keystroke analysis, free text analysis
user profile, intruder detection,
user verification

Tomasz Emanuel WESOŁOWSKI[1], Piotr PORWIK[1]

USER VERIFICATION BASED ON THE ANALYSIS OF KEYSTROKES WHILE USING VARIOUS SOFTWARE

The article presents the new approach to a computer users verification. The research concerns an analysis of user's continuous activity related to a keyboard used while working with various software. This type of analysis constitutes a type of free-text analysis. The presented method is based on the analysis of users activity while working with particular computer software (e.g. text editors, utilities). A method of computer user profiling is proposed and an attempt to intrusion detection based on k-NN classifier is performed. The obtained results show that the introduced method can be used in the intrusion detection and monitoring systems. Such systems are especially needed in medical facilities where sensitive data are processed.

1. INTRODUCTION

The ubiquity of computers and mobile devices increases the risk of unauthorized access to our data. In most cases access to computer devices is protected by simple methods. The most popular methods for computer user authentication and verification usually require elements such as passwords or tokens. These elements are very vulnerable to loss or theft. The alternative is to use biometric methods as they use the characteristics of the person being verified related to its appearance or behavior. The forgery of biometric characteristics is not impossible but much more difficult. The automatic recognition of individuals by means of biometrics can be based on the knowledge of their behavioral (e.g., computer user activity [6], [16], signature characteristics [5], [8], [9]) or physiological characteristics. Behavioral biometrics is related to the pattern of behavior of a person. Biometric methods, used in computer science for computer user verification, can be based on the analysis of a user's activity connected to different manipulators (e.g., computer mouse [14]) or a keyboard [1], [2], [13]. 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 [17]. Such detected characteristics allow to build a so called user's profile that can be used in the access authorization and verification systems.

Another issue is a onetime authorization made usually when starting a work with a device or an IT system. This type of user authentication can result in a serious threat especially in open type areas such as medical facilities, causing systems and data being vulnerable to an intruder

[1] University of Silesia, Institute of Computer Science, ul. Bedzinska 39, 41-200 Sosnowiec, Poland
e-mail: {tomasz.wesolowski, piotr.porwik}@us.edu.pl

attack. To ensure a high level of security, especially of medical data, information systems should be continuously monitored. Such a monitoring is performed by the Intrusion Detection Systems (IDS) that constantly monitor all operations performed by users, and then try to verify the user's identity.

The proposed approach concerns the field of biometrics, specifically computer user verification and intrusion detection based on user profiling. The profile of a user is built from the information on time dependencies that occur between the key events. The user's profile can be used in a Host-based Intrusion Detection System (HIDS) which analyzes (preferably in real-time) the logs with registered activity of the user responds when an unauthorized access is detected. A verification of a user based on the analysis of his typing habits while using a keyboard can effectively prevent an unauthorized access when a keyboard is overtaken by an intruder (so called masquerader) [10], [11]. Such a situation can easily happen for example in hospitals when the staff is busy with an emergency situation.

In the presented method the process of collecting a user's activity data is performed in the background, practically not involving a user. The innovation is, that the activity data is analyzed not as a whole but is divided into the activity related to the particular software used by the user. This allows to analyze the habits of a user while working with a particular software. Each medical facility normally uses a single medical IT System so it is reasonable to analyze the activity of a user connected to his work with a particular software.

2. RELATED WORKS

There is a number of papers related to a computer user profiling, authorization and verification based on the keystroke dynamics. Unfortunately there is also a number of issues related to the data sets used in research [15] which make it very difficult or even impossible to compare the works of different researchers.

An interesting approach to computer user authentication using dynamics of typing is described in [12]. Users had to type any text consisting of about 650 characters. Characters entered by the users were stored as a plain text, without any encryption. Next, the characters were divided into groups that were organized in a hierarchical structure. For the purpose of analysis the assumptions typical for English language were used. For this reason the authentication method was intended only for texts written in English. The authentication performance under non-ideal conditions was 87,4% in average and it decreased to 83,3% over two-week intervals. This method processes the plain text, which is a serious threat. Furthermore, it is limited to texts written in a determined language - English in this case.

