How to cite this article:

Venugopal, P. C., & Viji, K. S. A. (2019). Applying empirical thresholding algorithm for a keystroke dynamics based authentication system. *Journal of Information and Communication Technology, 18*(4), 383-413.

APPLYING EMPIRICAL THRESHOLDING ALGORITHM FOR A KEYSTROKE DYNAMICS BASED AUTHENTICATION SYSTEM

[1]Priya Chaliyath Venugopal & [2]Kamalan Saroja Angel Viji
[1]Department of Computer Science & Engineering, Noorul Islam Centre for Higher Education, India
[2]Department of Computer Science & Engineering, College of Engineering, India

priyacv89@gmail.com; angelhevin@yahoo.com

ABSTRACT

Through the application of a password-based authentication technique, users are granted permission to access a secure system when the username and password matches with that logged in database of the system. Furthermore, anyone who provides the correct username and password of a valid user will be able to log in to the secure network. In current circumstances, impostors can hack the system to obtain a user's password, while it has also been easy to find out a person's private password. Thus, the existing structure is exceptionally flawed. One way to strengthen the password-based authentication technique, is by keystroke dynamics. In the proposed keystroke dynamics based authentication system, despite the password match, the similarity between the typing pattern of the typed password and password samples in the training database are verified. The timing features of the user's keystroke dynamics are collected to calculate the threshold values. In this paper, a novel algorithm is proposed to authenticate the legal users based on the empirical threshold values. The first step involves the extraction of timing features from the typed password samples. The password training database for each user is constructed using the extracted features. Moreover, the empirical threshold limits are calculated from the timing features in the database. The second step involves user authentication by applying these threshold values. The

Received: 21 August 2018 **Accepted:** 28 April 2019 **Published:** 26 September 2019

experimental analyses are carried out in MATLAB simulation, and the results indicate a significant reduction in false rejection rate and false acceptance rate. The proposed methodology yields very low equal error rate of 0.5% and the authentication accuracy of 99.5%, which are considered suitable and efficient for real-time implementation. The proposed method can be a useful resource for identifying illegal invasion and is valuable in securing the system as a correlative or substitute form of client validation.

Keywords: Authentication, computer security, empirical threshold, feature extraction, keystroke dynamics.

INTRODUCTION

Due to the extended application of computers in financial, mechanical, military and individual exercises, a stringent framework to safeguard the storage of essential data has never been more imperative (Hemanidhi & Chimmanee, 2017; Hussien, Muda, & Yasin, 2018). Unauthorized access to log into secure systems can lead to significant financial misfortune, hence it is mandatory to construct high quality security measures in these computer networks (Mihajlov, Jerman-Blažič, & Ciunova Shuleska, 2016; Mohsin, Bakar, Hamdan, & Abdul, 2018). Typically, authentication is defined as the method of validating a legal user to access the secure network. The validation can be accomplished by cross-checking the distinct characteristics of a person. Based on the user's distinct characteristics, the authentication methods can be classified as token-based, knowledge-based and biometrics-based. The summary of these methods is shown in Table 1.

Table 1

Summary of various authentication methods

Method	Advantage	Disadvantage	Example
Token	Simple deployment	Loss	Swipe cards
	Cheap	Theft	Credit cards
			Mini devices
Knowledge	Cost-effectiveness	Intrusion	Password
	Simple implementation	Forgotten	PIN
		Hacked	Pattern

(continued)

Method	Advantage	Disadvantage	Example
Biometrics	Uniqueness	Need external hardware	Fingerprint
	Accuracy	Expensive	Face
	Deter sharing	Invasive	Iris
			Voice
			Keystroke*

*no external hardware needed – inexpensive

The most widely implemented approach in authenticating legitimate clients is by providing a separate username and a confidential personal identification number (PIN) or password. Unfortunately, anyone who inputs the right username and PIN of a legitimate client can be permitted to sign in to the secure networks. Impostors can hack into the system to obtains the client's password, and it has also been easier to discover a person's private password. Hence, this framework is exceptionally flawed. One way of fortifying the password based authentication technique is by using biometrics.

Biometric is an art of recognizing people by a particular biological or behavioral trademark such as face, speech, fingerprint, eye iris, signature and voice (Albashish, Sahran, Abdullah, Alweshah, & Adam, 2018; Boopathi & Aramudhan, 2017; Kaewwit, Lursinsap, & Sophatsathit, 2017; Mohamed, Zainudin, Sulaiman, Perumal, & Mustapha, 2018; Odei-Lartey et al., 2016; Van Zoonen & Turner, 2014; Zahari & Zaaba, 2017). The singularity of a client's biometric can lower the risk of account theft, as there is no compelling reason to memorize or preserve the secret password. However, the biometric-based authentication method would require extra equipment to record the particular biological trademark. Subsequently, this causes the security system to be more costly.

Keystroke dynamics is different, and involves the advance application of biometrics to study and record the person's individuality (Leggett & Williams, 1988; Leggett, Williams, Usnick, & Longnecker, 1991). The special type of biometrics that utilizes the typing rhythm of a character on the keyboard is called keystroke dynamics. Distinctive information can be obtained from the person's typing rhythm such as temperature and pressure of the client's fingers when they depress the keys, and timing information of subsequent keystrokes (Kotani & Horii, 2005). The temperature and pressure information does not change significantly between people and would require additional devices to capture the information, however the timing data of keystroke dynamics is the most widely recognized feature extracted from the typing rhythm. Although

there are various biometric validation frameworks, the keystroke validation method does not require any extra equipment for client identification (except for a typical console keypad needed to type the password). Typically, a computer system used for any biometric-based validation system has a built-in keyboard. Essentially, this framework merely captures the typing rhythm of the password entered during the login session.

After the introduction, a brief review of keystroke dynamics based authentication (KDA) system is presented. Further, the pictorial representation of essential timing features used in the proposed method, the proposed authentication system, preparation of training password database, the novel empirical thresholding algorithm and the database updating algorithm with the supportive formulas are explained in detail. The experimental analyses and the simulation results that show the superiority and robustness of the proposed KDA method are presented towards the final section along with the advantages, robustness, accuracy, and efficiency of the proposed KDA system based on the inference achieved from the various case studies.

KEYSTROKE DYNAMICS BASED AUTHENTICATION

The KDA method can be implemented in two distinct ways: static and dynamic (Furnell, Morrissey, Sanders, & Stockel, 1996; Gunetti & Picardi, 2005). The idea of different keystroke sequences for a single word has been introduced to seek the typing variance between various keystroke sequences of the word (Syed, Banerjee, & Cukic, 2016). This method suggests that different keystroke sequences can hold enough discriminative data to authenticate legal clients efficiently, and can be employed to both static and dynamic mode of authentication systems.

In the dynamic method, otherwise known as the free text method, the authentication depends solely on the client's typing rhythm irrespective of the typed content (Ahmed, 2009). The dynamic mode of KDA is primarily implemented as an anomaly detector to unsure the validity of the legal person throughout the interfacing session within the network. Moreover, clients are not required to recall predefined contents such as username, secret key or password. The primary function of the dynamic method is to guarantee that the legal person is still the same after the user has logged on. This type of authentication should be possible by registering the person's keystrokes while the person is interacting within the secure network. This mode of KDA is also known as continuous mode. The implementation in dynamic mode should be

done using an arbitrary database because keystroke timing features recorded during the interaction phase is arbitrary. The authentication should be carried out consistently while logged on to the secure computer system, and is not restricted to any session timeout.

In static mode, the keystrokes of a person is recorded and stored in the database during the sign-up phase, and then further validated during the login phase (Robinson, Liang, Chambers, & MacKenzie, 1998). The database used in the static mode of KDA is invariable (Killourhy & Maxion, 2009). The static mode is mainly employed as an entrance control of the authenticating system and avoids password sharing. The static KDA method further strengthens the network access security by using clients' login username and password data. The static KDA scheme has been implemented using Gaussian mixture method (GMM) and neural network (NN) classifications with successful authentication rate of 90% and 99% respectively. Most studies carried out on the application of KDA methods have been focused on the static mode of authentication (Al Nuaimi & Abdullah, 2017; Chen & Chang, 2004; Cho, 2006; Das, Mukhopadhyay, & Bhattacharya, 2014; Hosseinzadeh, Krishnan, & Khademi, 2006; Joshi & Phoha, 2007; Leggett et al., 1991; Maazouzi, Mohajir, & Achhab, 2017; Memon, 2017; Shehab, Khader, & Laouchedi, 2018).

