Computer User Profiling Based on Keystroke Analysis

Tomasz Emanuel Wesołowski and Piotr Porwik

Abstract The article concerns the issues related to a computer user verification based on the analysis of a keyboard activity in a computer system. The research focuses on the analysis of a user's continuous work in a computer system, which constitutes a type of a free-text analysis. To ensure a high level of a users' data protection, an encryption of keystrokes was implemented. A new method of a computer user profiling based on encrypted keystrokes is introduced. Additionally, an attempt to an intrusion detection based on the *k*-NN classifier is performed.

Keywords Behavioral biometrics · Keystroke analysis · Free-text analysis · User verification

1 Introduction

The main task of the biometrics is the automatic recognition of individuals based on the knowledge of their physical or behavioral characteristics [1–4].

The behavioral biometrics methods use, among other things, an analysis of the movements of various manipulators (e.g., a computer mouse or a graphical tablet) [5] or the dynamics and statistics of typing on a computer keyboard [6–9]. The analysis of the way how a keyboard is used involves detection of a rhythm and habits of a computer user while typing on a keyboard [10]. The detection of these dependencies allows a recognition of a so-called user profile. This profile can then be used in the access authorization systems.

T.E. Wesołowski (✉) · P. Porwik
University of Silesia, Institute of Computer Science, Bedzinska 39,
41-200 Sosnowiec, Poland
e-mail: tomasz.wesolowski@us.edu.pl

P. Porwik
e-mail: piotr.porwik@us.edu.pl

R. Chaki et al. (eds.), *Advanced Computing and Systems for Security*,
Advances in Intelligent Systems and Computing 395,
DOI 10.1007/978-81-322-2650-5_1

In the proposed method, the data stored in a user profile contains information on the sequence of keys and on time dependencies that occur between the key events. The advantage of the method is that collecting and analyzing a user's activity data is performed in the background, which makes it practically transparent for a user.

A computer user's profile can be used in a host-based intrusion detection system (HIDS). The task of a HIDS is to analyze the logs containing registered user's activity and the appropriate response when it detects an unauthorized access. These systems may analyze the profiled activity in a real time. A recognition of a user based on the analysis of the users' typing habits, while using a keyboard can effectively prevent an unauthorized access when a keyboard is overtaken by an intruder [11, 12].

This paper proposes a new method for creating a profile of the computer system user. It is assumed that the user works with a QWERTY type keyboard with standard layout of the keys. The encryption of the alphanumeric data entered via a keyboard is performed in order to prevent an access to users' passwords.

2 Reference Data Sets

2.1 Issues Related to Data Sets

There are a number of issues related to the data sets used in research on computer user profiling based on keystroke analysis. First of all, there are no standards for data collection and benchmarking as it is in some other fields of biometrics. For this reason, it is difficult to compare the works of different researchers.

The testing data sets used in experiments have some limitations. In most cases, the data sets are private and not available for other researchers [13, 14]. The form of a registered text differs between different approaches. Some researchers use in their study short phrases [15] or passwords [10, 16] that are typed many times by the same user while other register long texts [17, 18]. In case of long text analysis, users are asked either to copy a given text [18] or to type freely a limited number of characters [15, 19] using a software designed for this purpose. In the second case, it is a so-called "free-text" registration. However, a free-text recorded this way does not represent the situation when a user is typing while working with a computer on his tasks on a daily basis.

In order to develop an on-line type user profiling and intrusion detection method that could be implemented in a HIDS working in real time, it is necessary to analyze a continuous work of a computer user in an uncontrolled environment. There is some serious security issues connected to user's continuous activity registration. The approach presented in [13] is based on continuous work analysis. However, the data typed by the user is registered as an open text. As users often type private information (e.g., passwords, PINs) this approach constitutes a serious threat to a security of computer system. Another issue related to a continuous activity analysis is a registration software. So-called "key-loggers" are considered as malicious

software. For this reason, a HIDS has to perform a data analysis on the fly (without storing the activity data) or if the activity data has to be stored encryption is necessary.

