CUMULATIVE AND RATIO TIME EVALUATIONS IN KEYSTROKE DYNAMICS TO IMPROVE THE PASSWORD SECURITY MECHANISM

Giovanni Saggio, Giovanni Costantini, Massimiliano Todisco

Abstract. The password mechanism is widely adopted as a control security system to legitimate access to a database or a transaction content or computing resources. This is because of the low cost of the mechanism, the software routine simplicity, and the facility for the user. But the password mechanism can suffer from serious vulnerabilities, which have to be reduced in some way. An aid comes from the keystroke dynamic evaluation, which uses the rhythm in which an individual types characters on a keyboard. It has been demonstrated how the keystroke dynamics are unique biometric template of the users typing pattern. So, the dwell time (the time a key pressed) and the flight time (the time between "key up" and the next "key down") are used to verify the real user's identity. In this work we investigated the keystroke dynamic already reported in literature but with some differences, so to obtain additional benefits. Rather than the commonly adopted absolute times (dwell and fly times), we deal with cumulative and ratio ones (i.e. sum and ratio of dwell and fly times), taking into account that the latest are times which do not change even if the user's typing style evolves with practice.

INTRODUCTION

There are different methods of authentication, classifiable in two main categories: *natural properties of an individual*, i.e. physiological features such as fingerprints, iris, retina, face geometry, voice, etc. (Ratha et al., 2001; Luis-Garcia et al., 2003); *artificial measures*, either physical objects held solely by the user (e.g. magnetic card, key, badge, token, etc.), and/or personal private information known only to the authentic user (e.g. password, the maiden name of his/her mother, spouses, etc.) (Ahituv et al., 1987). Practically these methods rely upon what a person *possesses* or *is* or *knows*. Among the artificial measures, into the category of what a person knows, there is the "password mechanism", which remains the most predominant and widely adopted method all over the world. This is mainly due to the very low cost of verification in terms of CPU time and software routine simplicity, the installation and maintenance inexpensiveness, and the simplicity of the mechanism which must be learned and adopted by the user. Passwords are employed to identify authorized users where multiple individuals may have legitimate access to a database or a transaction content or to a set of computing resources. But it is commonly known that the password system suffers from strengths and vulnerabilities as discussed and detailed (Ahituv et al., 1987; Jobusch & Oldehoeft, 1989; Bishop & Klein, 1995). The deficiencies and vulnerabilities of the password from a security point of view have many reasons: too short password is easy to guess or discover; too long password is easy to forget; wrote password can be stolen or copied; somebody can watch the user typing so the secret falls; hackers can intrude the system with a trojan horse and stored password can be revealed. In order to overcome, at least partially, the mentioned deficiencies, user is often asked to adopt precautions, such as to refrain from using words related to his/her birthday, relatives, parents, etc., to include some numbers so to increase the set of characters to choice from, to change the password itself as often as possible. These precautions can lower but cannot eliminate the vulnerability of the password mechanism; moreover, sometimes they are not always practicable, as it happens, for instance, for the password assigned by a bank to his/her client to access to cash dispenser machine (bankautomat or bancomat) facilities.

METHODS

Since the larger the total number of possible passwords the more secure the login system, if a password consists of seven alpha-numeric characters, the total number of combinations is 36^7 so, if the password is tried randomly, it is really hard and time consuming to discover it. But this is only a hypothetic assumption since hackers do not randomly select a password, but limit their attempts on a real subset of that great number of combinations, basing on some logic hypothesis. The more these hypothesis are true and useful to the aim, the more limited the subset, the more easy for the hackers to reveal what should be unknown. So, it can be not strictly true that increasing the number of password characters we obtain a more secure protection system.

Generally speaking, for an authentication system the requirements are: reliability, easy to use, user acceptance, facility of implementation, and cost. In this paper we propose, analyze and investigate a method to increase the security associated to the password login system. It maintains simplicity and inexpensiveness, and increases the reliability of the overall system, being unobtrusive and totally "transparent" to the user, who is not requested to change his/her habits. In a certain way, our proposal merges together personal private information and physiological features. In fact we considered, in this work, the *typing speed* (a natural property of an individual) in writing each symbol of the password (a personal private knowledge) that is strictly dependent on each singular person. Particularly, we focused our attention in recording each single time associated to how long each single key is held pressed (*dwell time* or *keystroke duration*), and the elapsed time between two following pressed keys (*flight time,* also known as *digraph latency*), so investigating what it is known as *keystroke dynamics* or *typing rhythms*. We investigated here the *keystroke dynamic* already adopted and reported in literature to improve the password security mechanism, but with some differences with respect to other authors, so to obtain some additional benefits.