In the static method, client authentication is carried out in two steps; firstly, the password entered by the client is validated against the system database and secondly, the typing rhythm of the correctly entered password is verified (Das et al., 2014). Typically, the accuracy and security level of an authentication system is dependent on the type of verification method used. The advantages of the static method over dynamic method are as follows:

(i) The dynamic method has a single stage authentication process that uses only the typing similarity of the text content, whereas the static method has a two stage authentication process. Therefore, the static method poses difficulties for impostors to access the system as the authentication process not only includes a correct password match but also the distinctive text typing rhythm of the clients. Therefore, the KDA method with the static mode is more secure than the dynamic mode.

(ii) One of the disadvantages of the dynamic method is that the impostor may get authenticated to the secure computer system or app if the invader tried to log on to the security system with different typing rhythms at different intervals. Alternatively, the static method has a specified number of password attempts and few minutes of session timeouts

before being logged in. Hence, if the number of password attempts exceeded the specified limit or the timing features did not match the user's database, then the user is permanently blocked or access will be denied temporarily for few a hours. Thus, the static method is more robust than the dynamic method.

(iii) Generally, the dynamic method consumes more data size to store the long free text content in the system's database, and subsequently causes a delay in the authentication process as the system needs to search and verify over the more extensive database. In the static method, there is a limit to the length of the password characters. For example, the password-based authentication system for a Google account is limited to a minimum of 8 characters to maximum 60 characters. Thus, the size of the database is smaller than the ones that applies the dynamic method, as it consumes lesser space in the system memory. Therefore, the static method can authenticate the user more efficiently and much quicker than the dynamic method.

(iv) Both static and dynamic methods do not demand any extra hardware to capture the keystrokes. Hence the KDA is inexpensive when compared to other biometric-based authentication systems. As a whole, the static method of authenticating a person is more secure, robust, faster and effective compared to the dynamic method. The overview of two modes of the KDA method is shown in Table 2.

Table 2

Modes of keystroke dynamics based authentication method

Mode	Advantage	Disadvantage
Dynamic	Unforgettable	High memory utilization
		Time-consuming
Static	Two-stage authentication	Password forgot
	More secure	
	Less memory utilization	

The KDA method based on free text was introduced in (Gunetti & Picardi, 2005), which employed the distance measures of each digraph and recorded a false acceptance ratio (FAR) of less than 0.005% and false rejection ratio (FRR) of less than 5%. A neural network based KDA approach for the free text was introduced in (Ahmed & Traore, 2014). The approach recorded a FAR of 0.0152%, FRR of 4.82%, and equal error rate (EER) of 2.46% in

heterogeneous conditions; while in homogeneous conditions, it recorded a 0% FAR, 5.01% FRR, and EER 2.13%. Since the EER is higher than 2%, the method is not suitable for real-time application. A comparative study was carried out in (Kang & Cho, 2015) to identify the best options with regards to type of input device, character length, and authentication algorithm. The study concluded multiple inferences which were; PC keyboard is the best device to capture the keystrokes, the authentication accuracy was enhanced by increasing character length or sample size, and relative (R) + absolute (A) measures reported the best performing algorithm. The event sequences based KDA algorithm was introduced and implemented in both static and dynamic modes (Syed et al., 2016). The feature normalization presented in (Syed et al., 2016) had many advantages such as simplicity, faster search, and retrieval of n-graphs. Furthermore, the research suggested that the proposed KDA had better authentication accuracy, however none of the results showed any evidences for the actual accuracy of the system. A KDA method using pairwise user coupling and machine learning algorithms were presented for both static and dynamic text content resulting to a moderate accuracy of 89.7% (Mondal & Bours, 2017). A KDA system based on user-adaptive feature extraction was proposed in (Kim, Kim, & Kang, 2018) and utilized the free text of 13,000 keystrokes per user. This method revealed excellent authentication accuracy of 99.5%. Unfortunately, there were disadvantages due to computational complexity and higher memory utilization from processing larger data sets.

Contribution of the Work

Thorough study of existing literature, with the exception of works conducted by (Gunetti & Picardi, 2005; Kim et al., 2018), have suggested that most KDA methods presented in the articles were constrained by high EER and low authentication accuracy (Ahmed & Traore, 2014; Kang & Cho, 2015; Leggett et al., 1991; Mondal & Bours, 2017; Syed et al., 2016). The significant contributions of the proposed study are the inclusion of three more timing features, sole user database with limited password length, empirical thresholding algorithm, and database update rule.

(i) Despite good authentication accuracy, a major restriction on distance-measure based KDA method is the dependence on other users' text samples to construct a single user's keystroke database (Gunetti & Picardi, 2005). Therefore, in the authentication stage, the keystroke dynamics of the typed text is verified against the database of all legal users. This scenario raises serious concern on the effectiveness and adaptability of the distance measures based KDA method. Furthermore, this may led to further problems during real-time application, whereby the authentication is conducted continuously. The validation is typically

carried out on the keystroke dynamics of the typed text of a user against their database. Hence, this study creates a user timing feature database from their typed text samples when a new user signs up for a net banking application. This database is not dependent on other users' keystroke features, but solely realizes on their own timing information of typed content.

(ii) The user-adaptive feature extraction based KDA technique has a low ERR of 0.5%, which is the best among other available KDA methods. However, this method uses 13,000 keystrokes to make a single user's database, essentially consuming large amounts of storage space to build the vast database for all users. The authentication speed will decrease, as this method would take more time to search and verify the timing features of the typed text over a vast sample database during user authentication.

(iii) Most KDA based studies utilized two timing information, namely duration of a keystroke and the latency between two consecutive keystrokes. However, if a person types a lengthy string of words fast, the timing information will overlap and would be insufficient to develop a quality KDA. Consequently, to overcome the overlapping problem, this paper added three other timing information which are; press-press time between depressing the first key and depressing the following key, the release-release time between releasing the first key and releasing the following key, and press-release time between depressing the first key and releasing the following key.

(iv) Generally, a good authentication algorithm is required to identify the similarities between the timing features of the typed password and the samples in the database. In this study, the verification of the typing rhythm of a password is accomplished by the proposed empirical thresholding algorithm. The proposed algorithm uses the keystroke timing features of a user's keystroke database, and then calculates the empirical threshold limits for each timing features. Based on the threshold limits, the verification of all timing features in a correctly typed password is completed. The timing value of each feature in the password would be between the lower and upper threshold limits of the corresponding timing feature.

(v) During user authentication, there would be occasional deviations in the typing rhythms when a user enters their password after a few hours, days, or weeks. The deviation occurs due to the changes in a user's typing pattern, particularly when it differs from the previously typed password samples in the database. Hence, it is difficult to authenticate the legal user accurately, which subsequently leads to a decreases in the authentication efficiency due to such invariant and fixed password samples. In order to resolve the invariant password database, a database

update rule is employed for each successfully authenticated user to meliorate the authentication accuracy and security level.

Therefore, the primary objective of the proposed KDA system based on the static mode, is to improve the accuracy, security level, and speed of client identification and authentication with the aid of empirical thresholding algorithm and database update rule.

TIMING FEATURE EXTRACTION

The typing sample is simply texts entered at a computer keyboard,whereby the timing data on the keystrokes are recorded. Timing data is represented by two primary measures: the time at which a key is depressed and the time at which a key is released. These timing measures are used to compute the timing features, i.e., duration of a keystroke, the latency between two consecutive keystrokes, press-press time, release-release time and press-release time (Kim et al., 2018). The rising and falling edges of a five-character password "hello" are shown in Figure 1. t_1 is the time at which a key "h" is depressed and t_2 is the time at which a key "h" is released. Each interval between the rising and falling edges of a key can be used as an element in the timing feature vector. Thus, unique features can be distinguished from a typical sequence of the word "hello".