2.2 *Data Acquisition*

The biometric system presented in this paper is dedicated to and was tested in MS Windows operating systems. The identifiers of alphanumeric keys are encoded using the MD5 hash function. The same key identifier always receives the same code for the same user of the system.

The registration of user's activity data is performed automatically and continuously without involving a user. The data are captured on the fly and saved in the text files on the ongoing basis.

3 Data Analysis

The keyboard has been divided into groups of keys. The principle of the division is shown in Figs. 1 and 2.

The division of the keys is consistent with the following scheme for standard keyboard layout (Fig. 1):

- left function keys (with assigned identifiers *L*1–*L*14): *F*1…*F*7, *Esc, Tab, Caps lock, Left shift, Left ctrl, Windows, Left alt*;
- right function keys (with assigned identifiers *R*1–*R*25): *F*8…*F*12, *PrtScr, Scroll lock, Pause, Insert, Delete, Home, End, PgUp, PgDown, NumLck, Backspace, Enter, Right Shift, Right Ctrl, Context, Right alt, Arrows (up, down, left, right)*;
- alphanumeric keys (with assigned identifiers ID1–*ID*64);
- other keys.

In our approach, the tree structure includes 109 different groups G_{id}.

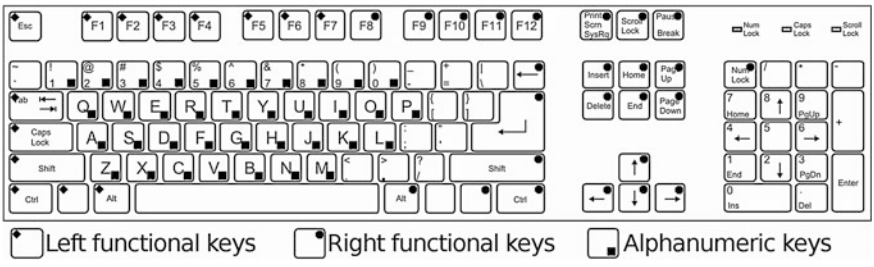

Fig. 1 QWERTY (105-keys) ISO standard keyboard with marked groups of keys

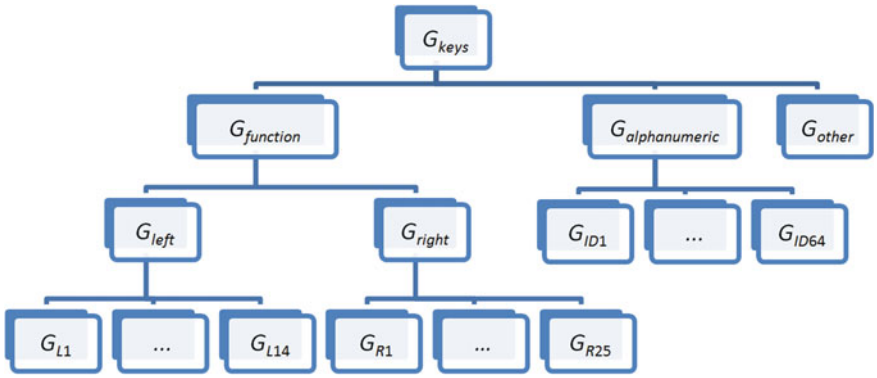

Fig. 2 Tree structure for organizing the groups of keys

Every use of the key is recorded in the next *i*th row of the text file as the following vector w_i:

$$w_i = [prefix, t_i, id] \tag{1}$$

where:

prefix—type of an event, *prefix* ∈ {'*K*', '*k*'}, key down → '*K*', key up → '*k*',
t_i—time of an event,
id—key identifier (e.g., *ID*1, *L*7, *R*15, etc.).