In literature there are works devoted to the keystroke dynamic, especially in the last two decades. Douhou & Magnus (2009) present a summary of the most relevant of those works.

In order to investigate how the keystroke dynamic can quantitatively improve the password security mechanism, some tests have to be performed. Some variables were considered, such as the numbers of the volunteers to be involved, the number of the passwords to be repeated, the number of the repetition times, and the elapsing time between two consecutive repetitions.

We believe that the mere number of participants cannot be significant if not associated to all the other variables. This hypothesis is validated by the fact that just the number of volunteers reported in literature has a relevant variation, since it ranges from only 4, (Ke et al., 2005), to 7 (Gaines et al., 1980), to 15 (Lau et al., 2004), to 17 (Legget & Williams, 1988), to 19 (Gingrich & Sentosa, 2008), to 50 (Revett et al., 2007), to 63 (Monrose & Rubin, 2000), even till 1254 (Douhou & Magnus, 2009) though not all of them realized a complete test. In any case, the most common reported number wander around 10÷15.

For our tests we decided to involve 16 volunteers since it appears to be the most convenient number in association to the testing procedure later described.

Moreover, we had to define which password to be considered, in number of characters. Latest works deal with single words of 7 characters (Douhou & Magnus, 2009), or ranges between 6÷10 characters (Yu & Cho, 2004), or 6÷15 characters (Revett et al., 2007), or with sentences of 43 (Lau et al., 2004), 537 (Leggett et al., 1991), 683 characters (Bergadano et al., 2002).

The data

In our work, we decided for two passwords to be repeated. These passwords were a "word" and a "number". This was because the "words" are commonly adopted to login personal computers, while the "numbers" are utilized to access a cash dispenser machine by a numeric keypad. A "word" as a password is commonly directly chosen by the user, while the "number" for cash machines is decided by the bank service in lieu of the user. For the most of people the ability to remember an expression decrease rapidly with increasing the number of characters, and it has long been accepted that people can remember readily an expression of about seven characters in length (Miller, 1956). So we decided as a seven-character length the word "*special*", with all lowercase symbols. For the most common cash dispensed machines, the numbers to access are of five-symbol length, so we decided for the number "*12057*". It is important to notice that the number was digitized only on the numeric lateral keypad of extended keyboards, just to mimic the PIN authentication on the board of common cash dispensed machines. The "word" and the "number" were repeated 30 times each, every repetition representing *one trial*, once a day, for ten consecutive days, always at the same hour and with the same boundary conditions (pc, desk, chair, room, temperature, light).

We then could count on 9,600 trials in total (16 participants, by 2 times 30 repetitions, by 10 days). Since the *flight times* before the first and after the last typed character/number have no significance, we had 6 flight times for each test of the word "*special*" and 4 for the number "*12057*", while 7 and 5 *dwell times* respectively. These make a total of 96,000 *flight times* and 115,200 *dwell times*. If a typing error is made, our home-made recording routine simple did not consider the password's attempt and, as it is the usual case of a real password input, the participant was asked to repeat the keystroke.

The participants

The experimental protocol of the tests was approved by the local ethics committee. The informed consents were obtained from all the 16 participants, but with differences within two groups of 8. In particular the first 8 volunteers (subjects from 1 to 8) were informed in details regarding what we were recording, the reason why and the final utilization of the data. Let's call this group *informed users*. The second group of 8 volunteers (subjects from 9 to 16) were informed of everything except that they accepted to know the final usage of the data only at the end of the entire procedure. Let's call this group of *partially informed users*. Our aim was so to verify if an informed user can influence the final results with respect to an uninformed one.

The 16 participants consisted of 10 men and 6 women, aged 29÷47 (mean 35.37; standard deviation 5.45), all from the same country (Italy) and of the same native language (Italian), but known English as a second language. In any case the English chosen word (*special*) was quite similar to the corresponded Italian one (speciale).

