Hidden Authentication of the User Based on Neural Network Analysis of the Dynamic Profile

Anastasiya Sivova, Alexey Vulfin, Konstantin Mironov and Anastasiya Kirillova
Department of Computer Engineering and Information Security Ufa State Aviation Technical University, K. Marks Str. 12, Ufa, Russia
sivova.ae@net.ugatu.su, vulfin.alexey@gmail.com, mironovconst@gmail.com, kirillova.andm@gmail.com

Keywords: Keyboard Handwriting, Hidden Authentication, Neural Network Classifier.

Abstract: The problem of continuous hidden user authentication based on the analysis of keyboard handwriting is considered. The main purpose of the analysis is to continuously verify the identity of the subject during his work on the keyboard. The aim of the work is to increase the efficiency of hidden user authentication algorithms based on a neural network analysis of a dynamic profile, formed by keyboard handwriting. The idea of user authentication using keyboard handwriting is based on measuring the time of keystrokes and the intervals between keystrokes, followed by comparing the resulting data set with the stored dynamic user profile. Studies have shown that analyzing the average value of the time each key is pressed is inefficient. It is proposed to analyze the holding time of a combination of several keys and the time between their presses. An approach in which not the times of pressing individual keys, but the parameters of pressing the most common letter combinations are analyzed, will increase the accuracy of recognition of dynamic images. An algorithm and software implementation for Russian keyboard layout have been developed, experiments conducted on field data allow us to conclude that the proposed method is effectively used to authenticate the user using keyboard handwriting.

1 INTRODUCTION

User authentication based on tracking his or her behavioral features is a relatively novel technique. Behavioral features are acquired, when the user is working with various manipulators: a computer mouse, keyboard, etc. The term “user information handwriting” (UIH) proposed means the style of the work with such manipulators [1]. UIH tracking allow us to define a unique pattern, which may be used as a mean of authentication or assessing the user’s state, level of computer literacy etc.

The aim of our work is to increase the efficiency of hidden user authentication based on a neural network analysis of a keyboard handwriting. The sub-tasks of our work are the following:

1) Development of an algorithm for analyzing the user's keyboard handwriting with use of neural network.
2) Development of a system for hidden authentication with use of above-mentioned algorithm.
3) System evaluation based on accumulated data.

Margins, column widths, line spacing, and type styles are built in. Some components, such as multileveled equations, graphics, and tables are not prescribed, although the various table text styles are provided. The formatter will need to create these components, incorporating the applicable criteria that follow.

2 BIOMETRIC AUTHENTICATION BASED ON KEYBOARD HANDWRITING

Keyboard handwriting is a dynamic biometric pattern including speed of typing, use of the main and additional parts of the keyboard, specific keystrokes and specific techniques and methods of working with the keyboard [2]. With the improvement of keyboard skills, individualism of the keyboard handwriting also grows [3]. The listed individual features are a part of the dynamic user profile and may be used to authenticate the user. The methods of continuous hidden keyboard monitoring make it possible to detect the substitution of a legitimate operator and block the system from intruder intrusion. The probability of false recognition, when using 5-letter word, is about 10^{-33} [4].

Dynamic authentication systems are capable to make biometric patterns hidden. The attacker is not able to use the previously prepared dummy (which is possible, when using static patterns such as fingerprint). The main disadvantage of dynamic biometric systems is that their functioning is affected by the psychophysiological state of a person [5, 6, 7]. He or she may be worried or calm, tired or alert, healthy or sick, etc.

Analyzing a user's dynamic profile allow the system to verify continuously the identity of the subject and to control that this particular subject is working on the computer. The principle of continuous authentication consists measuring the durations of keystrokes and intervals between them. These data are compared to available pattern of the user. According to a large-scale study of this approach, conducted by The National Institute of Standards and Technology, USA [8] the probability of correct recognition for users with established keyboard skills is 98%, which is enough for practical applications.

Keyboard handwriting authentication system should include three modules [9, 10, 11]:

1) Keylogger, which tracks keystrokes.

2) Module for generating reference templates for the handwriting based on the data from the keylogger. This template is generated, while the user is working on the computer.

3) Authentication Decision Module, which analyzes the characteristics of the current user and compares them with a reference sample.

Let:

$n = 1 \ldots 1024$ is the number of all possible combinations of two keys in Russian alphabet (“а”, “б”, ..., “я”);

T_i – the experimental time interval between keystrokes of the n-th combination;

T_i^R – the reference time interval between keystrokes of the n-th combination.

Then the feature vector of the k-th user, generated from the average values of keystrokes of the i-th key, is determined as (1):

$$T_i^{R.avg} = \frac{1}{m}\sum_{j=1}^{1} T_{ij}^{\,R}\,, \qquad (1)$$

where m – the number of keystrokes of the i-th key. The vector may be expanded with the variance or mean square deviation for the i-th key.