4 User Profiling

4.1 Time Dependencies Extraction

The first stage of feature extraction is to convert the text data file containing a set of vectors w_i, into a form of time dependencies between the keyboard events. The data file is searched for the rows with identical identifier *id*, then each pair of rows containing one key down event and following one key up event is converted into a vector v_{id} according to the following principle:

$$\begin{cases} w_i = ['K', t_i, id] \\ w_j = ['k', t_j, id] \end{cases} \rightarrow v_{id} = [t_i, t_j], i < j. \tag{2}$$

Vectors w_i of the same type (with the same identifier *id*) should be present in the data file even number of times. Otherwise, the vector, for which the pair was not found, will be considered as an artifact and will be removed.

Vectors v_{id} containing the element *id* are assigned to the group G_{id} in a leaf of the tree structure presented in Fig. 2. After enrollment, the same vector v_{id} is added

to all the groups higher in the branches of the tree structure until reaching the root group G_{keys}.

For example, if element *id* of a vector v_{id} is assigned an identifier $L1$ (it means that $id = L1$) than the mentioned vector v_{L1} will be added to the groups: G_{L1}, G_{left}, $G_{functional}$ and finally to G_{keys}.

4.2 Outliers Elimination

The user can use the keys of a keyboard freely, but the analysis of users activity is performed with some restrictions imposed on the key events. The next event cannot occur later than after the time t_{max} and the number of occurring consecutive events (that meet the first condition) cannot be less than c_{min}.

The values of parameters t_{max} and c_{min} have been determined experimentally.

For the exemplary keystroke sequence "*ABCDEF*" (Fig. 3) following times were recorded: $t_1 = 1.3$ s, $t_2 = 1.4$ s, $t_3 = 2.7$ s, $t_4 = 2.8$ s, $t_5 = 0.7$ s. Let $t_{max} = 2$ s and $c_{min} = 3$ be experimentally determined, than keystrokes '*A*', '*B*' and '*C*' will be considered to be correct, and the associated vectors w_i will be added to the data set, from which in the future user's profile will be established. Other events will be considered as outliers and discarded (keystroke '*D*' because $t_3 > t_{max}$ and keystrokes '*E*' and '*F*' because the number of elements in the group c_2 is lower than c_{min}).

4.3 Creating Feature Vectors

The next step of the activity data analysis is to create feature vectors. The groups G_{id} (Fig. 2) consist of vectors v_{id}. The total number of vectors that can be placed in the appropriate group is limited by g_{max}. The value of the $g_{\max}$ parameter has been determined experimentally and applies to all the groups of keystrokes in the tree structure.

When the number of vectors v_{id} assigned to the group G_{id} reaches its maximum value specified by g_{max}, then the feature vector is created. The group G_{id}, which recorded the number of vectors v_{id} equal to g_{max} is cleared and the process of adding further vectors v_{id} to the groups is resumed.

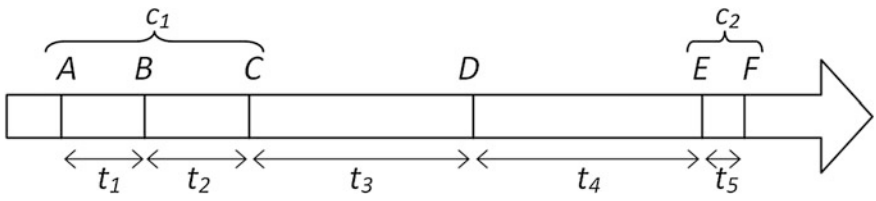

Fig. 3 The classification of keystrokes as outliers

The process ends when all the vectors v_{id} of a user have been processed or when the required number of feature vectors has been obtained.

4.4 Feature Vector

The feature vector is created based on the data contained in the groups G_{id} of the keystroke division structure (Fig. 2). For each group G_{id} standard deviation σ_{id} is calculated according to the formula (3).