Methods described in [15], [16] are based on the idea presented in [12] but they are free of some of its limitations - they are language independent and they are based on free text typed by the users during every day activities. The approach presented in this paper extends the idea introduced in [15], [16] by dividing the activity according to the software used by the user and performing the analysis separate for each used software.

3. USER'S ACTIVITY DATA

For the purpose of the research the dedicated data acquisition software was implemented. The software was designed to collect and save any event generated while using a keyboard and/or computer mouse. It is designed for MS Windows operating systems and does not require any additional libraries. Its purpose is to work continuously, recording the activity associated with a keyboard, mouse, and the use of popular programs. The registration of user's activity data is performed automatically and continuously without involving a user. The data

Table 1. Event prefixes and data description for a recorded user's activity.

Prefix	Event	Event Data
K	key down	an encrypted code of an alphanumeric key or a code of a function key
k	key up	
W	window change	encrypted name of a window and unencrypted name of a recognized software

are captured on the fly and saved in the text files on the ongoing basis (Fig. 1). To ensure user's private data protection the identifiers of alphanumeric keys are encoded using the MD5 hash function. The first line of the data file contains the screen resolution. The next lines contain the sequence of events related to user's activity. Each line starts with the prefix followed by the time of the event and additional information (Table 1).

```
RES,X1280,Y800
W,1396363173585,.Word.79de0f4d07dff692df05aaf526303b64
M,1396363173647,711,415
M,1396363173663,701,421
K,1396968151226,#e063bab5176c605c09cd53dbade73eb9
k,1396968151376,#e063bab5176c605c09cd53dbade73eb9
K,1396968152306,#1f5070fa63f3dbcd3ac74e86d080e0a5
k,1396968152446,#1f5070fa63f3dbcd3ac74e86d080e0a5
K,1396968153376,#e063bab5176c605c09cd53dbade73eb9
k,1396968153576,#e063bab5176c605c09cd53dbade73eb9
```

Fig. 1. An example of activity data.

4. DATA ANALYSIS

In the studies only the events related to window changes and a use of a keyboard are taken into consideration. For the purpose of the research a keyboard has been divided into groups of keys. It was assumed that for a standard QWERTY keyboard layout the principle of keys division is consistent with the following scheme:

- left function keys (with assigned identifiers $L1$ - $L14$): $F1..F7$, *Esc, Tab, Caps lock, Left shift, Left ctrl, Windows, Left alt*;
- right function keys (with assigned identifiers $R1$ - $R25$): $F8..F12$, *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 $ID1$ - $ID64$);
- other keys.

4.1. DATA PREPROCESSING

The first stage of data analysis is data preprocessing in order to extract the time dependencies of keystrokes generated while user was working with various software. Each window change or keyboard event (consisting of pressing or releasing the key) is stored in the subsequent i-th row of the input data file as a following vector w_i:

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

where:

$prefix$ - type of an event, $prefix \in \{'W',' K',' k'\}$ (Table 1),
t_i - timestamp of an event,

id - key identifier (e.g. $L1$, $L10$, $R25$, etc.) or encrypted window identifier with an unencrypted software name (if recognized, e.g., Chrome, Word).

In preprocessing stage vectors w_i are filtered and only keystrokes made while working with one of the recognized software are taken for further analysis - all the other keystrokes are omitted. At the moment the activity registration software recognizes the following labels: ”Word”, ”Excel”, ”Chrome”, ”Firefox”, ”Internet Explorer”, ”Matlab”, ”Notatnik”, ”Opera”, ”Outlook”, ”Thunderbird”. However, it can be configured to recognize also other labels in the future.

If one of the labels is found in the name of the window saved within the window change event data (in the id element of a vector w_i with $prefix='W'$) than all the consecutive keystrokes (vectors w_i with $prefix \in \{'K','k'\}$) until the next window change event are included into the analysis. If the name of the window does not include any of the defined labels than all the consecutive keystrokes until the next window change event are excluded from further analysis. The principle of data preprocessing is presented in Figure 2.