The state-of-the-art algorithm for obtaining the vector of dynamic characteristics of the user consist of the following steps:

1) Generation of the reference feature vector for all K users: $V_k = \left\{T_i^{R.avg} \mid i = \overline{1,n}\right\}$, $k = \overline{1,K}$.
2) Formation of a feature vector based on user signature: $V_k = \left\{T_i^{avg} \mid i = \overline{1,n}\right\}$.
3) Search for the most similar vector in the database (DB).

The disadvantage of the state-of-the-art is that it analyzes the average values of the retention time of each individual key and the time after it is pressed. If we consider the location of the keys on the keyboard, we can conclude that the time between keystrokes of adjacent keys will be significantly lower than the time of keystrokes located more remotely from each other. Therefore, the distribution of the collected parameters would not be normal. So, using the obtained values as a vector of biometric features is inefficient.

We propose to analyze holding time and the timeout between presses within the most widespread combinations of several keys. This may increase the accuracy of pattern recognition, since the user's actions in this case are automatic and the parameters would be normally distributed.

We propose to analyze the typing time of N-graphs: sequences of several keys pressed in a row. Analysis of digraphs, i.e. sequence of two keys pressed in a row, allow determination of three indicators: the holding time of two keys and the time between them. To classify the obtained values, it may not be enough, therefore, it is proposed to use N-graphs of higher dimension.

3 ALGORITHM FOR ANALYSIS OF KEYBOARD HANDWRITING

3.1 Biometric Features of the Keyboard Handwriting

The approach for identifying a subject based on continuous hidden authentication of computer system user in the process of working at a computer is proposed. As identification characteristics, the features of the user’s work with the keyboard are used – his keyboard handwriting, which is characterized by the key holding times (KHT) and the times between keystrokes (TBK). These characteristics can be measured by a standard keyboard.

Retention time also depends on overlays. Overlapping keystrokes occurs when one key has not yet been released and the other is already being pressed. There is a tendency to increase the number of overlays with increasing dialing speed. The vast majority of overlays occur when the keys of adjacent letters in a word are pressed with different fingers. However, with very fast sliding dialing, overlays are also possible. Out of the total amount of text entered by the user during the working day, it is proposed to process not individual keystrokes, but the so-called N-graphs – trigraphs and tetraphs – sequences of three or four consecutive keys.

The controlled input parameters are the reference values of Key Holding Times (KHT) t_1, t_2, $t_3, \ldots, t_n$ the for each key in the reference, as well as the time intervals between pressing adjacent keys (times between keystrokes, TBK) $t_{12}, t_{23}, t_{34}, \ldots t_{(n-1)n}$, i.e. exclusively time parameters, which may be measured by a standard keyboard.

KHT also depends on overlays. Overlapping keystrokes occurs when one key has not yet been released and the other is already being pressed. There is a tendency to increase the number of overlays with increasing dialing speed. Most overlays occur when the keys of adjacent letters in a word are pressed with different fingers. However, with very fast sliding dialing, overlays are also possible. In case of overlapping, the parameter $t_{(n-1)n}$ becomes negative. The controlled parameters t_n and $t_{(n-1)n}$ significantly depend on how many fingers are used during typing, as well as on user-specific combinations of typing movements.

An artificial temporary function, which represent the entire process of typing a phrase and include all the necessary information about the user's keyboard handwriting, is shown on Figure 1.

Let the user enter a phrase containing n characters over a period T from the keyboard. When this phrase is entered, $r = n + m$ keyboard events will occur: n key holdings and $n = m - 1$ pauses between holds. The temporary function at the time t_i will take the value $q(t_i)$, where q is scan-code i.e. key identifier on the keyboard. Overlapping is interpreted as a sum of two scan-codes of the pressed keys.

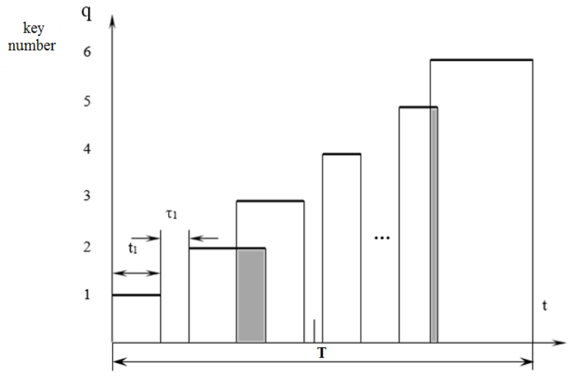

Figure 1: Time diagram of phrase typing.

The process of entering control phrase with $r = 11, m = 6, n = 5$, is illustrated by a time diagram (Figure 1). Temporary layout of the process is individual for each user and acts as a standard for keyboard handwriting.