Key hold time 'H': the time between depressing and releasing a key. For example, the hold time for texting a letter "h" is $H_1 = t_2 - t_1$.

Key latency time 'L': the time between releasing the first key and depressing the following key. For example, the latency time for texting a digraph "he" in a word "hello" is $L_1 = t_3 - t_2$.

Key press-press time 'PP': the time between depressing the first key and depressing the following key. For example, the PP time for texting a digraph "he" in a word "hello" is $PP_1 = t_3 - t_1$.

Key release-release time 'RR': the time between releasing the first key and releasing the following key. For example, the RR time for texting a digraph "he" in a word "hello" is $RR_1 = t_4 - t_2$.

Key press-release time 'PR': the time between depressing the first key and releasing the following key. For example, the PR time for texting a digraph "he" in a word "hello" is $PR_1 = t_4 - t_1$.

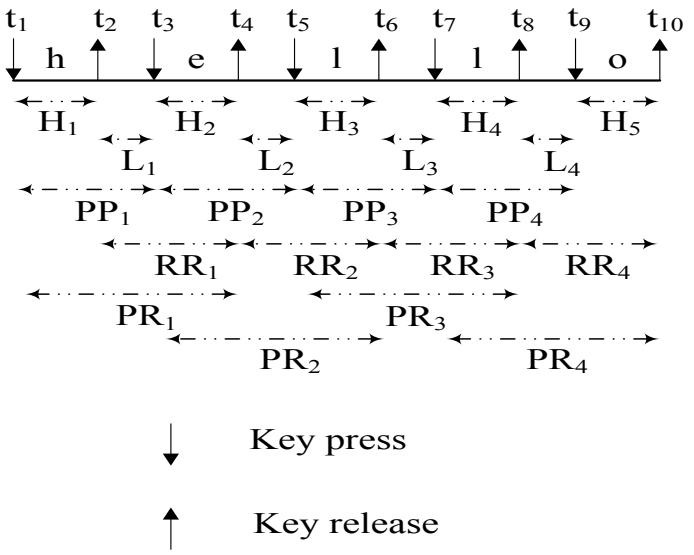

Figure 1. Keystroke Dynamics in Typing a Five Character Password "hello".

PROPOSED AUTHENTICATION SYSTEM

It is necessary to identify the similarities between the typing pattern of the typed password and the password samples in the training database to authenticate a legal client. In this proposed approach, the training database represents a set of verified unique passwords that are entered by the client during sign up for an authenticated system. The training database is used to measure the empirical threshold limits for all the five timing features of the password. In this study, a thresholding algorithm is presented to verify the validity of the client based on the threshold limits. First, the timing features are extracted from the typed password samples of a user during the sign up phase. The password training database for each client is constructed using the extracted timing features. Then, the empirical threshold upper and lower limits are calculated from the timing features in the database. Finally, the client validation is carried out using the threshold values.

A brief discussion on the training password database creation and threshold calculations are presented in the following sections.

Training Password Database

A net banking enrollment application has been considered for illustration purposes to understand the proposed concept clearly. Firstly, the clients in a bank who are interested in accessing the net banking application of the bank are requested to sign up at the bank or website. The legal client or user is a bank customer who has applied for internet banking services. The sign up process consists of two stages; (i) the username and password enrollment, (ii) timing feature extraction from the enrolled password sets. In the enrolment or sign up stage, the client has to create their username and password. The client will then confirm and re-type their unique password on the sign up page for 's' number of times to create a sample set. Timing features discussed in the previous section are extracted from the sample set and stored in the training database 'T'.

Let 's' be the number of password samples and 'n' be the number of characters in a password. The training database 'T' with timing features can be expressed as:

$$T = \left\{ H_{ij},\ L_{ik},\ PP_{ik},\ RR_{ik},\ PR_{ik} \right\} \qquad i=1,2,\ldots,s\ ; \qquad j=1,2,\ldots,n\ ; \qquad k=1,2,\ldots,n-1 \tag{1}$$

The training database can be rewritten in matrix form as,

$$T = \begin{bmatrix} H_{11},\ldots,H_{1n},L_{11},\ldots,L_{1n-1},PP_{11},\ldots,PP_{1n-1},RR_{11},\ldots,RR_{1n-1},PR_{11},\ldots,PR_{1n-1} \\ H_{21},\ldots,H_{2n},L_{21},\ldots,L_{2n-1},PP_{21},\ldots,PP_{2n-1},RR_{21},\ldots,RR_{2n-1},PR_{21},\ldots,PR_{2n-1} \\ \cdots \\ \cdots \\ \cdots \\ \cdots \\ H_{s1},\ldots,H_{sn},L_{s1},\ldots,L_{sn-1},PP_{s1},\ldots,PP_{sn-1},RR_{s1},\ldots,RR_{sn-1},PR_{s1},\ldots,PR_{sn-1} \end{bmatrix} \tag{2}$$

Where; H_{ij} is the hold time of a character, L_{ik} is the latency time between two successive characters, PP_{ik} is the time between depressing the first character and depressing the following character, RR_{ik} is the time between releasing the first character and releasing the following character, PR_{ik} is the time between depressing the first character and releasing the following character; i = 1 to

s, j = 1 to n & k = 1 to n-1. The training database 'T' has 5n-4 timing feature vector for each sample password.

For illustrative purposes, the password entered by the client is assumed as "hello", and thus the password length is n = 5. The structure of the training database with all five timing features of the password samples is shown in Figure 2.

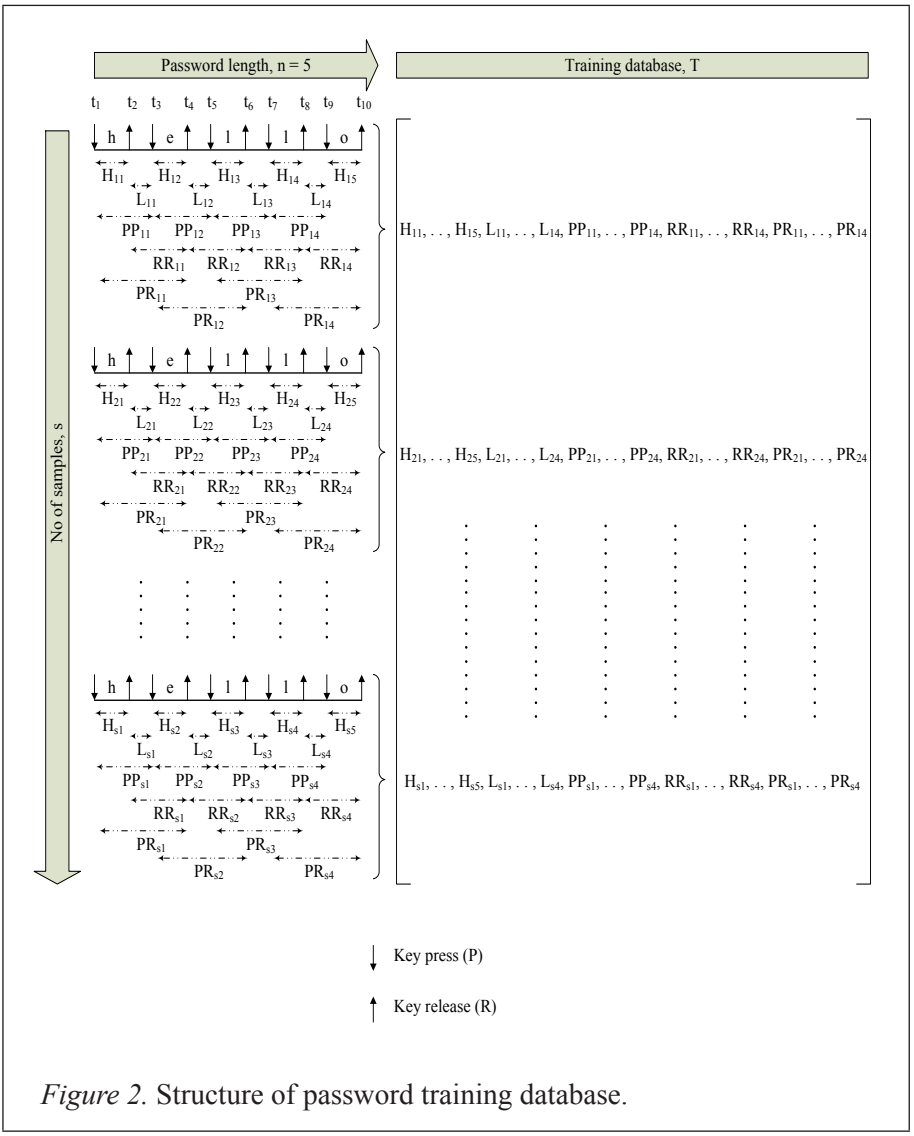

Figure 2. Structure of password training database.