The classification method

In addition to the main variables, it is fundamental to decide the classification method that can be more suitable for our aim.

Approaches based on Neural Networks have been undertaken (Brown & Rogers, 1993; Alexandre, 1996). Other works reported classifiers based on Probabilistic Neural Network (Revett et al., 2007), Auto-Associative Neural Network (Cho et al., 2000), Fuzzy algorithms (Huissien et al., 1989), Bayesian-like classifier (Monrose & Rubin, 2000), Support Vector Machines (Scholkopf et al., 2000; de Oliveira et al., 2005; Sang et al., 2004; Sung & Cho, 2006). According to our experience, here we adopted a system based on a multi-class SVM classifier.

Metric Proposal

The utilization of the keystroke dynamics has some recognized technological bottleneck, since it can be unstable and unreliable as it can vary from time to time for the same subject. So some proposals have been investigated till now. In Gingrich & Sentosa (2008) a list of the most interesting.

In literature it was investigated the so called *distance metric* too, i.e. the quantification in the similarity or difference between two typing samples, based purely on the relative latencies between every other keystroke (Bergadano et al., 2002), or between consecutive keystrokes instead (Lau et al., 2004). But there are other possibilities to decide which attributes must be considered to work with. They can be the mere duration, the latency, the digraph/trigraph latencies, the entropy, the edit distance and speed (for a list see in Revett et al., 2007).

Among all, here we decided to adopt a new attribute which, as far as we know, it was never investigated before.

The idea was born considering that it is clear that the typing style evolves with practice. A user who become more and more confident with the keyboard, will strike the same password more rapidly day by day. So, we intend here to consider not only the mostly adopted absolute values of *dwell* and *flight times*, but their *ratio*. In fact the user's typing style can evolve in rapidity, but the ratios of keystroke times can remain a constant.

The keystroke dynamics of the same person can be changing anyway due to many reasons, related to boundary conditions (a cold day may slow down the typist's fingers, a distraction for an irregularity occurs in the office environment, etc.), and/or to reasons strictly related to mood or to physical conditions (stress, fatigue, distraction, injury, etc.) but, again, the ratios between two (consecutive or not) dwell times, two flight times or one dwell and one flight time, can remain unchanged.

Sample Collection

Samples were collected using software developed under Max/MSP environment (Cycling74 Max/MSP, documentation available on the web at: http://cycling74.com/products/maxmspjitter/), with the aim of recording typing patterns with only few hours of work. Since each keystroke is captured solely by the key pressed, the press time, and the release time, the data to be recorded are of a small amount, so easily to store and, eventually, remotely transmitted with a low bandwidth requirement. The users had to utilize only a standard keyboard version with a QWERTY layout, and to type always assuming the same sitting posture.

In our work we use two different passwords, the word "*special*" and the number "*12057*". Each of the 16 volunteer was engaged for 10 days and each day he/she recorded 30+30 samples. So, we collect 300 trials per password per volunteer for a total of 9,600 trials per password. These trials were randomly split into 1,680 training, 360 testing, and 360 validation samples.

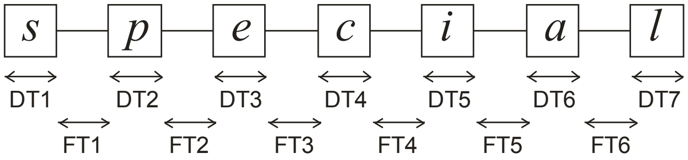
Three different typology of times were considered: *absolute, cumulative* and *ratio*. Let's details the differences considering, for instance, the word "*special*". As, schematized in Figure 1, there were 13 times associated to this word: DT1 (Dwell Time 1) was the time associated to how long the key "*s*" was held pressed; FT1 (Flight Time 1) was the elapsed time between the letter "*s*" and the letter "*p*"; DT2 was for the letter "*p*"; FT2 was for the time between "*p*" and "*e*"; DT3 was for the letter "*e*".. and so on till FT6 for the time between "*a*" and "*l*", and DT7 for the letter "*l*".