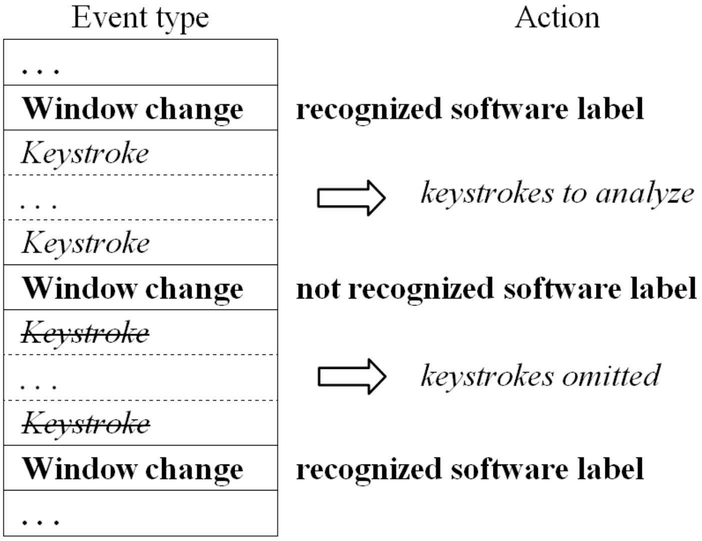

Fig. 2. The principle of data preprocessing.

Not all the users were working with all of the recognized computer programs. But most of them used a web browser (Chrome, Firefox or Internet Explorer) and/or Word text editor.

4.2. OUTLIERS ELIMINATION

The data analysis is performed with some restrictions imposed on the key events in order to eliminate the outliers. A user can use the keys of a keyboard freely, but in the data analysis process it was assumed that the keystrokes form sequences of events. The rules for outliers elimination are as follows:

1) a next event cannot occur later than after the time t_{max} and
2) the number of occurring consecutive events (that meet the first condition) cannot be less than c_{min}.

This means, that the event is added to the sequence only if the time that has elapsed since the previous event does not exceed the maximum allowed time t_{max} between two events. A sequence, in which number of elements meeting the first condition does not reach the minimum number of elements c_{min} is omitted in further analysis. The values of parameters t_{max} and c_{min} have been determined experimentally.

4.3. TIME DEPENDENCIES EXTRACTION

Next, time dependencies between the keyboard events are extracted. A set of vectors w_i extracted in the preprocessing stage from the input text data is converted into a set of vectors v_{id} representing time dependencies between the keyboard events. The extracted set of vectors w_i consists now only of vectors representing keystroke events. The consecutive pairs of rows (vectors w_i) containing an identical identifier id are taken from the data set, and 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 formula:

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

In the studies the time dependencies are represented by dwell times for individual keys and the time between keystrokes for pairs of keys. Vectors w_i of the same type (with the same identifier id) should be present in the data file an even number of times. Otherwise, the vector, for which the pair was not found, will be considered as an artifact and will be removed.

5. USER PROFILING

In the first stage of user profiling groups of keystrokes are created. The groups are organized into two tree structures T_{keys} and T_{pairs} (Fig. 3). For single keys each vector v_{id} containing the timestamps of a pair of keystrokes with the identifier id is assigned to the group G_{id} in a leaf of one of the tree structures presented in Figure 3. After enrollment, the same vector v_{id} is added to all the groups higher in the hierarchy of the particular branch until reaching the root group G_{keys}. For example, if element id of a vector 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}, $G_{function}$ and finally to G_{keys}. By analogy, vectors v_{id} representing pairs of keys are added to the groups G_{id} in the second tree structure T_{pairs}. The total number of different groups G_{id} in both tree structures is 113.

