ISSN: 2454-9940

INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT

E-Mail :
editor.ijasem@gmail.com
editor@ijasem.org

www.ijasem.org

INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT

ISSN 2454-9940

www.ijasem.org

Vol 19, Issue 2, 2025

A Novel Approach to User Authentication During Password Change: Static Keystroke Dynamic Authentication (SKDA)

[1] Mr.Khaja Pasha Shaik,
[2] SHAIK MOHIUDDIN, [3] MOHAMMED ABDUL GAFFAR, [4] ABDUL RAHMAN,
[1] Assistant Professor, Department of AIML, Lords Institute of Engineering & Technology
[234] Student, Department of AIML, Lords Institute of Engineering & Technology.

Abstract—

For authentication purposes, keystroke dynamics is seen as a supplementary component. The user is recognized in static keystroke dynamics, in particular, by using the time feature, which is recorded when the user inputs their login ID and password. It is necessary to include temporal characteristics into the user profile in order to do this. The availability of keyboard timing data is low during a password change, for example, therefore unconventional features might be useful in this case. This article delves into the topic of identifying users when they change their passwords using non-traditional features like NumLock, Shift, CapsLock, etc. In addition, the article explains how to construct a model for password changing based on the non-traditional characteristics of static keystroke dynamics.

Index Terms—Keystroke, Non-conventional, Authentication, Static, Password

INTRODUCTION

The timing features need recording the exact moment each key is pressed and released for the specified characters. Because of this, it is necessary to capture the password many times before it can be utilized as a support factor for interpreting the user's timings. Continuous keystroke dynamics record data on key presses such Caps Lock, Shift, and Backspace to enhance decision performance. Non-Conventional features are those that do not record time. Our previous study showed that experimentally combining non-conventional characteristics with timing features enhanced system performance in static keystroke dynamics and made user authentication more effective. When it comes to authenticating users for access to personal computers, online services, the internet, etc., passwords are by far the most popular choice. Due to the many weaknesses of password-based authentication, multi-factor authentication is often used to enhance security [1]. Which leads to dissatisfaction among users. Support factor authentication is suggested as a solution to this problem. Keystroke dynamics, which is dependent on user behavioral traits—specifically, typing behavior—is one component that helps with authentication [2] [3]. Therefore, keystroke analysis is used to confirm the user's identity with the password's validity. Each user is said to have their own distinct behavioral biometric method. Static and continuous keystroke dynamics are the two main categories. While continuous keystroke dynamics records keystrokes continually while the user is using their personal computer, static keystroke dynamics only records the user's typing rhythm at login time. Since the system cannot continually collect keystrokes, the static keystroke dynamics approach is used for identity management systems that only verify the user's validity during the login procedure. Much study has focused on traditional features—timing characteristics used to simulate static keyboard dynamics—as a means of user authentication. Making advantage of It is necessary to repeat the training cycle in order to construct a new model for the user every time they update their password in static keystroke dynamics, as this model is also constructed using training of timing characteristics. This implies that until the system is trained, we will not have the ability to make advantage of this supplementary component. During the password-changing process, this research suggests using non-traditional elements in static keystroke dynamics. Since a model based on the user's non-conventional attributes will be accessible throughout the training cycle, it will aid in user prediction. In light of this potential, the authors of this work provide a model that takes into account non-traditional aspects to aid in the process of changing a user's password up until the model is generated using timing features for the new password. Here is the structure of the paper: The paper's literature review and studies on keystroke dynamics, non-traditional characteristics, and the ex isting dataset are presented in Section II. In Section 3, we learn about the Keystroke Dynamics System and all its capabilities. We also get into the case when the password has to be changed and the difficulties that come with relying only on the timing characteristics.

ISSN 2454-9940
www.ijasem.org
Vol 19, Issue 2, 2025

Section IV provides information regarding the non-conventional features of keystroke dynamics, including how to capture them, challenges in using them in a scenario where the password needs to be changed, the need for policies, descriptions of the policies used to frame the passwords and text, and test cases for static keystroke dynamics. In Section V, we detail the suggested method for user authentication during password changes by combining static keystroke dynamics with non-conventional and temporal aspects. We also provide results from the experiments that validated the model. Section VI delves into the research's experimental findings. The article is concluded with closing observations in Section VII.