DT1, FT1, DT2, FT2, .. to DT7 are the *absolute times*, since they represent one time for one occurrence (key pressed or latency between keys). The *absolute times* are the commonly adopted times in literature to implement classifiers so to recognize the user (examples in Monrose and Rubin, 2000; Lau et al., 2004; Yu and Cho, 2004; Revett et al, 2006); Douhou and Magnus, 2009). A slight differentiation come from the *tri-graph latency* for which it was considered not just the dwell time (or *digraph latency*) but the elapsed time between the first key pressed and the third key pressed (Revett et al., 2007; Gingrich and Sentosa, 2008) or even the more sophisticated *n-graph*, but this approach was found to be not practical for a large database due to its scalability problem (Gunetti and Picardi, 2005).

But we associated to any key and "pause", can be associated a c*umulative time*, since we can consider for, let's say the key "*e*", not just the time DT3, but the time elapsed from the real beginning of the procedure, so DT3+FT2+DT2+FT1+DT1.

Finally we can base our classification on the *ratio* *times*, i.e. considering the values coming from the ratios: DT7/FL6, FT6/DT6, DT6/FT5, FT5/DT5, DT5/FT4, FT4/DT4, DT4/FT3, FT3/DT3, DT3/FT2, FT2/DT2, DT2/FT1, FT1/DT1.

Figure 1. The word "special" and the associated times



Multi-Class SVM Classification

A SVM identifies the optimal separating hyperplane (OSH) that maximizes the margin of separation between linearly separable points of two classes. The data points which lie closest to the OSH are called support vectors. It can be shown that the solution with maximum margin corresponds to the best generalization ability (Shawe-Taylor and Cristianini, 2000). Linearly non-separable data points in input space can be mapped into a higher dimensional (possibly infinite dimensional) feature space through a nonlinear mapping function, so that the images of data points become almost linearly separable. The discriminant function of a SVM has the following expression:

$$f(\mathbf{x})=\sum_{i}\alpha_{i}y_{i}K(\mathbf{x}_{i},\mathbf{x})+b \qquad (1)$$

where x_i is a support vector, $K(x_i,x)$ is the kernel function representing the inner product between x_i and x in feature space, coefficients α_i and b are obtained by solving a quadratic optimization problem in dual form (Shawe-Taylor & Cristianini, 2000). Usually, a soft-margin formulation is adopted where a certain amount of noise is tolerated in the training data. To this end, a user-defined constant C > 0 is introduced which controls the trade-off between the maximization of the margin and the minimization of classification errors on the training set (Shawe-Taylor & Cristianini, 2000).

SVMs were originally designed to work with dichotomies. A standard way to solve multi-class problems is to consider them as a collection of binary sub-problems, and then to combine their solutions.
In this context, the one-versus-all (OVA) approach has been used. The OVA method constructs *N* SVMs, N being the number of classes. The *i-th* SVM is trained using all the samples in the *i-th* class with a positive class label and all the remaining samples with a negative class label.
Our system uses 8 OVA SVM classifiers, one of each volunteer, whose input is represented by a feature vector based on *absolute, cumulative* and *ratio* values of *dwell* and *flight times*.
Our first classifications were based on the collected data of *dwell* and *flight absolute times*, exactly as they were and as literature reports. But for comparison, we performed the same classifications also on the basis of *cumulative* and *ratio times*, according to our purpose. Let's go into details, for example concerning the word "*special*".
The password authentication is given when the discriminant function of the corresponding SVM classifier is positive.
The SVMs were implemented using the software SVM light developed by Joachims (Joachims T., 1999). A linear kernel was used:

$$K\left(\mathbf{x}_{i},\mathbf{x}_{j}\right)=\mathbf{x}_{i}\cdot\mathbf{x}_{j} \qquad (2)$$

Linear SVMs need a regularization parameter *C* to be determined. To this end we looked for the best parameter values in a specific range using a grid-search on a validation set. More details will be given in the following Section.

EXPERIMENTAL RESULTS

In this section, we report on the simulation results of our password authentication system. The results are summarized by two statistics:

- False Acceptance Rate (FAR) — the percentage of an impostor that managed to login to the system

- False Rejection Rate (FRR) — the percentage of a valid user that is being denied an authentication

We have trained and validated the SVMs on the 1,680 samples of the training set and the 360 samples of the validation set, we have tested the system on the 360 samples of the test set. Besides, to compare the accuracy of our system with a system based on absolute values of dwell and flight times, other trials were performed on the same data set.
As an example, Figure 2 reports the absolute times vs. key numbers of 100 trials regarding subject 1. Figure 3 reports the cumulative times vs. key numbers of the same subject.