Empirical Threshold Calculation

An appropriate validation method is required to recognize the resemblance between the typing rhythm of the correctly entered password and the password samples in the training database 'T'.

This study presents an empirical threshold based client validation method. The verification of the typing rhythm of a password is accomplished through the proposed thresholding algorithm. The thresholding algorithm uses the keystroke timing features of a user's training database 'T', and then calculates the empirical threshold limits for each timing features. The threshold minimum and maximum limits for each timing features are calculated from the training database 'T' and are expressed as follows:

$$Th_{min} = \left[min(H_{i1}), \ldots, min(H_{in}), min(L_{i1}), \ldots, min(L_{in-1}), min(PP_{i1}), \ldots, min(PP_{in-1}), min(RR_{i1}), \ldots, min(RR_{in-1}), min(PR_{i1}), \ldots, min(PR_{in-1}) \right] \quad i=1,2,\ldots,s \tag{3}$$

$$Th_{max} = \left[max(H_{i1}), \ldots, max(H_{in}), max(L_{i1}), \ldots, max(L_{in-1}), max(PP_{i1}), \ldots, max(PP_{in-1}), max(RR_{i1}), \ldots, max(RR_{in-1}), max(PR_{i1}), \ldots, max(PR_{in-1}) \right] \quad i=1,2,\ldots,s \tag{4}$$

Authentication Criteria

When the client accesses the secure login page using a username and password, the password typed by the client is first validated against the training database. If the typed password matches with that from the database, then the timing features for the correct password are calculated. The new timing feature vector 'P_r' can be expressed as,

$$P_r = \left[H_{r1}, \ldots, H_{rn}, L_{r1}, \ldots, L_{rn-1}, PP_{r1}, \ldots, PP_{rn-1}, RR_{r1}, \ldots, RR_{rn-1}, PR_{r1}, \ldots, PR_{rn-1} \right] \tag{5}$$

After the calculation of the timing features, a threshold based client authentication is conducted by cross referencing each timing feature to identify the values that lie between its minimum threshold Th_{min} and maximum threshold Th_{max} limits.

$$Th_{minr} \leq P_r \leq Th_{maxr} \qquad r=1,2,\ldots,n,n+1,\ldots,2n-1 \tag{6}$$

Assume 'A_L' to be the minimum acceptance level for the presented KDA method, i.e. the minimum number of timing features that lies between the upper and lower threshold limits to allow the client to access the secure

system during the authentication phase. Since the typing rhythm of a person will not be similar most of the time, a minimum A_L should be considered for authentication. Therefore, the client is authenticated to access the secure computer system or web application if A_L (or greater) value of the timing features lies between their minimum and maximum threshold limits. The value of A_L for the proposed authentication method is estimated in the comprehensive analysis section. The simple flow of the authentication process is as shown in Algorithm 1.

Algorithm 1: The proposed authentication process

Step 1: Set flag = 0

Step 2: For j = 1 to 2n-1

Step 3: If $Th_{\min j} \leq T_j \leq Th_{\max j}$

Step 4: flag = flag + 1

Step 5: Else

Step 6: flag = flag + 0

Step 7: End

Step 8: End

Step 9: If flag > A_L*(5n-4)/100

Step 10: (i) The client is legal and allowed to access the secure system or application

Step 11: (ii) Perform the database update (discussed in the following section)

Step 12: Else

Step 13: Illegal login and the permission is denied

Step 14: End

DATABASE UPDATE

In a password-based authentication system, the clients are required to create a username and password during the registration or sign up process, whereby they are requested to confirm their password for a minimum of two times.

By applying the proposed method in this study, the client is required to re-type their password continuously for 's' number of times to create the training database within a few minutes during the client sign up phase. It is estimated that the timing features of the newly typed password samples should resemble the timing features of previously typed password samples in the database. However, there may be small deviations in typing rhythms when the client types the same password after a few hours, days or weeks. Therefore, this poses complications to authenticate the legal client correctly after some duration of hours, days or weeks, if the A_L of the feature vector is set as 100%.

Typically, the database used in the static mode of KDA method is fixed (Killourhy & Maxion, 2009). When the KDA method uses the fixed database, the accuracy of the system is reduced due to the invariable and fixed database. As mentioned earlier, there are some differences in keystroke dynamics when the same user types their password after some duration of hours, days or weeks. Therefore, in order to improve the accuracy and security level of the proposed method, the database update is implemented for every successful validation of a legal client. The summary of the database update is described as; whenever a password typed by the client is in line with the authentication criteria mentioned in the previous section, the password database update rule is performed to enhance the accuracy of the proposed method. In the database update process, the first row in the training database 'T' is removed and a new timing feature vector 'P_r' obtained from the latest legal login is included into the last row of the database 'T'.

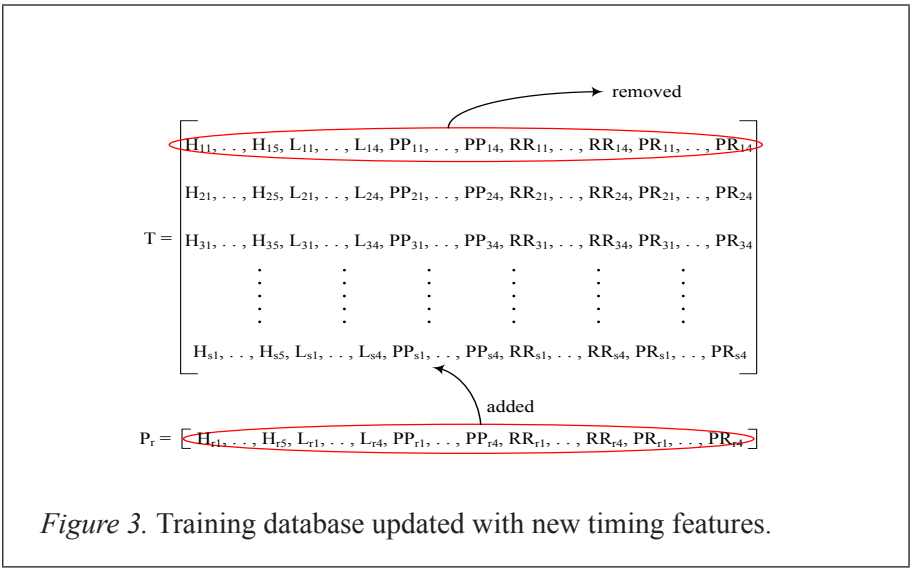

Figure 3. Training database updated with new timing features.

The proposed database update process can reduce the dissimilarities in timing features, allowing clients to access the login page after an extended period of the time (days or weeks). The process of updating the training database is depicted in Figure 3.