Let the number of vectors $v_{id} = [t_i, t_j]$ registered in the group G_{id} be N. Then:

$$\sigma_{id} = \sqrt{\frac{1}{N}\sum_{k=1}^{N}(t_k - t_{id})^2} \tag{3}$$

where
t_k—dwell time of the k-th key belonging to the group G_{id} and $t_k = t_j - t_i$,
t_{id}—the average dwell time:

$$t_{id} = \frac{1}{N}\sum_{k=1}^{N} t_k \tag{4}$$

Finally, each feature vector consists of 109 standard deviation σ_{id} values (features). The above-described process is repeated and for a given user the next feature vector is created. The process is repeated as long as the required number of feature vectors has been obtained. The required number of feature vector was experimentally established. In our case, from the biometric point of view, the optimal number of feature vectors in a user's profile was equal to 1000.

5 The Results Obtained

The activity of four computer users has been registered within 1 month. In total, there are 123 files containing continuous activity data. The number of feature vectors generated for each user was

- *user*1—1470,
- *user*2—1655,
- *user*3—1798,
- *user*4—1057.

The feature vectors were normalized to the range of [0, 1]. Literature sources indicate a high efficiency of the k-NN classifiers [4, 6, 15, 16, 20, 21]. For this reason, intrusion detection was carried out by means of k-NN classifier.

Table 1 Parameter values used in the study

Parameter	Value
t_{max}	650 ms
c_{min}	5
g_{max}	15
k-NN	$k = 3$
Acceptance threshold	$\tau = 0.5$

The studies were verified using *leave-one-out* method and additionally repeated 20 times for different subsets of feature vectors of an intruder. The results obtained from the tests were averaged.

The testing procedure took into account the parameters described in Table 1. The values of parameters have been determined experimentally in order to obtain the lowest values of the EER.

5.1 Tuning the Parameters

In the first stage of the study, the experiments were performed to select the optimal values of the biometric system parameters. As an example, the results of experiments performed to obtain the optimal values of parameters k and t_{max} are presented. Figure 4 depicts the results of tuning the parameter k for the k-NN classifier.

Based on experiments the value of k was set to 3.

In Fig. 5, the results of experiments performed in order to obtain the optimal value of the t_{max} parameter are presented.

The best results were obtained for the value of the parameter t_{max} equal to 650 ms. For values below 400 ms, the outliers elimination process rejected too many keystrokes and there was not enough user's activity data left to create the necessary number of feature vectors for experiments.

5.2 The Final Results

The final results of the study are presented in Figs. 6 and 7. The charts in Figs. 6 and 7 should be interpreted as follows. The columns represent different pairs of computer owner (legitimate user) and intruder, for which the tests were performed. Each column indicates 20 intruder detection tests of the same intruder for different subsets of input data (dots). Squares represent the average score of 20 attempts for each pair of users. The dashed line represents the average value of the EER for the presented in this paper biometric system based on the analysis of the use of a computer keyboard.