Figure 2. Absolute time vs. Key number for word “special”

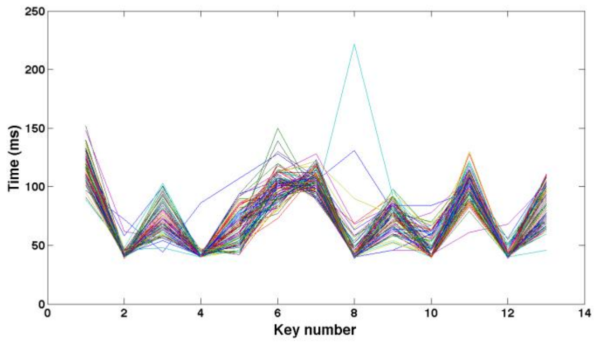
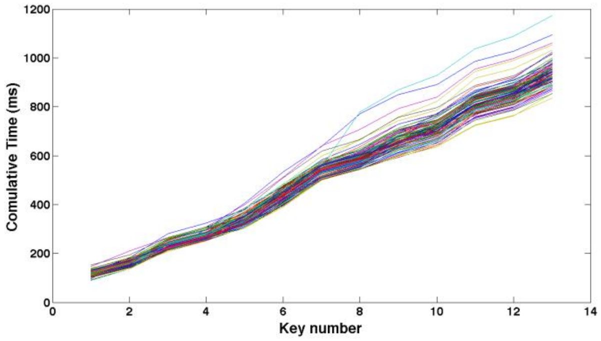


Figure 3. Cumulative time vs. Key number for word “special”



It shows not just the mere dwell and flight times but each time is referred to the start point (time zero). So, for instance, the for the letter “*p*” of the word “*special*” it is reported its dwell time, plus the flight time between the “*p*” and the “*s*”, plus the dwell time of the letter “*s*”. As it can be notice in this way the points of each trial lies on a sort of “straight line”, so denoting a linear behavior. In fact, we calculated the *coefficient of determination* R^2 for each of the trials, so discovering that it falls in the meaningful range of $0.8856 < R^2 < 0.9965$, with a standard deviation equal to “only” 0.011. It appears that the differences among trials merely relies on the slopes of these lines.
Figure 4 shows 100 trials again regarding the subject 1 but there are reported the *ratio values* (DT7/FL6, FT6/DT6, DT6/FT5, etc.) vs. key numbers. As it can be noticed, the ratio times furnish information too, since they are quite repeatable.

Figure 4. Time ratio vs. Key number for word “special”

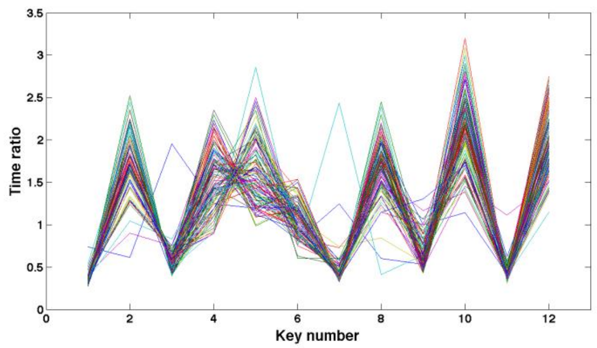


Table 1. Word “special” for informed users, partial dataset.

	Abs values FRR %	Cumulvalues FRR %	Ratio values FRR %	Abs values FAR %	Cumulvalues FAR %	Ratio values FAR %
User 1	6.87	5.79	0.55	2.98	2.68	1.16
User 2	3.23	0.87	0.84	0.98	0.90	0.93
User 3	7.99	7.41	7.49	1.53	1.39	1.34
User 4	0.13	0.14	0.00	1.06	0.43	0.38
User 5	3.89	3.49	3.55	1.88	1.61	1.56
User 6	1.90	1.87	0.34	2.55	2.31	0.99
User 7	6.43	5.51	5.69	0.97	0.78	0.77
User 8	0.15	0.09	0.01	0.78	0.43	0.12
Overall	**3.82**	**3.15**	**2.31**	**1.59**	**1.32**	**0.91**

The results concerning the word “*special*” and the code “*12572*” for the *informed users* are outlined in Tables 1 and 2. The results concerning the *partially informed users* are outlined in Tables 3 and 4. We intentionally decided to discharge the data for the first two days of tests, since we noticed how this could improve the classification results.
In fact, the experiment conducted using the complete dataset, including the data for the first two days, resulted with worse results.
Moreover, our results are comparable to those reported in the literature, demonstrating the consistency of our method.