As a feature vector of biometric (keyboard) features of *V*, we use the values of the function $q(\Delta t)$, where Δ t is determined as (2).

$$\Delta t = \frac{t_1 - t_0}{n} \tag{2}$$

The typing of certain N-graph for various users differs. Therefore, it is necessary to bring the feature vector to a single length. For this purpose, vector normalization is applied. Thus, the length of the vector of biometric features for each of the users will be equal to n – the number of samples equal to 32.

Development of a reference sample for one user require a series of L samples, which constitute a representative sampling for the s-class $\Psi^{(s)} = \{V_i\}, i = \overline{1, L}$.

In general, the system can include multiple users $K = \{k_1, k_2, \ldots, k_M\}$. Each user is represented by its reference pattern and associated with a certain class from the set $s = \{s_1, s_2, \ldots, s_M\}$. Thus, an unambiguous mapping from the set of users $\{K\}$ to the set of classes $\{s\}$ is developed. Development of the reference samples for M legitimate users require M training samplings respectively $\Psi^{(s_1)}, \Psi^{(s_2)}, \ldots, \Psi^{(s_M)}$.

When the system is in the authentication mode, unknown user (x) presents a sample of keyboard handwriting as a vector of biometric parameters $V^{(s_x)} = \{v_j\}$, $j = \overline{1, N}$. The system should form a description of the unknown x-class on the basis of the vector $V^{(s_x)}$, compare it with the standards of all users registered in the system $\{k_1, k_2, \ldots, k_M\}$ and make the authentication decision based on the results.

In this formulation, the task is classifying the vector $V^{(s_x)}$ into $M+1$ exclusive classes: M classes from the set $s = \{s_1, s_2, \ldots, s_M\}$, and $(M+1)$-th class reserved for all other users, united by the concept of "aliens". If there is a procedure for preliminary authorization of users, the task is simplified and reduces to the classification of the vector $V^{(s_x)}$ into two classes: s_o – "own", that is, belonging to any class from $\{s\}$, and s_a – "alien", that is, not belonging to any class from $\{s\}$.

3.2 Selection of Informative Values of Biometric Features

Only the most frequent N-graphs are processed by the algorithm. A frequency dictionary of trigraphs and tetragraphs is generated in order to choose the most frequent of them. These N-graphs are selected for analysis. In the authentication mode users are authenticated based on the analysis of these N-graphs.

3.3 Model of User Authentication

Authentication decision is made based on the difference between actual data and reference pattern. The input information consists of the values of KHT and TBK.

The minimum number of neurons in the hidden layer, which provide the solution to the interpolation task, is determined by the expression (3) [12]:

$$n_2 = \operatorname{int}\left[\frac{m(R-1)}{n+m+1}\right], \qquad (3)$$

where n_2 – the number of neurons in the hidden layer;
n – the number of neurons in the input layer;
m – the number of neurons in the output layer;
R – the dimension of the training sampling. Operation int() means rounding up to an integer.

Substituting the values in the (3), we get:

$$n_2 = \operatorname{int}\left[\frac{3(30-1)}{3+5+1}\right] = \operatorname{int}(9.7) = 10$$

The classical neural network for biometrical authentication is shown in Figure 2.

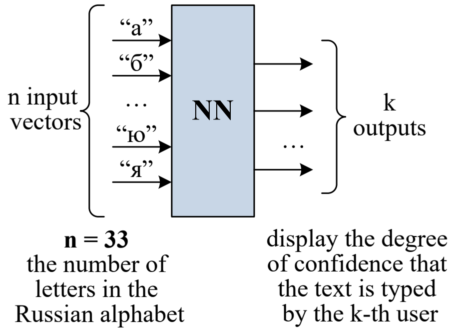

Figure 2: Classical neural network for biometrical authentication.

Average KHT are analyzed in this approach, which is inefficient. In the proposed system, the KHT and TBK of several consecutively pressed keys act as a vector of biometric features. In addition, we propose to use the modular structure of a neural network, in which each network make a decision for only one of the selected N-graphs. The modular approach allows us to divide the authentication task into subtasks, solve them individually with separate neural networks, and then combine the results.

Large neural network can suffer from interference, as new data can dramatically change existing connections. The modular approach makes it easy to scale the network, because adding or removing modules for a specific N-graphs is possible without retraining the entire network.

Depending on which feature vector is fed to the input of the neural network, it is proposed to use two approaches for the modular structure of the network.