The primary objective of the database update process is to store the most recent timing features of the correctly authenticated password. This process allows the database to be updated with the latest timing features, while improving the accuracy of validation. The updated training database can be written as:

$$T = \begin{bmatrix} H_{21}, \ldots, H_{2n}, L_{21}, \ldots, L_{2n-1}, PP_{21}, \ldots, PP_{2n-1}, RR_{21}, \ldots, RR_{2n-1}, PR_{21}, \ldots, PR_{2n-1} \\ H_{31}, \ldots, H_{3n}, L_{31}, \ldots, L_{3n-1}, PP_{31}, \ldots, PP_{3n-1}, RR_{31}, \ldots, RR_{3n-1}, PR_{31}, \ldots, PR_{3n-1} \\ \cdots \\ \cdots \\ \cdots \\ H_{s1}, \ldots, H_{sn}, L_{s1}, \ldots, L_{sn-1}, PP_{s1}, \ldots, PP_{sn-1}, RR_{s1}, \ldots, RR_{sn-1}, PR_{s1}, \ldots, PR_{sn-1} \\ H_{r1}, \ldots, H_{rn}, L_{r1}, \ldots, L_{rn-1}, PP_{r1}, \ldots, PP_{rn-1}, RR_{r1}, \ldots, RR_{rn-1}, PR_{r1}, \ldots, PR_{rn-1} \end{bmatrix} \quad (7)$$

After the update of the training database using new timing features, the threshold limits are re-estimated using (3) and (4).

Algorithm 2: To create password training database and threshold limits

Step 1: Client has to create their username and password in signup page.
Step 2: Verify or retype the password for 's' number of times to create a sample set.
Step 3: Calculate all the five timing features for each password sample.
Step 4: Create a training database using measured timing features of each password sample.
Step 5: Estimate threshold minimum and maximum limits for every timing features using (3) and (4).
Step 6: The estimated threshold limits are further used for client authentication.

Algorithm 3: To validate a client to access the security system

Step 1. Password typed by the client during login is compared against the database. If the password matches with the database, then go to the next step, else the client is illegal, and the permission to access the secure system is denied.

(continued)

Algorithm 3: To validate a client to access the security system

Step 2: Estimate the five timing features time of the correctly typed password.

Step 3: Validate each timing features whether it lies between the minimum and maximum threshold limits. If minimum AL of the timing features lies between the threshold limits, then go to the next step, else the client is illegal, and the permission to access the secure system is denied.

Step 4: The client is legal and authenticated to access the secure system or web application.

Step 5: Perform the database update using (7).

Step 6: Estimate the threshold limits for the updated database using (3) and (4).

RESULTS AND DISCUSSION

Based on the results of the experimental analyses, the behavior of the proposed KDA method with respect to the client database size and acceptance level of authentication is examined in this section. Moreover, the quality of the authentication with different acceptance levels on timing features is discussed to understand the scalability of the proposed technique. Different acceptance level and different training sample sizes are explored to gain an understanding on the overall performance of the proposed KDA method. The acceptance level ranges from 10% to 100% and the training sample size 's' is between 3 to 30. Each client's database consists of 's' number of password samples with '5n-4' timing features. The experimental analysis presented in this section provides significant insight on the behavior of the proposed KDA method in practical security applications.

Experimental Setup

The proposed authentication system was developed and validated on different people using MATLAB simulation software in Windows 10 laptop with i7 CPU 1.8GHz, 4GB RAM. The timing features are calculated using five distinct stopwatch timers. Each stopwatch has a dedicated interrupt and timer. The illustration of calculating the hold time and latency time are as follows:

(i) When a person presses a key, the interrupt in stopwatch 1detects the state of the key press, and turns "ON" timer 1. After a period of time, if the person releases the key, the interrupt 1 senses the state of key release, and it turns "OFF" timer 1. The time interval between timer 1 "ON" and "OFF" is estimated to be the hold time of a character.

(ii) Similarly, the latency time between the two successive characters is calculated using stopwatch 2. When the user releases the first key, the interrupt 2 detects the state of key release, and it turns "ON" timer 2. If the client presses the next key, the interrupt 2 identifies the state of the key press, and it turns "OFF" timer 2. The time interval between timer 2 "ON" and "OFF" is estimated to be the latency time between two successive characters. Consecutively, the remaining three timings are computed using three separate stopwatches by conducting the hold and latency time calculations.

Performance Indices

Like other biometric-based methodologies, the application of the proposed KDA method is assessed through different indices (Clavel, Ehrette, & Richard, 2005; Kim et al., 2018; Leggett et al., 1991).

False Acceptance Rate: The rate at which an impostor is authenticated as a valid client (Clavel et al., 2005). A higher FAR suggests that the illegal person is frequently authenticated, and that the security of the application is minimal.

False Rejection Rate: The rate at which a valid client is denied access to the secure app or page (Clavel et al., 2005). A higher FRR suggests that the legal client is frequently denied.

Equal Error Rate: As FAR rate decreases, FRR rate increases, and vice versa. The intersection point of FAR and FRR is defined as EER (Kim et al., 2018). For any authentication systems, both FAR and FRR rates are the same at EER.

Overall accuracy: The percentage of authentication accuracy is defined as the number of participants that are validated correctly, out of 100 participants (combination of both legal clients and impostors). The simple formula to compute the overall accuracy using FAR and FRR are as follows:

$$\% \quad Accuracy = 100 - (FRR + FAR) / 2 \tag{8}$$

Comprehensive Analysis

The proposed KDA method examines 100 legal clients to validate the accuracy and robustness of the security system. Each legal clients are allowed to log in to the secure page or app for 100 times at different time intervals to calculate

FRR. To calculate FAR, 100 different impostors were permitted to access the secure system by typing the same password.

Table 3 to 6 depicts the experimental results of the proposed security framework in client authentication with different sample sizes and acceptance levels. Figure 4a and 4b show the FRR and FAR comparison analysis of the proposed KDA method for user authentication with different sample sizes and acceptance levels. Evidently, the accuracy of the proposed security framework improved as the size of the samples in the database increased.

Table 3

Results of client authentication with s = 3

s = 3	Acceptance level									
	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
FRR	0	0	0	0	3	8	9	28	41	61
FAR	100	91	87	66	53	28	9	1	0	0

Table 4

Results of client authentication with s = 10

s = 10	Acceptance level									
	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
FRR	0	0	0	0	1	4	5	25	36	56
FAR	100	83	75	58	48	21	6	0	0	0

Table 5

Results of client authentication with s = 20

s = 20	Acceptance level									
	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
FRR	0	0	0	0	0	2	3	22	31	51
FAR	100	79	68	53	36	19	3	0	0	0

When a client accesses the secure network, the password typed by the client is authenticated against the samples in their password database, i.e. all five timing features in the typed password are verified against the corresponding

threshold limits. These threshold limits have been estimated from the client sample database typed at different intervals during the sign up phase.

Table 6

Results of client authentication with s = 30

s = 30	Acceptance level									
	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
FRR	0	0	0	0	0	1	1	8	21	37
FAR	100	72	58	41	22	8	0	0	0	0

Based on the results from Table 3 to 6, the FRR and FAR values decreases as the size of the samples in the client database increases. Therefore, as the size of the client sample increases, the more timing data of the typed password matches that from the client's database. Comparison plots were drawn to highlight this relationship for both FRR and FAR with different sample sizes and acceptance levels. Figure 4a and 4b shows the comparison of FRR and FAR respectively for different sample sizes and acceptable levels.

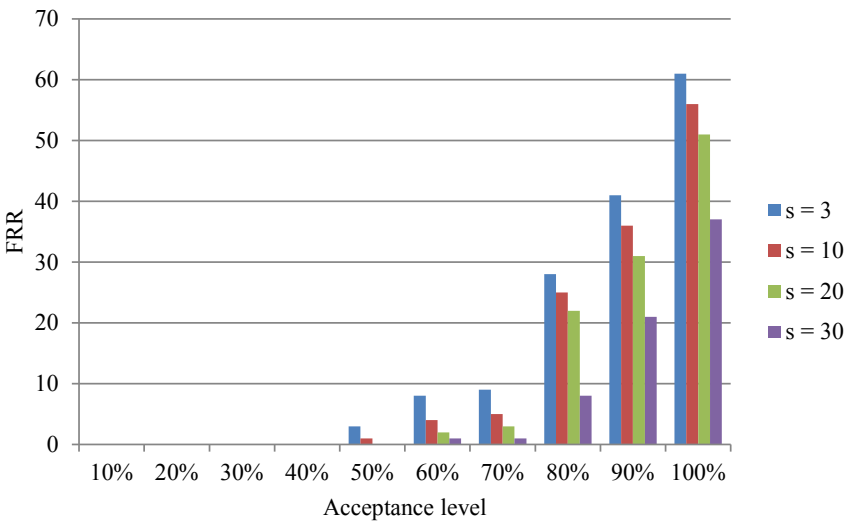

Figure 4a. FRR with different sample sizes and acceptance levels.