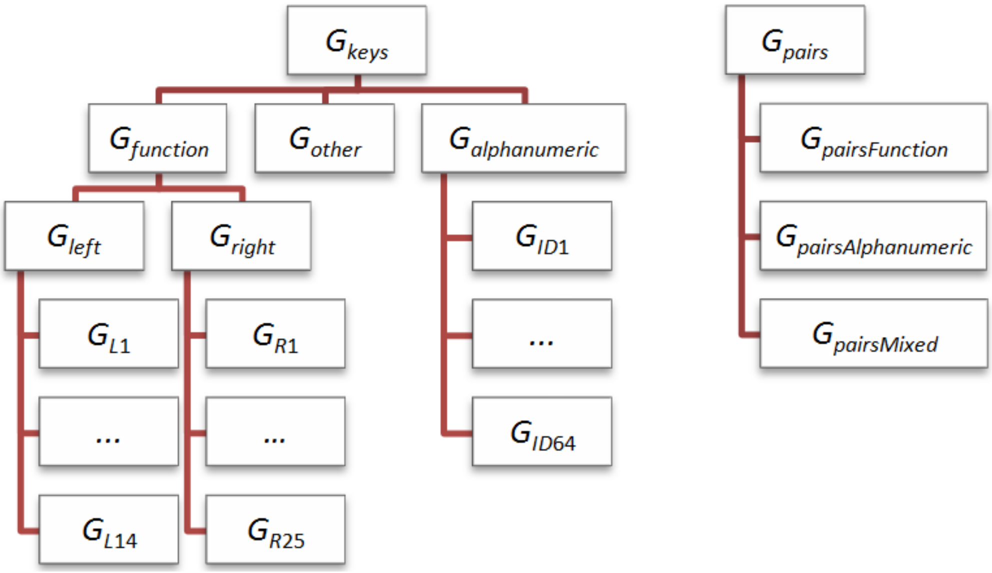

Fig. 3. Tree structures for organizing the groups of keys and key pairs.

The next step of user profiling is to create feature vectors. Each group G_{id} organized into the tree structures T_{keys} and T_{pairs} stores an information on a number of vectors v_{id} added to this group. A maximum number of vectors that can be placed in a single group is limited by the parameter g_{max} that has been determined experimentally and is the same for all of the groups. When, in any of the groups G_{id}, the number of vectors v_{id} assigned to this group reaches the specified maximum value g_{max} the feature vector is created and this group is cleared. The process is resumed and further vectors v_{id} are being added to the groups - further feature vectors are being created.

The feature vector is based on the data from all the groups G_{id} of the structures T_{keys} and T_{pairs}. Separately for each of the 113 groups G_{id} separately the standard deviation σ_{id} (3) is calculated according to the following formula:

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

where:

N_{id} - the number of vectors $v_{id} = [t_i, t_j]$ registered in the group G_{id},

t_k - dwell time of the k-th key belonging to the group G_{id} in T_{keys} or time between keystrokes for pairs of keys belonging to the group G_{id} in T_{pairs},

t_{id} - the average of values in group G_{id}:

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

Finally, each feature vector consists of 114 features (113 standard deviation σ_{id} values and the identifier of the software used when the activity was recorded). The process is repeated until the required number of feature vectors has been obtained or until all the vectors v_{id} have been processed. A subset of the feature vectors of the given user constitutes its profile.

6. THE RESULTS OF MASQUARADER DETECTION

The activity of ten computer users has been registered within one month. Four of them shared the same computer system, so in the collected data the corresponding keys have the same id for all the users. Six of the users used separate hosts so the same key has a different encoded id for different user. The goal of this study was to detect masqueraders - intruders that use the opportunity to gain access to the system when the already authorized legitimate user is temporarily not present at the working place. For this reason only the data of the users who share the host could be analyzed. 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 feature vectors were normalized to the range of [0,1]. Literature sources indicate a high efficiency of the k-NN classifiers [1], [3], [4], [7], [9], [12]. For this reason intrusion detection was carried out by means of k-NN classifier.

In the first stage of the study the experiments were performed to select the optimal values of the biometric system parameters. The values of parameters have been determined experimentally in order to obtain the lowest values of the EER (Table 2).

For analyzing the general activity of the user the profile consisted of 1000 feature vectors. In case of this study, when only the keystrokes made while working with recognized software were taken into consideration and parts of the data were omitted, there was not enough vectors generated and users' profiles were built based on all the generated feature vectors.

Table 2. Values of biometric system parameters used in the study.

Parameter	Value
t_{max}	650 ms
c_{min}	5
g_{max}	15
k-NN	$k = 3$

The tests were performed by using 100 samples of data - 50 representing the legitimate user (to whom the analyzed profile belonged) and another 50 representing an intruder. Figure 4 depicts an example of the distances obtained when analyzing the behavior of the users while working with a text editor. The first 50 samples represent data subsets of the legitimate user and samples no. 51-100 represent a masquerader.