First approach. A vector of user biometric features normalized to 32 samples is supplied to the input of the first neural network (let us call it "network of the first type"). The first network is responsible for recognizing the input N-graph. It activates the second network ("network of the second type"), which was trained on this image. The second network determines from which user the biometric feature vector was received, containing the number of representations of the recognizable N-graph. The output is the values characterizing the degree of confidence that the text was typed by each of k users. The final decision is made by the decision unit, analyzing the data received from the neural networks. Thus, the decision to authenticate the user will be made based on data received from several neural networks at once. The structural diagram of the described approach is shown in Figure 3.

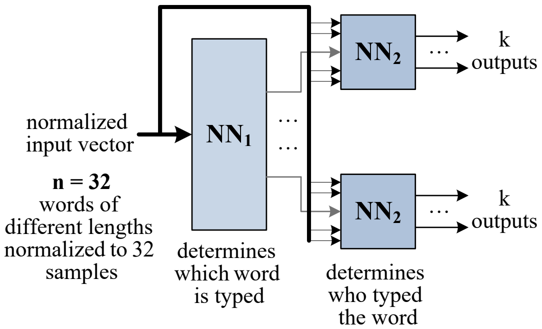

Figure 3: Diagram of the first approach.

Second approach. If each network of the second type is trained on only one N-graph, only the values of the KHT and TBK can be fed into the network input, without considering, which sequence of characters was typed.

Thus, it is possible to throughout the first network and use another classifier instead of it. The input of the classifier consist of KHT, TBK and the identifier of the N-graph. Based on these data, the classifier the neural network of the second type. The final decision is also made by the decision unit as in the first approach.

The structure of the second approach is shown in Figure 4.

Input vector is much smaller in the second approach than in the first one and consists of only 5 or 7 signs for trigraphs and tetraphs, respectively, compared to 32 in the first approach. This will allow the neural network to learn faster and with smaller samplings; moreover, the probability of getting into local minima with this approach is reduced.

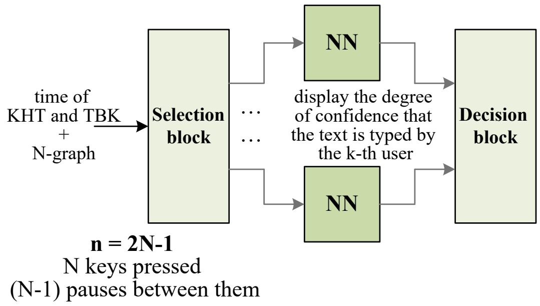

Figure 4: Structural diagram of the second approach.

4 SYSTEM FOR HIDDEN AUTHENTICATION

4.1 Algorithm of Hidden Authentication

The hidden authentication system consists of three modules [13, 14, 15]:

- Module for collecting the data;
- Module for data preprocessing and generating feature vector;
- Module for learning and recognition using neural network classifier.

The block diagram of the authentication system is shown in Figure 5.

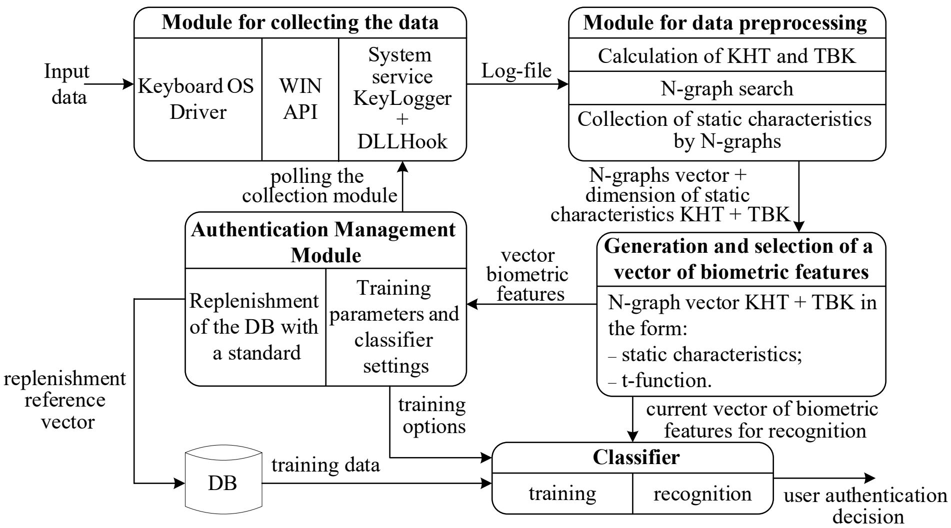

Figure 5: Structural diagram of the second approach.

Let us consider each of the modules more precisely.

Biometric authentication is based on the creation of reference representations of identifiable users. A reference is created when the system is in data collection mode. Registration of keyboard handwriting is carried out by the KeyLogger software module (keylogger).

The algorithm of the pre-processing and feature generation module is depicted as a flowchart in Figure 6. This module analyzes the data obtained using the keylogger. The logbook is analyzed line by line and a list of N-graphs is compiled, including the characteristics of KHT and TBK. The obtained values are used as an input vector of biometric features for the neural network.