Table 2. Code “12572” for informed users, partial dataset.

	Abs values FRR %	Cumul values FRR %	Ratio values FRR %	Abs values FAR %	Cumul values FAR %	Ratio values FAR %
User 1	4.38	3.95	0.86	2.88	2.13	0.97
User 2	2.34	1.47	1.40	0.86	0.74	0.83
User 3	7.45	6.35	6.64	2.24	1.88	1.84
User 4	0.31	0.28	0.00	1.64	0.80	0.69
User 5	1.10	0.90	0.80	1.63	1.01	0.81
User 6	3.87	3.23	1.12	2.59	2.28	1.44
User 7	5.93	4.31	4.44	0.99	0.85	0.81
User 8	0.32	0.18	0.08	1.19	0.76	0.54
Overall	**3.21**	**2.58**	**1.92**	**1.75**	**1.31**	**0.99**

Table 3. Word “special” for partially informed users, partial dataset.

	Abs values FRR %	Cumul values FRR %	Ratio values FRR %	Abs values FAR %	Cumul values FAR %	Ratio values FAR %
User 9	10.89	10.31	9.84	10.32	9.19	11.88
User 10	13.21	12.75	13.50	3.59	2.53	4.35
User 11	11.87	11.00	11.12	5.11	4.80	4.31
User 12	14.63	13.50	14.04	5.33	4.61	5.46
User 13	10.32	9.34	10.45	10.74	8.54	10.68
User 14	10.76	9.47	10.06	4.13	3.47	3.95
User 15	15.11	14.40	13.98	8.14	7.47	7.31
User 16	11.62	10.74	10.54	7.27	6.73	5.90
Overall	**12.30**	**11.43**	**11.69**	**6.82**	**5.92**	**6.73**

Table 4. Code “12572” for partially informed users, partial dataset.

	Abs values FRR %	Cumul values FRR %	Ratio values FRR %	Abs values FAR %	Cumul values FAR %	Ratio values FAR %
User 9	10.82	10.63	9.35	6.67	5.65	4.56
User 10	12.64	11.44	12.27	1.65	1.28	1.89
User 11	8.50	8.27	8.39	4.06	3.57	3.18
User 12	8.25	7.26	8.16	3.98	3.13	4.01
User 13	13.41	12.57	13.48	8.95	7.83	9.69
User 14	11.45	10.37	11.38	7.05	6.28	6.89
User 15	8.11	7.47	7.01	5.47	4.89	4.23
User 16	9.75	9.54	9.34	7.55	6.84	6.36
Overall	**10.36**	**9.69**	**9.92**	**5.67**	**4.93**	**5.10**

DISCUSSION AND CONCLUSIONS

The FRR% and FAR% values, obtained by our classification method, evidence that it is possible to improve the password security mechanism considering not just the dwell and flight times but their cumulative times and ratios. Figure 5 and 6 evidence this assertion. But this is true mainly when the user is aware that his/her keystrokes are recorded to implement the user’s authentication. In fact, in such a circumstance, the user tends to adopt a more repeatable, and then recognizable, typing way. On the contrary, an uninformed user tends naturally to do not replicate his/her typing habits in a way that the classification by means of ratio times can slightly worsen the identification.