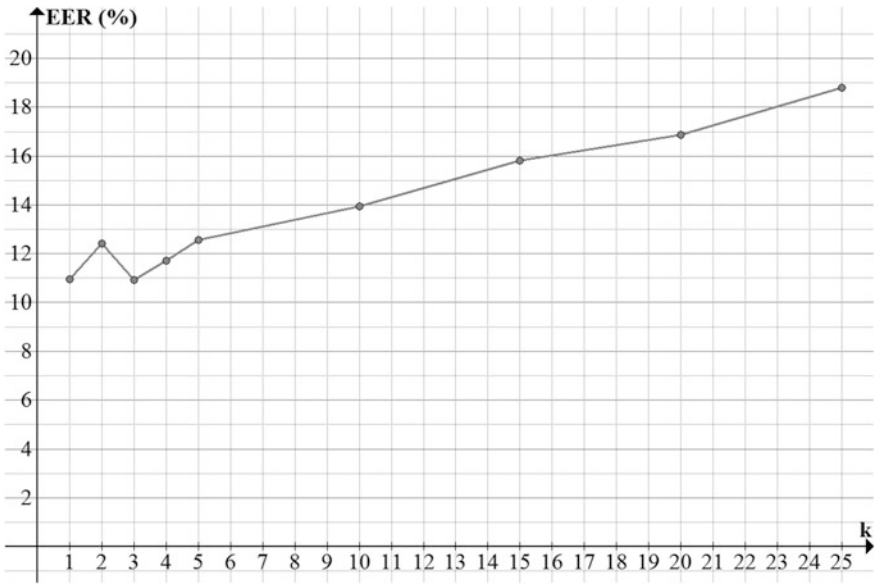

Fig. 4 The influence of parameter k on EER values of the method

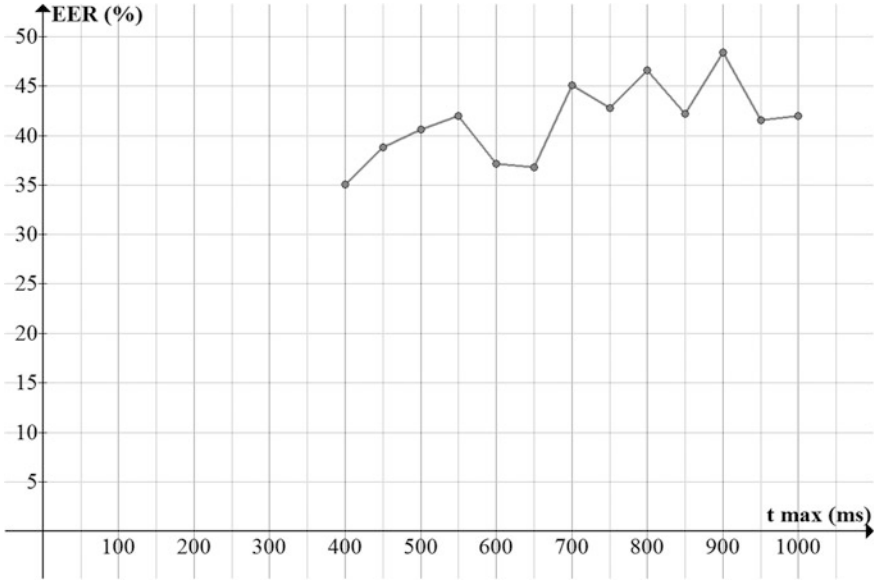

Fig. 5 The results of tuning the parameter t_{max}

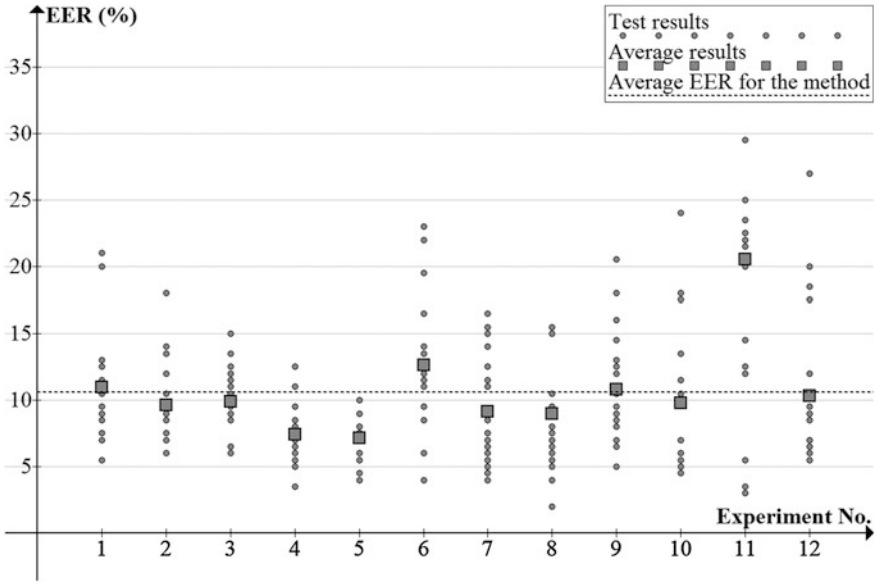

Fig. 6 Values of EER for the intrusion detection method and user's profile with outliers elimination