The second and third modules are logically combined and executed sequentially. The resulting set of examples is divided into learning sampling and test sampling for cross-validation. Then the procedure of supervised training of the neural network is applied.

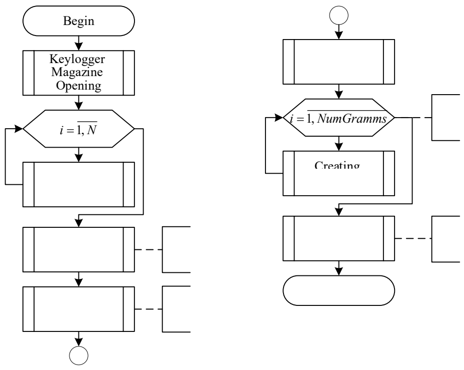

Figure 6: Algorithm for pre-processing and generating feature vector.

The general algorithm for obtaining feature vector with the subsequent provision of the obtained training sampling is presented as a flowchart in Figure 7.

4.2 Functioning of the System for Hidden Authentication

The system consists of several modules, which carry out their work invisibly for the user.

KeyLogger write a specialized log-file, which include timing of key pressing and two versions of key codes: scan code and virtual code used by the operating system to identify keys. Thus, each line in the log include the following data: scan code, key status, timing datum and virtual key code.

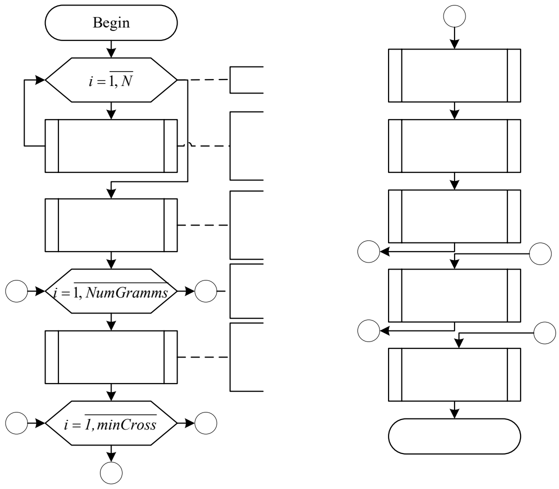

Figure 7: Algorithm for training and validation of the classifier.

The next module is used for creating biometric feature vector, which will be given to the input of the neural network. The program extracts the data from the keylogger log. They are parsed line by line, and all values are entered into an array of sample structures. For each keystroke, a search is made for the moment it is pressed, given that the first keystroke begins at time $t = 0$. The time of key releasing is added to the sample array, and the line that previously contained this parameter is deleted.

The next stage of the program is building an array of N-graphs. The values of the virtual key codes of the sample array are analyzed for this purpose. Starting with the first element in step 1, the values of N consecutive keys are entered into a new word array. Only N-graphs, which are found in the text more than 15 times and typed by all users, are selected All other N-graphs are deleted. For the subsequent analysis, only the values of the most frequently encountered N-graphs are left with the data on pressing and releasing each key that make up the N-graph itself.

Since the time of typing a phrase is different for all users, normalization of the vector and the time chart by the number of samples n equal to 32 is carried out. As a result, a normalized vector of biometric features is constructed.

4.3 Test Results for the Prototype of Hidden Authentication System

Jarque-Bera test [16] and Lilliefors test [17] allow us to validate the hypothesis that the variables analyzed in the classical approach do not obey the law of the normal probability distribution. The experiment showed that analysis of the average timing for single key pressing is inefficient. The easiest graphical way to check the nature of the data distribution is to plot a histogram. If it has a bell-shaped symmetrical appearance, we can conclude that the analyzed variable has an approximately normal distribution. For example, Figure 8 shows the histograms of the distribution of the retention time for the keys “a” and “б”.

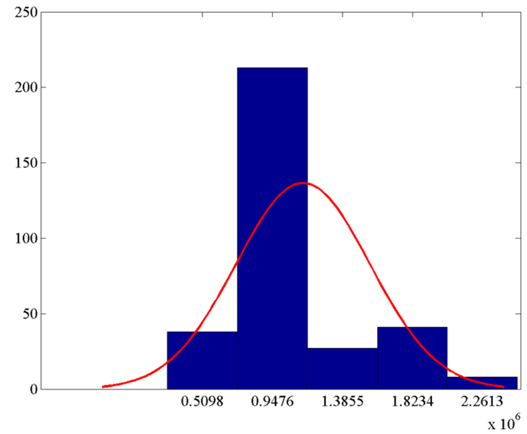

Figure 8: Histograms of the distribution of the retention time for the keys “a”.