Figure 5. FRR% and FAR% values vs. Informed / Partially informed users vs. Absolute, Cumulative, Ratio Times, for the word “special”

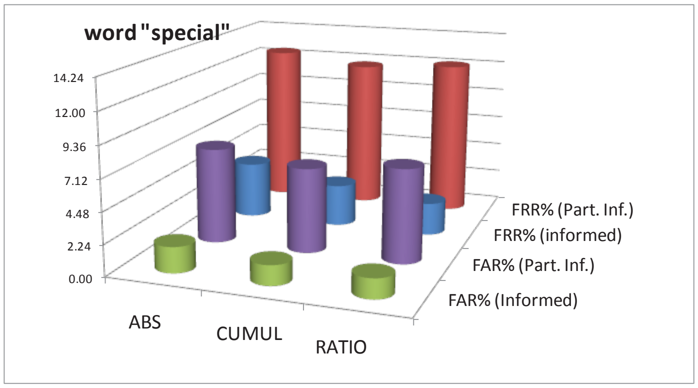
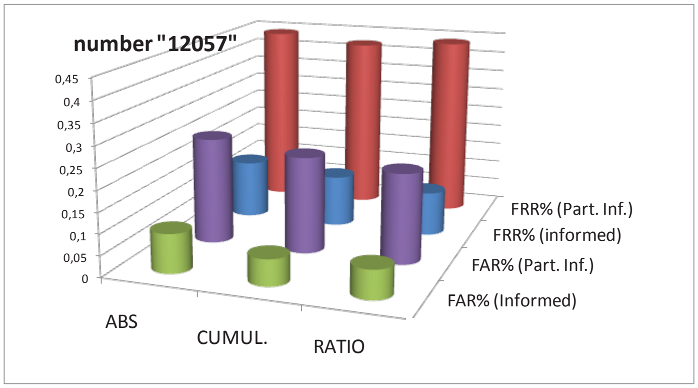


Figure 6. FAR% and FRR% values vs. Informed / Partially informed users vs. Absolute, Cumulative, Ratio Times, for the number “12057”



The proposed system was demonstrate to be likely to lead to tangible benefits in terms of improved performances and robustness of the password verification system if the requested boundary conditions are satisfied.
Another, interesting, conclusion coming from our work can be highlighted, i.e. the difference between a password made only by characters and a password made only by numbers. It resulted how the classifier furnisher slightly better results for the code “*special*” rather than the code “*12572*”. We believe this can be due to the fact that numbers are more closer one each other in a numeric keypad with respect the letters in a extended keyboard, so keypress and especially release times can be less repeatable because of minor differences.
Our future work will intend to investigate the

enhancement in password security when the user is asked to replicate the same password associating the action to a musical rhythm. Another important element to be considered could be the pressure applied on each key when typing, even if this will imply a bit of more sophistication in the hardware tool.

As regard the SVMs classifiers, a further improvement could be obtained using our proposed method with non linear kernel function, such as Radial Basis Function (RBF), Polynomial or Hyperbolic tangent

REFERENCES

1. McGuiness, Buchanon, Davenport, ML, BM, and Higgins, USA (2010). “Using Computers in Business Office”, J Computer and Information Tech, AcademyPublish.org, Vol 1, No 1, pp 78101.

2. Ahituv N, Lapid Y, Nuemann S, Verifying the authentication of an information system user. Comput. Secur. 6 2 (April 1987), pp. 152–157

3. Alexandre TJ, Biometrics on smartcards: an approach to keyboard behavioral signature, in: Second Smart Card Research and Advanced Applications Conference, 1996.

4. Bergadano F, Gunetti D, Picardi C, User Authentication through Keystroke Dynamics.” ACM Transactions on Information and System Security, Vol. 5, No. 4, Nov., p. 367-397, 2002

5. Bishop M., Klein D.V., Improving system security via proactive password checking, Computers & Security, 14 (1995) 233-249

6. Brown M, Rogers SJ, User identification via keystroke characteristics of typed names using neural networks, Int. J. Man-Mach. Stud. 39 (6) (1993) 999–1014.

7. Cho S, Han C, Han D, Kim H. Web-based keystroke dynamics identity verification using neural network, J Organ Comput Electron Commerce 2000;10(4):295e307.

8. de Oliveira, M., Kinto, V.S.E., Hernandez, E.D.M. and de Carvalho, T.C. (2005) User Authentication Based on Human Typing Patterns with Artificial Neural Networks and Support Vector Machines, SBC.