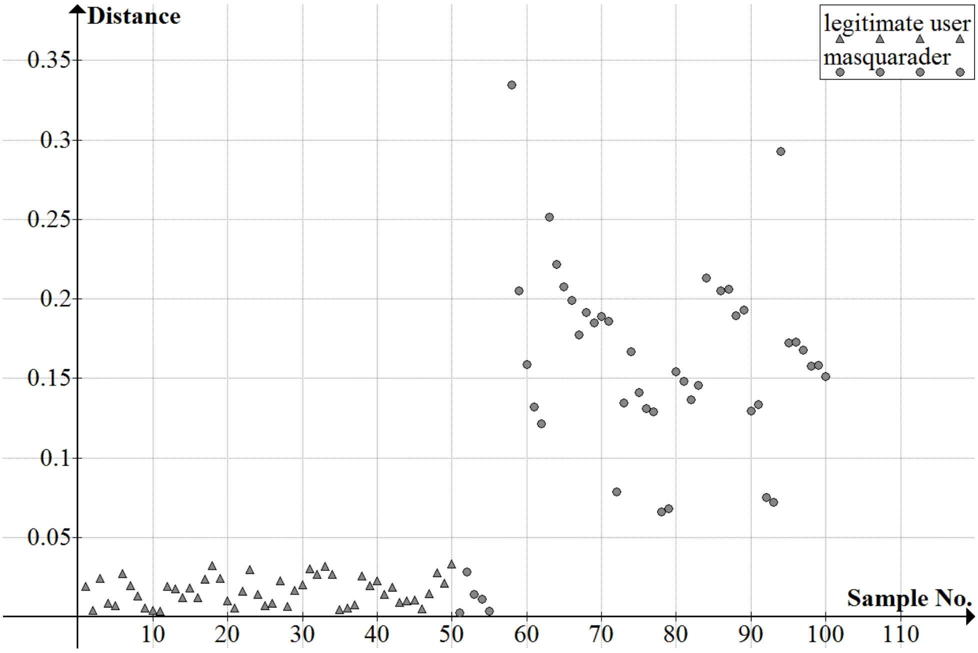

Fig. 4. Example of distances when analyzing the activity of users while working with text editor only.

For the comparison Figure 5 presents an example of the distances obtained when analyzing the general behavior of users without taking into account which software is used. The first 100 samples represent data subsets of the legitimate user and samples no. 101-200 represent a masquerader. It can be clearly noticed that the distribution of the masquerader's and the legitimate user's samples differs in both cases. When analyzing the activity of users working with text editor the activity of the intruder has clearly different characteristics than in case of the general activity analysis.

The FAR and FRR curves for the method based on the analysis of users' activity have been designated (Fig. 5). The optimal performance of the biometric system analyzing the activity of the computer user while working with particular software was achieved for values of the parameters presented in Table 2 and for the acceptance threshold $\tau = 0,0355$. The average value of the EER for the studies was established at the level of 7%.