Another graphical way to check the nature of the distribution is to build the so-called quantile plots (Q-Q plots, Quantile-Quantile plots). The quantile graphs for the distribution of the retention time of the “a” keys are shown in Figure 9.

The obtained graphs confirm the hypothesis that the average values for the retention time of individual keys do not obey the normal distribution are obtained.

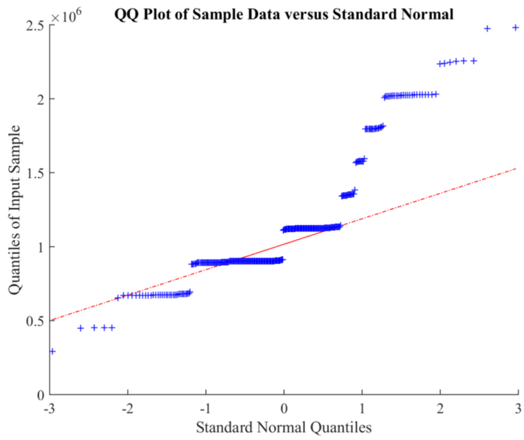

Figure 9: Graphs of the distribution quantiles for the retention time of the keys “a”.

Figure 10 show the distribution of the average time that the “a” key was pressed, depending on the phrase in which the letter was used.

In addition, the time between pressing two adjacent keys, depending on the typed combination, is also different, as shown in Figure 11.

As can be seen from Figures 10, 11, the average time of keystrokes in different combinations is different, as well as the time between holding the keys. Therefore, it was proposed to use KHT and TBK of the most key combinations.

During the experiment, User 1 used the “ст” combination 75 times in his work behind the keyboard. The ordered values of the key holding time “с” in the specified combination are shown in Figure 12.

In the same key combination, the ordered time values between the keystrokes “с” and “т” are shown in Figure 13.

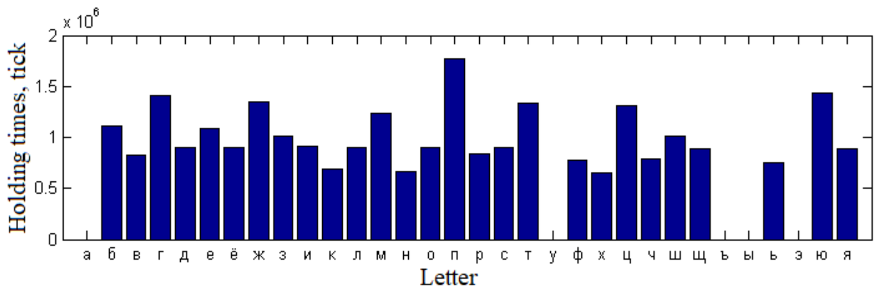

Figure 10: Average holding time of the “a” key for pairs “aa” ... “aя”.

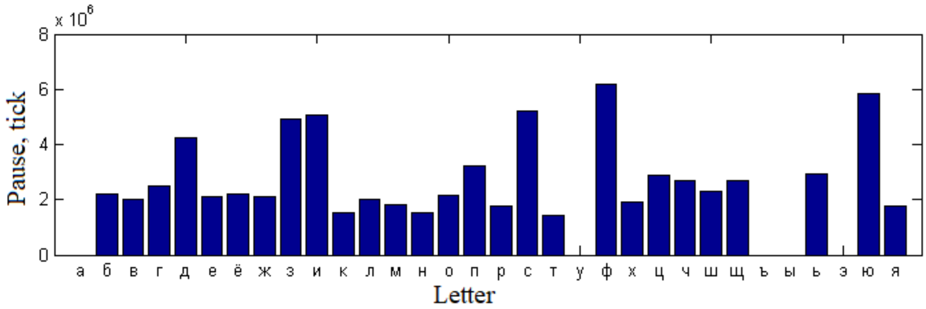

Figure 11: Average time between keystrokes for pairs “аа” ... “ая”.

