skills and the FRR decreases. The four digits to measure the press-release and release-press times were relevant for the results obtained for the FRR and FAR. In general, a 0% FAR was obtained and this is the strength of this method. This is to say that the method does not accept an incorrect user

and a rejection of a correct user has the only consequence that he should try again

4.2. Comparisons with previous studies

One way to measure the efficiency of the authentication model is by comparing the results obtained here with those of other studies related to keystroke biometrics. Table II summarizes several studies on keystroke biometrics, and includes the four studies with the lower percentages of FRR and FAR. The best results were obtained with SSG, with the error percentages below those of the other studies, and the case of the experiment on the network is above only one other study [3].

Table II. Comparisons with previous studies

Authors Ref.	Simples	S.A.*	FRR	FAR
[1]	2560	2	0.35	0.29
[2]		2	0.18	0.23
[3]	Varied	2	0.14	0.28
[4]	130 – 180	1	0.26	0.05
[5]		2	0.11	0.09
[6]		2	0.08	0.03
[7]	Varied	2	0.06	0.02
This work				
SSG	500	2	0.05	0
SSG on network	400	2	0.19	0
UG	7000	2	0.26	0
UG on network	18000	2	0.36	0
VG	200	2	0.25	0

*Number of sequences or phrases used

5. Future studies and conclusions

This authentication model is designed for keyboards with 101-104 keys. However, the viability of its implementation on smaller keyboards such as those of cash registers, PDA, smartphones and mobile phones is being investigated. It is particularly of interest to determine the minimum group of keys with which it is possible to recognize a user with certainty. The following factors need be considered for this:

1. The number of fingers used to strike the keys on a keyboard.
2. The number of keys needed to type an authentication phrase.
3. The possibility of recognizing a user with a short phrase (less than 10 characters).
4. A timer available where the authentication is required.

Tests carried out with a numerical keyboard showed that the identification of a user with this model has a practical approach. However, the authentication phrase needs to have at least 15 characters and the user must learn it, and this leads to the need for a user to be able to be authenticated with a shorter phrase. Authentication on a mobile phone with a normal phrase might be possible using typical writing. That is, the writing of messages on mobiles where the events will continue to be release-press. However, in order to write a five-character phrase it is necessary to press more than five keys as each key represents three letters of the alphabet. It may be said that one could type a short phrase by pressing and releasing a considerable number of keys to be evaluated. The tests provided satisfactory results in obtaining a 0% error rate for the FAR and an average of 35% for the FRR. This technique represents a technology of low cost authentication as it does not require additional hardware and uses the traditional keyboard to measure biometrics. As it is small, it may be installed in a USB that functions as a biometric hardware key to control access to systems and web applications.

6. References

[1] Araújo Lizárraga, Sucupira Jr., Yabu-uti y Ling. "Autenticación personal por dinámica de tecleo basada en lógica difusa" Universidad Estatal de Campinas (UNICAMP).

[2] D. Umphress and G. Williams, "Identity Verification Through keyboard Characteristics" International Journal Man-Machine Studies, Academic Press, 1995.

[3] W.G. de Ru and J.H.P. Eloff, "Enhanced Password Authentication through Fuzzy Logic" IEEE Expert / Intelligent Systems & Their Applications, Noviembre/Diciembre 1997.

[4] Acevedo Daniel, Glemarys Hernández y Eugenio G. Scalise P. "Identificación de Usuarios Basado en el Reconocimiento de Patrones de Tecleo" Universidad Central de Venezuela, Facultad de Ciencias 2000.

[5] Enzhe Yu, Sungzoon Cho, "Keystroke dynamics identity verification problems and practical solutions" Department of Industrial Engineering, College of Engineering, Seoul National University, 2004.

[6] Obaidat M. S. "Keystroke dynamics based Authentication" Monmouth University Applied Science University 2002.

[7] Marino Tapiador Mateo. "Biometría de tecleo, autenticación de usuarios" Ingeniería Informática, Universidad Autónoma de Madrid, Mayo del 2000.

[8] A.M Montiel, F. Rius y F. J. Baron, "Elementos básicos de estadística económica y empresarial", 3ª Ed. Prentice Hall 2004.

[9] A. Rubio 2005. Estadística experimental, práctica, útil y sencilla. Chihuahua, México.

[10] IAfB International Association for Biometrics and International Computer Security Association (ICSA). "Glossary of Biometric Terms" 2005.

[11] Merlat Máximo, Paz Gonzalo, Sosa Matias, Martinez Marcelo. Seguridad Informática, Hackers 1998.

[12] Fabian Monrose, Aviel D. Rubin, "Keystroke Dynamics as a Biometric for Authentication" New York University, New York, NY 1999.

[13] Sue Berg "Glossary of Computer Security Terms. Technical Report NCSC-TG-004" National Computer Security Center, Octubre 1988.