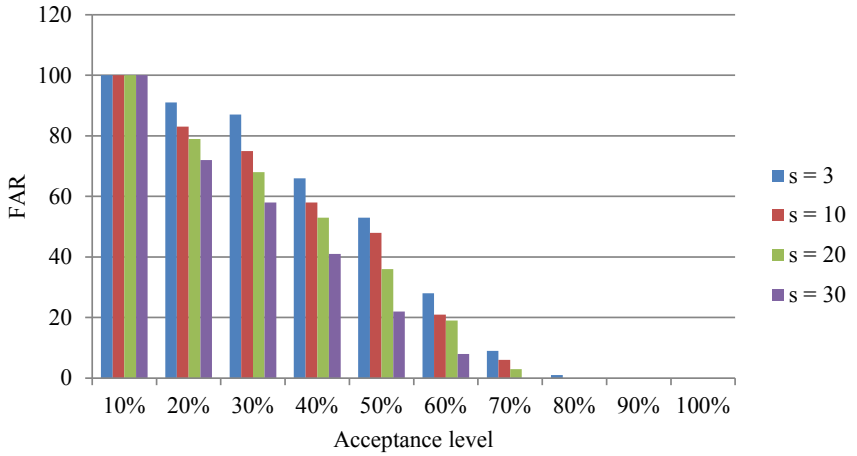

Figure 4b. FAR with different sample sizes and acceptance levels.

Table 7

Accuracy improvement in FRR (%)

Change in sample size, Δs	Acceptance level									
	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
3 to 10	0	0	0	0	2	4	4	3	5	5
3 to 20	0	0	0	0	3	6	6	6	10	10
3 to 30	0	0	0	0	3	7	8	20	20	24

Table 8

Accuracy improvement in FAR (%)

Change in sample size, Δs	Acceptance level									
	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
3 to 10	0	8	12	8	5	7	3	1	0	0
3 to 20	0	12	19	13	17	9	6	1	0	0
3 to 30	0	19	29	25	31	20	9	1	0	0

Table 7 and 8 show the percentages of improvement in FRR and FAR with respect to changes in database size. Each row in Table 7 depicts the percentage of improvement in FRR when the sample size varies from 3 to 10, 3 to 20 and 3 to 30. From Table 7, the FRR value decreases significantly when the acceptance level is set above 60%. Table 8 depicts the percentages of improvement in FAR when the sample size changes from 3 to 10, 3 to 20 and 3 to 30. For all sample sizes ranging from 3 to 30, FAR is zero if the acceptance level is above 90%, in contrast to the FRR value that remains high at the same level of acceptance. Hence, to ensure a more secure authentication system with a lower number of training samples, an acceptance level above 90% is most applicable. However, this level produces a high FRR value that would reduce the accuracy of the authentication system. Therefore, to distinguish the optimal acceptance level and sample size, FRR-FAR equilibrium graphs were drawn for all the four sample sizes s = 3, 10, 20 and 30 and are depicted in Figure 5a, 5b, 5c, and 5d respectively. In all cases, FAR value decreases while FRR value increases, when the level of acceptance increases. EER lies around 70% of acceptance level. The rate of EER is high when s = 3 and decreases significantly when s = 30. Hence, for practical implementation, the optimal acceptance level and sample size can be chosen as 70% and 30 respectively to achieve maximum accuracy.

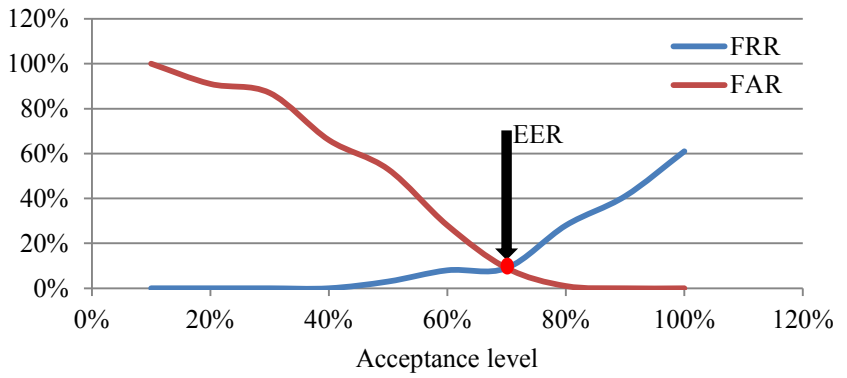

Figure 5a. FAR and FRR equilibrium at s=3.

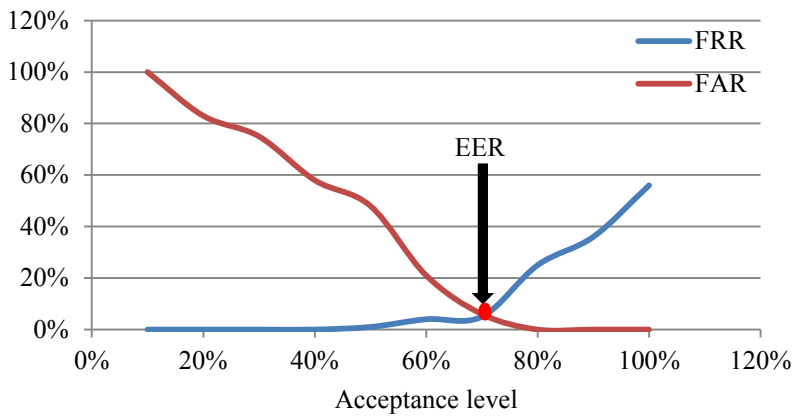

Figure 5b. FAR and FRR equilibrium at s=10.

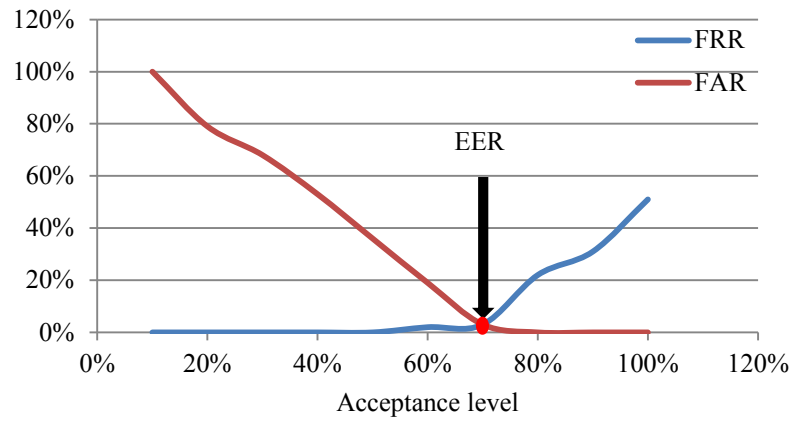

Figure 5c. FAR and FRR equilibrium at s=20.

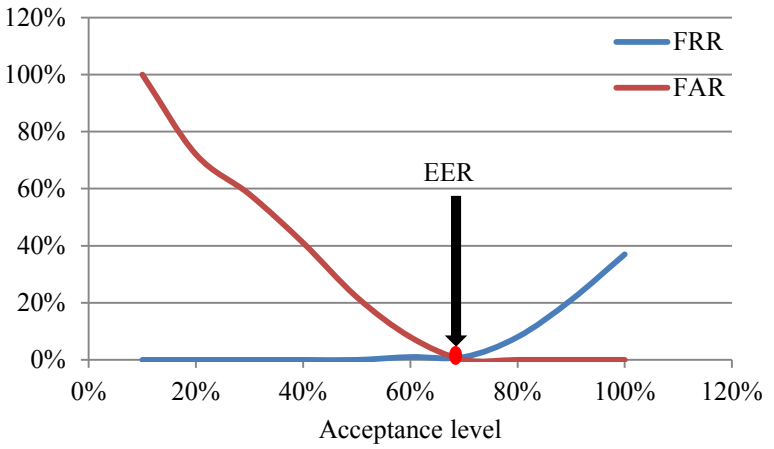

Figure 5d. FAR and FRR equilibrium at s=30.