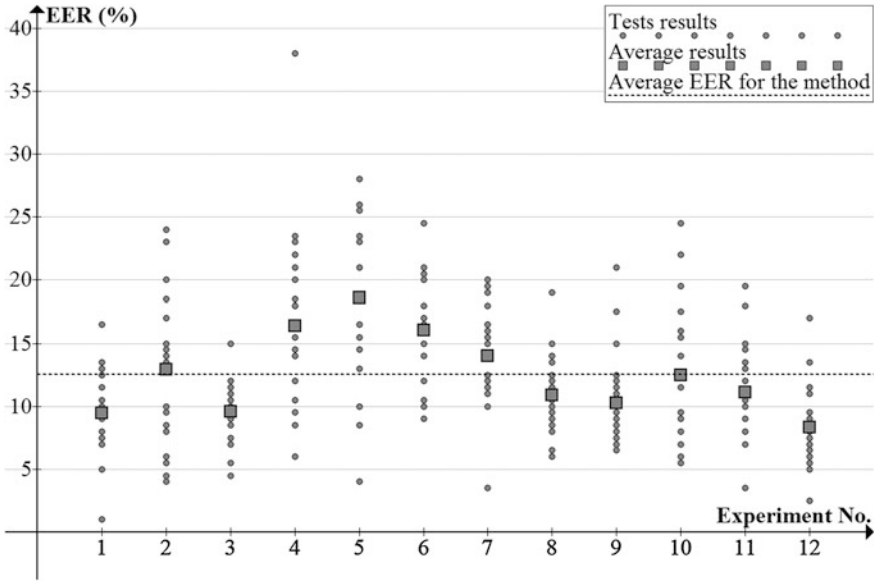

Fig. 7 Values of EER for the intrusion detection method and user's profile without outliers elimination

Figure 6 depicts the results of tests performed with the outliers elimination while in Fig. 7 the results of experiments conducted without eliminating the outliers are presented. The average value of the EER for all experiments was established at the level of 12.512 % without eliminating the outliers and at the level of 10.592 % with the outliers elimination.

6 Conclusions and Future Work

In this study, a series of experiments allowing an optimal selection of the parameters of the biometric system based on a use of a keyboard in order to determine the lowest value of the EER was performed. The achieved value of the EER equal to 10.59 % is better than the ones announced in [13, 16, 18, 19, 22].

The introduced method of a user's activity registration allows the analysis of a user's continuous work in an uncontrolled environment while performing everyday tasks. Additionally, a high security level was achieved by means of the MD5 hashing function.

Because the presented intrusion detection method uses a relatively high amount of data in order to create a user's profile and detect, the attack of a masquerader it is suitable for implementation in the off-line type intrusion detection systems.

In the future, the authors intend to explore the suitability of the other methods of data classification. As users often perform similar types of tasks during everyday activity future studies should consider the analysis of a user's activity in particular programs (e.g., text editors, web browsers). Additional research should be performed for users who work in the network environments where an intruder detection and localization is more difficult.

Acknowledgments The research described in this article has been partially supported from the funds of the project "DoktoRIS—Scholarship program for innovative Silesia" co-financed by the European Union under the European Social Fund.

References

1. Kudłacik, P., Porwik, P.: A new approach to signature recognition using the fuzzy method. Pattern Anal. Appl. **17**(3), 451–463 (2014). doi:10.1007/s10044-012-0283-9
2. Kudłacik, P., Porwik, P., Wesołowski, T.: Fuzzy Approach for Intrusion Detection Based on User's Commands. Soft Computing, Springer, Berlin Heidelberg (2015), doi:10.1007/s00500-015-1669-6
3. Pałys, M., Doroz, R., Porwik, P.: On-line signature recognition based on an analysis of dynamic feature. In: IEEE International Conference on Biometrics and Kansei Engineering, pp. 103–107, Tokyo Metropolitan University Akihabara (2013)
4. Porwik, P., Doroz, R., Orczyk, T.: The k-NN classifier and self-adaptive Hotelling data reduction technique in handwritten signatures recognition. Pattern Analysis and Applications, doi:10.1007/s10044-014-0419-1