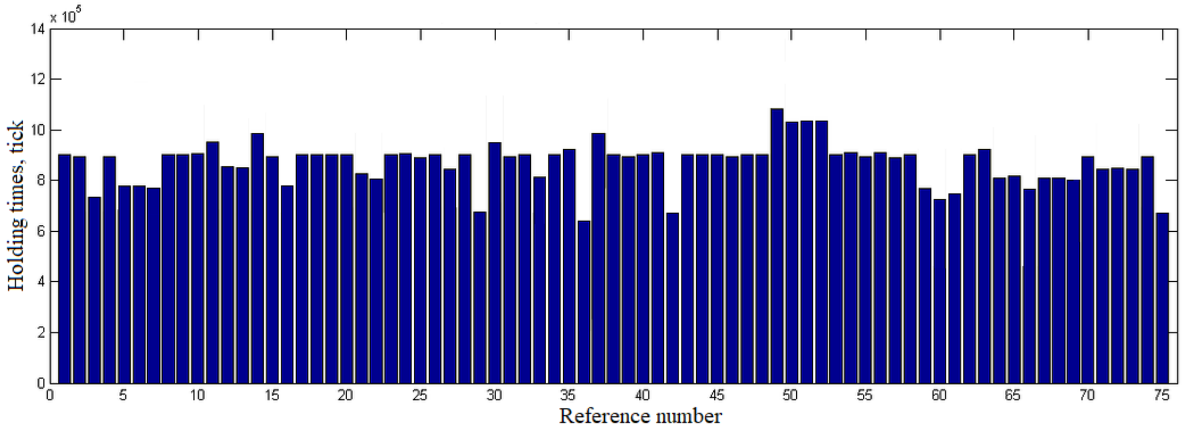

Figure 12: Ordered values of the key holding time “с” in the combination “ст”.

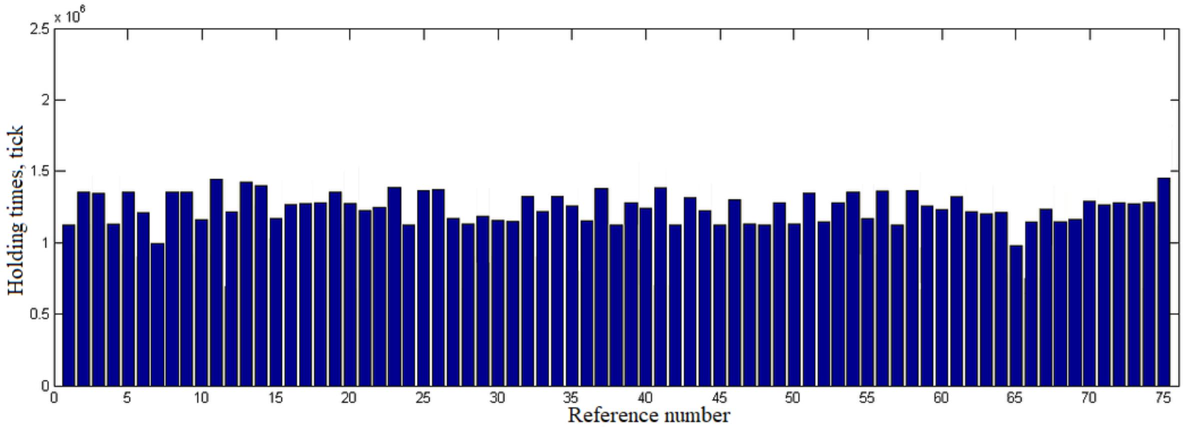

Figure 13: Ordered values of time between keystrokes “с” and “т” in the combination “ст”.

As can be seen from Figures 12, 13, the user types the same key combination in a similar way, therefore, the data on the typing time of corresponding N-graph can be used for training and testing a neural network.

Three users took part in the experiment. Trigraphs selected by frequency of occurrence, as well as a general list, are presented in Table 1.

The whole set of obtained vectors was divided into 10 subsets for 10-validation. The results are listed in Table 2.

Table 1: Selected trigraphs for various users.

User 1		User 2		User 3		Total information
N-graph	qty	N-graph	qty	N-graph	qty	
али	15	ани	17	або	16	ени
ана	16	ени	20	ани	28	льн
ель	23	ите	15	еле	23	ния
ени	24	льн	15	ени	19	нны
ите	15	мен	15	ефо	19	про
льн	24	ния	17	леф	19	
нал	15	нны	17	льн	19	
ния	18	при	15	ния	24	
нны	17	про	19	нно	22	ени
нов	16	чен	17	нны	24	льн
ные	17			ног	15	ния
ных	21			ной	29	нны
ова	17			ные	16	про
ого	16			ова	23	
ост	28			ого	22	
пол	20			онн	19	
при	17			оро	16	
про	19			ост	30	
				пол	18	
				про	24	
				ред	17	
				ров	17	
				ств	17	
				фон	19	

Table 2: Trigraph recognition percentage.

Trigraph		ени	льн	ния	нны	про	**Total**
Quantity		63	58	59	58	62	
Correct recog-nition (%)	Method 1	96.34	98.48	100	99.81	100	98.926
	Method 2	99.12	100	99.62	98.48	98.6	99.164

The table shows the values of the correct user recognition for each selected trigraph when using two methods. The results obtained using both methods are averaged and entered in the final column of the table. The results of the test in the first two passes during training on the N-graph "ния" are the following (Table 3):

Table 3: Inaccuracy matrix.