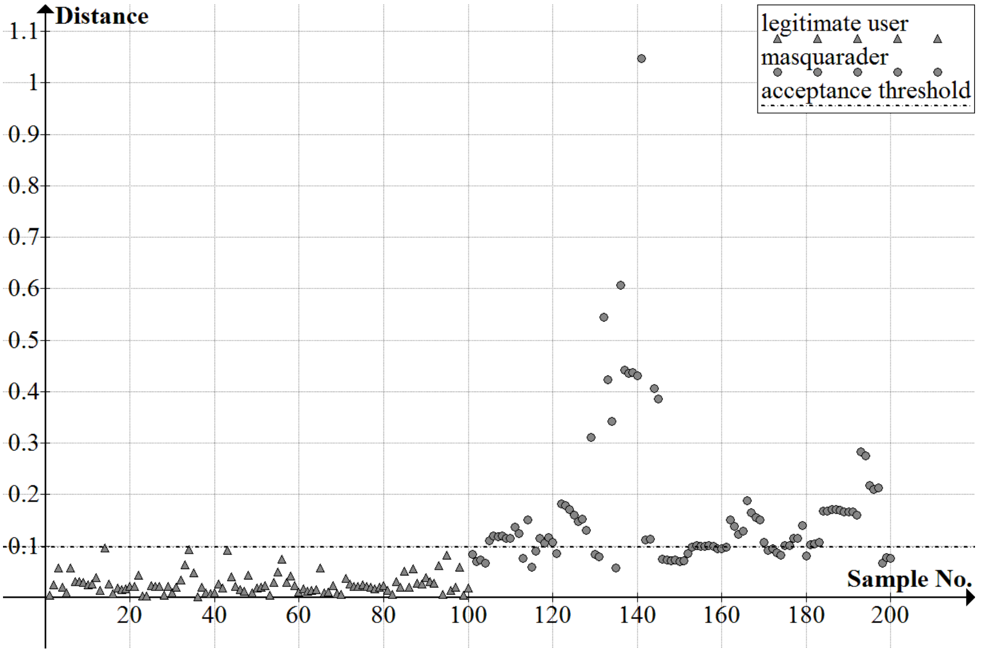

Fig. 5. Example of distances when analyzing the general users activity.

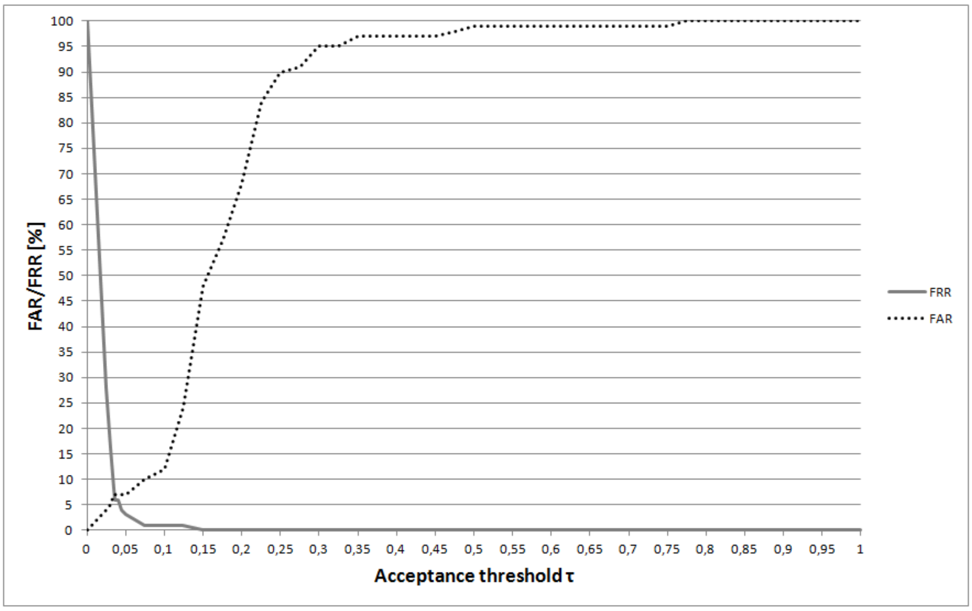

Fig. 6. The dependence of FAR and FRR on acceptance threshold.

7. CONCLUSIONS

The aim of the study was to analyze the activity of computer users working in a selected environment, using a particular software, for example a medical system used in a hospital or other medical facility. This approach eliminates differences when recording a user's activity connected to various additional tasks performed only occasionally by users. However, when focusing on a single software, due to the limited size of a data set, it was more difficult to perform the analysis as in a case of a general activity analysis. The number of samples used in experiments was reduced and the number of analyzed feature vectors was equal to 100 (50 vectors of legitimate user and 50 of a simulated intruder). During the experiments it has been observed that the characteristics of a user's activity differs when working with various software.

This means that the keyboard was used in a different way when working with text editors, web browsers or calculation sheets. Finally, because not all of the users had been working with the same set of software for the experiments the text editor was chosen due to the highest number of available feature vectors.