From Table 6, at 70% acceptance level, the FRR is 1 and FAR is 0. From Table 7, there is a significant improvement in FRR if both the acceptance level of authentication and database size increases. Similarly, there is a significant improvement in FAR observed in Table 8, if the acceptance level is set between 40% and 70%. Based on the inferences made from the data in Table 6 to 8, the best acceptance level, A_L for the presented KDA method is 70% followed by a training sample size, s at 30. Therefore, the value of FAR being at 0 for A_L = 70% and s = 30 does indicate that no impostor can access the secure network, and subsequently, increases the security of the authentication system. At the same acceptance level of 70% on different sample sizes of 3, 10 and 20; FAR is 9%, 6%, and 3% respectively. These results suggest that the application of the KDA method with smaller sample sizes could allow impostors to access the network, and in turn decreases the security of the authentication system. Thus, the proposed KDA method that utilizes the sample size s = 30 and acceptance level, A_L of 70% is the most secure biometric authentication technique with the minimal FRR of 1%. The minimum false rejection rate implies a very low possibility of denying a legal user the access into a secure system. Although a legal client has been rejected as an impostor on the first or second attempt, the proposed parameters for the authentication system has a higher probability to authenticate the legal client successfully in their successive attempts.

Comparative Study

A comparative study has been carried out for the proposed and existing methods to compare the superiority, proficiency, accuracy and security levels

of the proposed KDA method. In this study, the proposed KDA method was investigated for 100 legal clients and 100 impostors, with the acceptance level, A_L of 70% and databases consisting of 30 training password samples. In (Leggett et al., 1991), an identity verification method based on dynamic keystroke properties was implemented. From the study, it was concluded that the method had FRR of 11.1% and FAR of 12.8% for dynamic mode; FRR of 5.5% and FAR of 5.0% for static mode. 36 individuals were requested to enter 2 typing samples. The first sample consisted of 1400 characters treated as reference data, while a second sample of 300 characters were used as test data. Each person typed the same text samples at two different time periods, separated by several days. A KDA technique had been introduced in (Furnell et al., 1996), and verified both static and dynamic modes. In static mode,15 participants were used to collect text samples. Each participant were allowed to submit 35 typed samples. A total of 30 participants were analyzed in dynamic mode, whereby each participant typed two reference samples and two more samples for testing. The reference sample text consisted of 2200 characters. The experimental results depicted that both static and dynamic modes had an overall accuracy rate of 92.5% with FRR of 0% and FAR of 15% for dynamic mode; FRR of 7% and FAR of 8% for static mode.

An authentication method based on free text had been presented in (Gunetti & Picardi, 2005), which utilized the distance measures of keystroke digraphs and was examined on 205 different users. For experimental analyses, 40 participants were employed to collect 15 typed samples each. The 40 participants were treated as valid clients for a secure network. Furthermore, another set of 165 participants were treated as impostors, and each were requested to type a single text sample. Each sample of free text had 800 characters for both valid users and impostors. The test results showed that this method produced FRR of less than 5% and FAR of less than 0.005% (Gunetti & Picardi, 2005).

A KDA system using pairwise user coupling and machine learning algorithms have been presented in (Mondal & Bours, 2017) and the technique was used to analyzed both free text and fixed text inputs. Various case studies were conducted in online exam based KDA database, and the method had achieved an overall accuracy level of 89.7% from a sample size of 500. A neural network based method combined with monograph and digraph for free text analysis of keystrokes was presented in (Ahmed & Traore, 2014). This method yielded FAR of 0.0152%, FRR of 4.82%, and EER of 2.46% in heterogeneous conditions; whereas in homogeneous condition, it produced FAR of 0%, FRR of 5.01%, and EER of 2.13%. Since the EER was more than 2% in both homogeneous and heterogeneous conditions, this method is unsuitable for real-time implementation.

A user-adaptive feature extraction based KDA method that utilized free text input was proposed to improve client authentication (Kim et al., 2018). The experiment was conducted on 150 participants with 13,000 keystrokes per user.

Even for a larger sample size of 1000, the user adaptive feature extraction based authentication method produced EER of 0.44%. Table 9 depicts the comparative studies of the proposed KDA and the existing methods. From Table 9, it is observed that the proposed KDA method has the lowest FRR (1%) and FAR (0%) values compared to the other existing methods (Furnell et al., 1996; Gunetti & Picardi, 2005; Kim et al., 2018; Leggett et al., 1991; Mondal & Bours, 2017). These rates are optimal for real-time implementation. The index FRR corresponds to the accuracy level of authenticating a legal user, which in turn represents the security level of the system. Therefore, the proposed KDA method with the low FRR and FAR values suggests that the security system is highly accurate and relatively more secure.

Table 9

Comparative studies of the proposed and existing methods

Method	Sample size	Character length	FRR (%)	FAR (%)	Accuracy (%)	The speed of authentication (sec)
Dynamic mode (Leggett et al., 1991)	2	1400	11.1	12.8	88.1	-
Static mode (Leggett et al., 1991)	2	-	5.5	5	94.75	55
Dynamic mode (Furnell et al., 1996)	2	2200	0	15	92.5	-
Static mode (Furnell et al., 1996)	35	-	7	8	92.5	49
(Gunetti & Picardi, 2005)	15	800	5	0.01	97.5	33
Heterogeneous (Ahmed & Traore, 2014)	1500	5500	4.82	0.015	97.6	-
Homogeneous (Ahmed & Traore, 2014)	1500	5500	5.01	0	97.5	-
(Mondal & Bours, 2017)	500	-	-	-	89.7	-
(Kim et al., 2018)	1000	1300	-	-	99.5	-
Proposed KDA	30	5	1	0	99.5	7

The accuracy of the proposed KDA method is computed using (8) and shows a 99.5% accuracy, similar to the method presented in (Kim et al., 2018). However, the proposed KDA method uses the training database size of 30, whereas the method in (Kim et al., 2018) utilizes a larger database size of 1000. This further supports the applicability of the proposed KDA method as it consumes lesser memory to store the training database compared to other KDA methods. Since the dynamic or larger training database method consumes more data size to store the lengthy free text content in the system, the validation process would take more time to run the searches and verify the user due to the massive database. The proposed KDA method resolves this problem and authenticates the client much quicker than the other existing dynamic methods. The speed of authentication is defined as the time interval between the instant at which a legal user enters the password in the login page and the instant at which the user is allowed to access the secure network after the successful validation of the user, using the KDA method. To verify the authentication speed of the proposed KDA method, user authentication is performed as many as 10 times to measure the average time taken to authenticate a user. There are no preexisting data available that relates to the authentication speed of the previous methods in other works of literature (Furnell et al., 1996; Gunetti & Picardi, 2005; Kim et al., 2018; Leggett et al., 1991; Mondal & Bours, 2017). Therefore, the same number of sample size s = 30 for text content "hello" has been used to estimate the authentication speed for both proposed and existing methods in static mode. The average speed of authentication of a legal client in the proposed KDA and other existing methods were calculated and presented in Table 9. The proposed KDA method was able authenticate a person much faster than the other methods.

CONCLUSION

This paper studied the application of a novel KDA method based on empirical threshold values that was developed to authenticate a legal client who accesses a secure web page or application. Various analyses were carried out in different environmental conditions, and the results suggested that the proposed KDA method with the aid of empirical thresholding algorithm and database update rule has established a better security level compared to other existing techniques. The maximum password length used in the proposed KDA method was limited to ensure less memory was consumed when data was stored in the training database. Due to the lower memory utilization of the system, the proposed empirical thresholding algorithm and authentication criteria was able authenticate a legal user much faster than the other existing KDA methods. The proposed method was carried out on 100 legal clients

and 100 unauthorized persons, and the experimental results showed that the method performed well in authenticating and differentiating the legal clients and impostors. The results also revealed that the proposed KDA method has a sharply diminished FRR and FAR values. The proposed KDA method has a low EER of 0.5%, with authentication accuracy of 99.5%. These values support the proposed KDA method to be suitable and efficient for real-time implementation. KDA techniques that uses a fixed database causes the accuracy of the system to decrease due to the invariable and fixed database. The empirical thresholding algorithm and database update rule presented in the proposed method significantly improved robustness, accuracy, and efficiency of client authentication. Thus, the proposed keystroke dynamics can be employed to validate an individual's identity in various applications, and has been proven to enhance the security of the network. The proposed feature extraction and threshold techniques can be employed in both static and dynamic modes of authentication.