9. Douhou S, Magnus JR, The reliability of user authentication through keystroke dynamics, Statistica Neerlandica (2009) Vol. 63, nr. 4, pp. 432–449

10. Gaines, R. S., W. Lisowski, S. J. Press and N. Shapiro (1980), Authentication by keystroke timing: some preliminary results, Unpublished report, Rand Publication Series R-2526-NSF

11. Gingrich JHD, Sentosa A, A k-Nearest Neighbor Approach for User Authentication through Biometric Keystroke Dynamics, Proceedings of the IEEE Communications Society, 2008

12. Gunetti D, Picardi C, Keystroke Analysis of Free Text, ACM Transactions on Information and System Security, Vol. 8, No. 3, August 2005, Pages 312–347.

13. Hussien B, McLaren R, Bleha S, An application of fuzzy algorithms in a computer access security system, Pattern Recog. Lett. 9 (1989) 39–43.

14. Ke X, Manuel R, Wilkerson M, Jin L, Keystroke Dyamics: A Web-based Biometric Solution.” 13th USENIX Security Symposium, 2005

15. Lau E, Liu X, Xiao C, Yu X, Enhanced User Authentication Through Keystroke Biometrics,

Computer and Network Security, Final Project Report for the Massachusetts Institute of Technology, 2004

16. Leggett J, Williams G, Verifying identity via keystroke characteristics, Int. J. Man-Mach. Stud. 28 (1) (1988) 67–76.

17. Leggett J, Williams G, Usnick M, Longnecker M. Dynamic identity verification via keystroke characteristics. Int J Man Machine Stud 1991; 35:859e70.

18. Luis-Garcia R., Alberola-L'ope C., Aghzout O., Ruiz-Alzola J., Biometric identification systems, Signal Processing 83 (2003) 2539–2557

19. Joachims T., Making large-Scale SVM Learning Practical. Advances in Kernel Methods - Support Vector Learning, B. Schölkopf and C. Burges and A. Smola (ed.), MIT-Press, 1999.

20. Jobusch DL, Oldehoeft AE, A Survey of Password Mechanisms: Weaknesses and Potential Improvements. Part 1, Computers & Security, 8 (1989) 587-604

21. Miller GA, The magical number seven, plus or minus two: some limits on our capacity for processing information, Psychol. Rev., (1956).

22. Monrose F, Rubin AD, Keystroke dynamics as a biometric for authentication, Future Generation Computer Systems 16 (2000) 351–359

23. Obaidat, M. S. and B. Sadoun (1997), Verification of computer users using keystroke dynamics. IEEE Transactions on Systems, Man, and Cybernetics 27, 261–269

24. Ratha NK, Senior A, Bolle RM, Tutorial on automated biometrics, in: Proceedings of International Conference on Advances in Pattern Recognition, Rio de Janeiro, Brazil, March 2001

25. Revett K, Gorunescu F, Gorunescu M, Ene M, Magalhães PSS, Henrique D, Authenticating computer access based on keystroke dynamics using a probabilistic neural network, 2nd Annual International Conference on Global e-Security, Docklands, UK, 20 - 22 April 2006

26. Revett K, Gorunescu F, Gorunescu M, Ene M, Tenreiro de Magalhães S, Dinis Santos HM, A machine learning approach to keystroke dynamics based user authentication, Int. J. Electronic Security and Digital Forensics, Vol. 1, No. 1, 2007, pp. 55-70

27. Sang, Y., Shen, H. and Fan, P. (2004) Novel Imposters Detection in Keystroke Dynamics Using Support Vector Machines, LNCS Springer Berlin/Heidelberg, pp.666–669, ISSN 0302-9743.

28. Scholkopf B, Williamson RC, Smola AJ, Shawe-Taylor J, Platt JC. Support vector method for novelty detection. In: Solla SA, Leen TK, Mu¨ller K-R, editors. Advances in neural information processing systems, vol. 12. MIT Press; 2000. p. 582e8.

29. Shawe-Taylor J., Cristianini N., An Introduction to Support Vector Machines, Cambridge University Press, 2000.

30. Sung, K.S. and Cho, S. (2006) 'GA SVM wrapper ensemble for keystroke dynamics authentication', International Conference on Biometrics, Hong Kong, pp.654–660.

31. Yu E, Cho S, Keystroke dynamics identity verification - its problems and practical solutions, Computers & Security (2004)