The new approach introduced in this paper allowed to reduce the EER of the intrusion detection method based on the presented computer user's profile. The in this study achieved value of EER = 7% is better than the one of the methods based on the analysis of the general activity of a user announced in [12], [15], [16]. Additionally the proposed method of recording user's activity data introduces a high level of security through the use of MD5 encoding function. This allows the analysis of user's continuous work in real conditions. It is an innovative solution, however, it causes that the comparison with other methods is difficult because most methods are based on an open text and limited length, fixed text analysis.

In future, the authors intend to explore the suitability of other methods of data classification for intruder detection. Additional research should be performed for users who work in the network environments where intruders detection and localization is more difficult. The future studies should also consider merging a user's activity data connected to the use of computer mouse and keyboard.

ACKNOWLEDGEMENT

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.

BIBLIOGRAPHY

[1] ALSULTAN A., WARWICK K. Keystroke dynamics authentication: A survey of free-text methods. IJCSI International Journal of Computer Science Issues, July 2013, Vol. 10. pp. 1–10.

[2] BANERJEE S., WOODARD D. Biometric authentication and identification using keystroke dynamics: A survey. Journal of Pattern Recognition Research, 2012, Vol. 7. pp. 116–139.

[3] HU J., GINGRICH D., SENTOSA A. A k-Nearest Neighbor approach for user authentication through biometric keystroke dynamics. IEEE International Conference on Communications, 2008. pp. 1556–1560.

[4] KILLOURHY K., MAXION R. Comparing anomaly-detection algorithms for keystroke dynamics. International Conference on Dependable Systems & Networks (DSN-09), 2009. IEEE Computer Society Press, pp. 125–134.

[5] KUDŁACIK P., PORWIK P. A new approach to signature recognition using the fuzzy method. Pattern Analysis & Applications, 2014, Vol. 17. pp. 451–463.

[6] KUDŁACIK P., PORWIK P., WESOŁOWSKI T. Fuzzy approach for intrusion detection based on user's commands. Soft Computing, 2015. Springer-Verlag Berlin Heidelberg.

[7] LOPATKA M., PEETZ M. Vibration sensitive keystroke analysis. Proceedings of The 18th Annual Belgian-Dutch Conference on Machine Learning, 2009. pp. 75–80.

[8] PALYS M., DOROZ R., PORWIK P. On-line signature recognition based on an analysis of dynamic feature. IEEE International Conference on Biometrics and Kansei Engineering, 2013. Tokyo Metropolitan University Akihabara, pp. 103–107.

[9] 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, 2014, Vol. 17.

[10] RAIYN J. A survey of cyber attack detection strategies. International Journal of Security and Its Applications, 2014, Vol. 8. pp. 247–256.

[11] SALEM M., HERSHKOP S., STOLFO S. A survey of insider attack detection research. Advances in Information Security, 2008, Vol. 39. Springer US, pp. 69–90.

[12] TAPPERT C., VILLIANI M., CHA S. Keystroke biometric identification and authentication on long-text input. Behavioral Biometrics for Human Identification: Intelligent Applications, 2010. pp. 342–367.

[13] TEH P. S., TEOH A. B. J., YUE S. A survey of keystroke dynamics biometrics. The Scientific World Journal, 2013, Vol. 2013. p. 24 pages.

[14] WESOŁOWSKI T., PALYS M., KUDŁACIK P. Computer user verification based on mouse activity analysis. New Trends in Intelligent Information and Database Systems, 2015, Vol. 598 of Studies in Computational Intelligence. Springer International Publishing, pp. 61–70.

[15] WESOŁOWSKI T. E., PORWIK P. Computer user profiling based on keystroke analysis. Advanced Computing and Systems for Security, 2015, Vol. 395 of Advances in Intelligent Systems and Computing. Springer.

[16] WESOŁOWSKI T. E., PORWIK P. Keystroke data classification for computer user profiling and verification. Computational Collective Intelligence, 2015, Vol. 9330 of Lecture Notes in Computer Science. Springer International Publishing, pp. 588–597.

[17] ZHONG Y., DENG Y., JAIN A. Keystroke dynamics for user authentication. Computer Vision and Pattern Recognition Workshops, IEEE Computer Society Conference, 2012. pp. 117–123.