ACKNOWLEDGEMENT

This research received no specific grant from any funding agency in the public, commercial, or not-for profit sectors.

REFERENCES

Ahmed, A. A. E. (2009). Employee surveillance based on free text detection of keystroke dynamics. *Handbook of Research on Social and Organizational Liabilities in Information Security*, 47–63.

Ahmed, A. A., & Traore, I. (2014). Biometric recognition based on free-text keystroke dynamics. *IEEE Transactions on Cybernetics*, *44*(4), 458–472. doi:10.1109/TCYB.2013.2257745

Al Nuaimi, Z. N. A. M., & Abdullah, R. (2017). Neural network training using hybrid particle move artificial bee colony algorithm for pattern classification. *Journal of Information and Communication Technology*, *16*(2), 314–334.

Albashish, D., Sahran, S., Abdullah, A., Alweshah, M., & Adam, A. (2018). A hierarchical classifier for multiclass prostate histopathology image gleason grading. *Journal of Information and Communication Technology*, *17*(2), 323–346.

Boopathi, M., & Aramudhan, M. (2017). Dual-stage biometrics-based password authentication scheme using smart cards. *Cybernetics and Systems*, *48*(5), 415–435.

Chen, W., & Chang, W. (2004). Applying hidden Markov models to keystroke pattern analysis for password verification. In *Information Reuse and Integration, 2004. IRI 2004. Proceedings of the 2004 IEEE International Conference on* (pp. 467–474). IEEE.

Cho, T.-H. (2006). Pattern classification methods for keystroke analysis. In *SICE-ICASE, 2006. International Joint Conference* (pp. 3812–3815). IEEE.

Clavel, C., Ehrette, T., & Richard, G. (2005). Events detection for an audio-based surveillance system. In *2005 IEEE International Conference on Multimedia and Expo* (pp. 1306–1309). doi:10.1109/ICME.2005.1521669

Das, R. K., Mukhopadhyay, S., & Bhattacharya, P. (2014). User authentication based on keystroke dynamics. *IETE Journal of Research*, *60*(3), 229–239.

Furnell, S. M., Morrissey, J. P., Sanders, P. W., & Stockel, C. T. (1996). Applications of keystroke analysis for improved login security and continuous user authentication. In *Information systems security* (pp. 283–294). Springer.

Gunetti, D., & Picardi, C. (2005). Keystroke analysis of free text. *ACM Transactions on Information and System Security (TISSEC)*, *8*(3), 312–347.

Hemanidhi, A., & Chimmanee, S. (2017). Military-based cyber risk assessment framework for supporting cyber warfare in Thailand. *Journal of Information and Communication Technology*, *16*(2), 192–222.

Hosseinzadeh, D., Krishnan, S., & Khademi, A. (2006). Keystroke identification based on gaussian mixture models. In *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings* (Vol. 3, pp. III–III). doi:10.1109/ICASSP.2006.1660861

Hussien, H. M., Muda, Z., & Yasin, S. M. (2018). New key expansion function of Rijndael 128-bit resistance to the related-key attacks. *Journal of Information and Communication Technology*, *17*(3), 409–434.

Joshi, S. S., & Phoha, V. V. (2007). Competition between SOM clusters to model user authentication system in computer networks. In *Communication Systems Software and Middleware, 2007. COMSWARE 2007. 2nd International Conference on* (pp. 1–8). IEEE.

Kaewwit, C., Lursinsap, C., & Sophatsathit, P. (2017). High accuracy EEG biometrics identification using ICA and AR model. *Journal of Information and Communication Technology*, *16*(2), 354–373.

Kang, P., & Cho, S. (2015). Keystroke dynamics-based user authentication using long and free text strings from various input devices. *Information Sciences*, *308*, 72–93. doi:10.1016/j.ins.2014.08.070

Killourhy, K. S., & Maxion, R. A. (2009). Comparing anomaly-detection algorithms for keystroke dynamics. In *Dependable Systems & Networks, 2009. DSN'09. IEEE/IFIP international conference on* (pp. 125–134). IEEE.

Kim, J., Kim, H., & Kang, P. (2018). Keystroke dynamics-based user authentication using freely typed text based on user-adaptive feature extraction and novelty detection. *Applied Soft Computing*, *62*, 1077–1087.

Kotani, K., & Horii, K. (2005). Evaluation on a keystroke authentication system by keying force incorporated with temporal characteristics of keystroke dynamics. *Behaviour & Information Technology*, *24*(4), 289–302.

Leggett, J., & Williams, G. (1988). Verifying identity via keystroke characterstics. *International Journal of Man-Machine Studies*, *28*(1), 67–76.

Leggett, J., Williams, G., Usnick, M., & Longnecker, M. (1991). Dynamic identity verification via keystroke characteristics. *International Journal of Man-Machine Studies*, *35*(6), 859–870. doi:10.1016/S0020-7373(05)80165-8

Maazouzi, Z. E., Mohajir, B. E. E., & Achhab, M. A. (2017). A systematic reading in statistical translation: From the statistical machine translation to the neural translation models. *Journal of Information and Communication Technology*, (2), 34.

Memon, Q. A. (2017). Neural network-based double encryption for jpeg2000 images. *Journal of Information and Communication Technology*, *16*(1), 137–155.

Mihajlov, M., Jerman-Blažič, B., & Ciunova Shuleska, A. (2016). Why that picture? Discovering password properties in recognition-based graphical authentication. *International Journal of Human–Computer Interaction*, *32*(12), 975–988.

Mohamed, R., Zainudin, M. N. S., Sulaiman, M. N., Perumal, T., & Mustapha, N. (2018). Multi-label classification for physical activity recognition from various accelerometer sensor positions. *Journal of Information and Communication Technology*, *18*(2), 209–231.

Mohsin, M. F. M., Bakar, A. A., Hamdan, A. R., & Abdul, M. H. (2018). An improved artificial dendrite cell algorithm for abnormal signal detection. *Journal of Information and Communication Technology*, *17*(1), 33–54.

Mondal, S., & Bours, P. (2017). Person identification by keystroke dynamics using pairwise user coupling. *IEEE Transactions on Information Forensics and Security*, *12*(6), 1319–1329. doi:10.1109/TIFS.2017.2658539

Odei-Lartey, E. O., Boateng, D., Danso, S., Kwarteng, A., Abokyi, L., Amenga-Etego, S., … Owusu-Agyei, S. (2016). The application of a biometric identification technique for linking community and hospital data in rural Ghana. *Global Health Action*, *9*, 29854.

Robinson, J. A., Liang, V., Chambers, J. M., & MacKenzie, C. L. (1998). Computer user verification using login string keystroke dynamics. *IEEE Transactions on Systems, Man, and Cybernetics-Part a: Systems and Humans*, *28*(2), 236–241.

Shehab, M., Khader, A. T., & Laouchedi, M. (2018). A hybrid method based on cuckoo search algorithm for global optimization problems. *Journal of Information and Communication Technology*, *17*(3), 469–491.

Syed, Z., Banerjee, S., & Cukic, B. (2016). Normalizing variations in feature vector structure in keystroke dynamics authentication systems. *Software Quality Journal*, *24*(1), 137–157. doi:10.1007/s11219-014-9263-1

Van Zoonen, L., & Turner, G. (2014). Exercising identity: Agency and narrative in identity management. *Kybernetes*, *43*(6), 935–946. doi:10.1108/K-06-2013-0126

Zahari, M. K. M., & Zaaba, Z. F. (2017). Intelligent Responsive Indoor System (IRIS): A potential shoplifter security alert system. *Journal of Information and Communication Technology*, *16*(2), 262–282.