18	0	0
0	17	0
0	0	24
0	**0**	**0**
Sensitivity = 1 Specificity = 1 Correctness = 100%		

16	0	1
0	16	0
0	0	21
0	**0**	**0**
Sensitivity = 1 Specificity = 0.94 Correctness = 98.15%		

5 CONCLUSIONS

The following results were obtained:

- algorithm for transforming keyboard handwriting log into feature vector has been developed;
- algorithm for analyzing the user's keyboard handwriting based on neural network classifiers has been developed;
- modular structure of the neural network has been developed that correctly recognizes users in 99.164 % of cases;
- prototype system of hidden user authentication was developed.

Proposed system allows one to:

- authenticate the user according to the typed text (i.e. answer the following question: is it really that particular employee or someone else?);
- detect the substitution of the user in cases where an employee without access rights is trying to get the access through the computer of the qualified colleague;
- find the author of a specific text – which of the users in the company entered text on this PC in a suspicious period of time;
- identify the user in an atypical state and the specific period of time during which the user remained in this state;
- prevent the attempt of unauthorized access to the system in cases where the attacker managed to circumvent all previous lines of protection.

ACKNOWLEDGMENTS

The reported study was funded by RFBR according to the research project No. 20-08-00668 "Development and research of the methodology, models and methods of complex analysis and cybersecurity risk management of process control systems of industrial facilities using cognitive modeling technology and data mining".

REFERENCES

[1] M.N. Eshwarappa and M.V. Latte, "Multimodal biometric person authentication using speech, signature and handwriting features," International Journal of Advanced Computer Science and Applications, Special Issue on Artificial Intelligence. 2011, pp. 77-86.

[2] T.V. Zhashkova, O.M. Sharunova and E.Sh. Isyanova, “Neural network identification of a person’s personality type by keyboard handwriting,” International Student Scientific Herald, 2015, no. 3.

[3] M. Cortopassi and E. Endejan, “Method and apparatus for using pressure information for improved computer controlled handwriting recognition, data entry and user authentication,” U.S. Patent, no. 6,707,942, 24 March 2004.

[4] S.M. Didenko, “Development and research of a computer model for the dynamics of the user-mouse system”, Tyumen, 2007.

[5] ГОСТ Р 54412-2011 – ISO/IEC/TR 24741:2007 “Information technology. Biometrics. Biometrics tutorial,” Standartinform, 2012.

[6] ГОСТ Р ISO/IEC 19794-2008 “Automatic identification. Biometric identification. Formats for the exchange of biometric data,” Standartinform, 2009.

[7] ГОСТ Р ISO/IEC 1978-4-2014 “Information technology. Biometrics. Biometric software interface,” Standartinform, 2016.

[8] A.V. Skubitsky, “Analysis of the applicability of the method of reconstructing dynamic systems in biometric identification systems by keyboard handwriting,” Informacionnye tehnologii, vol. 6, no. 1, 2008.

[9] R. Sharipov, M. Tumbinskaya and A. Abzalov, “Analysis of Users' Keyboard Handwriting based on Gaussian Reference Signals,” 2019 International Russian Automation Conference (RusAutoCon). IEEE, 2019, pp. 1-5.

[10] O. Vysotska and A. Davydenko, “Keystroke Pattern Authentication of Computer Systems Users as One of the Steps of Multifactor Authentication,” International Conference on Computer Science, Engineering and Education Applications. Springer, Cham, 2019, pp. 356-368.

[11] R. Chen, S. Kutten and E. Biham, “User authentication system and methods,” U.S. Patent no. 9,680,644, 13 June 2017.

[12] V.I. Vasiliev and B.G. Ilyasov, “Intelligent management systems. Theory and practice,” tutorial. M: M.: Radiotehnika, 2009, 392 p.

[13] Z.H.U. Yunzhou, and X. Jiang, “System and method for user authentication with exposed and hidden keys,” U.S. Patent no. 8,132,020, 6 March 2012.

[14] A. Schwartz and G.A. Woodward, “Composition and method for hidden identification,” U.S. Patent no. 4,767,205, 30 August 1988.

[15] N. Harun, W.L. Woo and S.S. Dlay, “Performance of keystroke biometrics authentication system using artificial neural network (ANN) and distance classifier method,” International Conference on Computer and Communication Engineering (ICCCE'10), IEEE, 2010, pp. 1-6.

[16] T. Thadewald and H. Büning, “Jarque–Bera test and its competitors for testing normality–a power comparison,” Journal of Applied Statistics, vol. 34, no. 1, 2007, pp. 87-105.

[17] N.M. Razali and Y.B. Wah, “Power comparisons of shapiro-wilk, kolmogorov-smirnov, lilliefors and anderson-darling tests,” Journal of statistical modeling and analytics, vol. 2, no. 1, 2011, pp. 21-33.