5. Wesołowski, T., Pałys, M., Kudłacik, P.: computer user verification based on mouse activity analysis. Stud. Comput. Intell. **598**, 61–70 (2015). Springer International Publishing
6. Alsultan, A., Warwick, K.: Keystroke dynamics authentication: a survey of free-text methods. J. Comput. Sci. Issues **10**(1) 1–10 (2013) (Issue 4)
7. Araujo, L.C.F., Sucupira Jr., L.H.R., Lizarraga, M.G., Ling, L.L., Yabu-Uti, J.B.T.: User authentication through typing biometrics features. IEEE Trans. Signal Process. **53**(2) 851–855 (2005)
8. Banerjee, S.P., Woodard, D.L.: Biometric authentication and identification using keystroke dynamics: a survey. J. Pattern Recognit. Res. **7**, 116–139 (2012)
9. Teh, P.S., Teoh, A.B.J., Yue, S.: A survey of keystroke dynamics biometrics. Sci. World J. **2013**, Article ID: 408280, 24 pp. (2013) doi:10.1155/2013/408280
10. Zhong, Y., Deng, Y., Jain, A.K.: Keystroke dynamics for user authentication. In: IEEE Computer Society Conference, Computer Vision and Pattern Recognition Workshops, pp. 117–123 (2012), doi:10.1109/CVPRW.2012.6239225
11. Raiyn, J.: A survey of cyber attack detection strategies. Int. J. Secur. Its Appl. **8**(1), 247–256 (2014)
12. Salem, M.B., Hershkop, S., Stolfo, S.J.: A survey of insider attack detection research. Adv. Inf. Secur. **39**, 69–90, Springer US (2008)
13. Dowland, P.S., Singh, H., Furnell, S.M.: A preliminary investigation of user authentication using continuous keystroke analysis. In: The 8th Annual Working Conference on Information Security Management and Small Systems Security (2001)
14. Saha, J., Chaki, R.: An Approach to Classify Keystroke Patterns for Remote User Authentication. J. Med. Inf. Technol. **23**, 141–148 (2014)
15. Lopatka, M., Peetz, M.: Vibration sensitive keystroke analysis. In: Proceedings of the 18th Annual Belgian-Dutch Conference on Machine Learning, pp. 75–80 (2009)
16. Killourhy, K.S., Maxion, R.A.: Comparing anomaly-detection algorithms for keystroke dynamics. In: International Conference on Dependable Systems and Networks (DSN-09), pp. 125–134. IEEE Computer Society Press (2009)
17. Rybnik, M., Tabedzki, M., Adamski, M., Saeed, K.: An exploration of keystroke dynamics authentication using non-fixed text of various length, In: IEEE International Conference on Biometrics and Kansei Engineering, pp. 245–250 (2013)
18. Tappert, C.C., Villiani, M., Cha, S.: Keystroke biometric identification and authentication on long-text input. In: Wang, L., Geng, X. (eds.) Behavioral Biometrics for Human Identification: Intelligent Applications, pp. 342–367 (2010), doi:10.4018/978-1-60566-725-6.ch016
19. Gunetti, D., Picardi, C., Ruffo, G.: Keystroke analysis of different languages: a case study. Adv. Intell. Data Anal. **3646**, 133–144 (2005)
20. Foster, K.R., Koprowski, R., Skufca, J.D.: Machine learning, medical diagnosis, and biomedical engineering research—commentary. Biomed. Eng. Online **13**, 94 (2014), doi:10.1186/1475-925X-13-94
21. Hu, J., Gingrich, D., Sentosa, A.: A K-nearest Neighbor Approach for User Authentication through Biometric Keystroke Dynamics. In: IEEE International Conference on Communications, pp. 1556–1560 (2008)
22. Filho, J.R.M., Freire, E.O.: On the equalization of keystroke timing histogram. Pattern Recognit. Lett. **27**(13), 1440–1446 (2006)