LITERATURE SURVEY

For authentication and verification reasons, keystroke dynamics employ feature extraction to collect the user's typing characteristics [4] [5]. In addition to the text itself, the keyboard also defines the precise time of each key press and release. These are recorded as characteristics of the timing of keystrokes. The characteristics are sent back into the classification system from the basic data collected while typing [6]. Every researcher uses their own unique set of user authentication characteristics and classification methods [7]. [8]. Any two-character sequence that contains spaces, digits, letters, and punctuation is called a digraph. Digraphs may not be able to tell who is using them just by looking at them. Combining digraphs with additional characteristics, such tri-graphs and n-graphs, serves varied uses, such as determining the typing mistake rate. What follows is a discussion of a few such instances. Using trigraphs and digraphs, D. R. Gentner et al. [9] were able to detect the sorts of user typing mistakes. To improve the typing experience, Roth et al. [10] added digraph components to the sound of keystrokes. One of its basic tenets is that, when pushed, each key will produce a slightly different sound, according to the individual user. Their goal was to master a virtual alphabet by assembling groups of similar keystrokes. The digraph latencies inside the virtual letter pairs were then used to compute the score. Digraphs, trigraphs, and n-graphs are keyboard properties that are context dependent only [11]. In order to get beyond these dependencies Diagraphs, tri-graphs, and n-graphs were used by Dowland and Furnell [12] in conjunction with the following keyword latencies: AutoID, Left character, Right character, Latency, and Timestamp. When digraph latencies were used, they got the best results. Section A. Unusual Characteristics Common characteristics of keystroke dynamics include dwell time (the length of time a key is held down, indicating a typing style) and latency time (the distance between keystrokes, indicating typing speed) [4] [13]. These characteristics are referred to as timing features because of their relationship to time. Although the data about the keys used in a keystroke has not been tested, it has been noted that timing elements are heavily used in static keystroke dynamics. In order to verify a user's identity, it could be helpful to see how they often use certain keys, such as Shift, Caps Lock, the Number Key, or the Left or Right Shift Key. In continuous keystroke dynamics, they are referred to as non-conventional characteristics. In order to authenticate the user using machine learning methods, keystroke dynamics must include user identification and categorization. Training data from the user profile is required in order to understand the user's typing pattern. The data is verified for accuracy after the categorization process is finished [14]. A. Data Set The majority of researchers have created and used their dataset for testing and training purposes, as well as to compare with other algorithms that are comparable [15] [16]. Except for data sets that specifically address timing aspects like KeyPress, KeyRelease, Hold time, and Flight/latency time, no other data sets are currently available. In order to authenticate users and handle password changes, this study aims to use non-traditional aspects in static keystroke dynamics. There is no dataset available for non-standard characteristics. These include things like using the Shift key (left or right) or the Number key (top or right section of the keyboard). Furthermore, the research found that compared to the continuous keystroke dynamics technique, the static keystroke dynamics approach uses a lot fewer non-conventional characteristics [16]. Therefore, in order to conduct the study, we had to construct our dataset. Our goal was to include non-traditional aspects in the static keystroke dynamic technique during authentication and when the user wishes to update their password. As there is little data available for the changed password, we suggest using non-conventional features in this work to address the change of password problem. What follows is an analysis of the policy that was developed with this use case in mind.

KEYSTROKE DYNAMICS SYSTEM AND FEATURES

INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT
ISSN 2454-9940
www.ijasem.org
Vol 19, Issue 2, 2025

It is called keystroke dynamics, and it involves studying a sequence of key occurrences and the time between them. It is essential to separate KeyPress and KeyRelease in order to record keystroke timing characteristics. Additional temporal characteristics are extracted from these two collected features. The delay between the times of KeyPress and KeyRe lease is known as the hold time [17]. The time that elapses between two successive KeyReleases is known as the UP-Up time [18]. The time it takes for a key to be released and then pressed again is called the up-down time [18]. Time between two consecutive key presses is known as "down-down" [17]. • Time to descend A key press's down-up time is the amount of time that elapses before the next key release [17]. The first step in developing a user profile for a keystroke dynamics system is to record the user's keystroke characteristics. In order to build a profile for each user, the keystroke dynamic system examines their typing habits. Various timing aspects are derived from the user's typing rhythm. During the authentication step, these attributes might be used as a reference in a training set. The user's typing speed in the provided input sample will be compared to the training set's reference template during the authentication/testing step. Features that rely on timing may help verify a user's identity from their profile. Until the user profile is established, the approach cannot be utilized. Scenario A: Password Change Password verification makes use of the method outlined in the previous section. In contrast, if the user changes their password, the model that was trained using an old password may not function anymore and a new model would have to be created. Only by tracking the user's typing habits for certain words can the learned timing characteristics be of value. Password and login input is dependent on the model created for that particular password and does not allow for the acquisition of extra typing information due to static keystroke dynamics. A use case where a time-based model is useless is the changing of password; in this work, we explore the use of non-conventional characteristics in static dynamics. Therefore, the case of changing the password is taken into account and spoken about. Since it's conceivable for two users to have the same typing speed, timing characteristics cannot be relied upon to confirm the benign user. Problems with using timing characteristics (B) Two users enter the password "India@2018" with almost identical timing distances in various time aspects, as shown in Figure 1. However, their patterns of pressing and releasing the Shift key and the Numbers key vary. Therefore, it could be useful to look for non-standard patterns of feature utilization together with timing characteristics that might identify the user. Figure 2 shows that two people's use of unconventional characteristics might be very different from one another. The recording of the non-standard typing style is therefore anticipated to be more useful in differentiating people, particularly in cases when the time information is unavailable, such as when a person changes their password. As a result, we investigate the difficulty of using non-traditional characteristics for user identification.

NON-CONVENTIONAL FEATURES

Among the non-standard aspects of keystroke dynamics is the ability to record the pattern of pressing Shift, Caps Lock, Num, and Number keys. Since the system has plenty of time and chances to catch the user's behavior while using the keys, non-conventional characteristics are simpler to capture in continuous keystroke dynamics. But it's not easy to capture non-conventional aspects in static keystroke dynamics, because the system only knows the user's typing information when they log in. Previous work of ours explored the potential of static keystroke dynamics that make use of non-traditional characteristics.

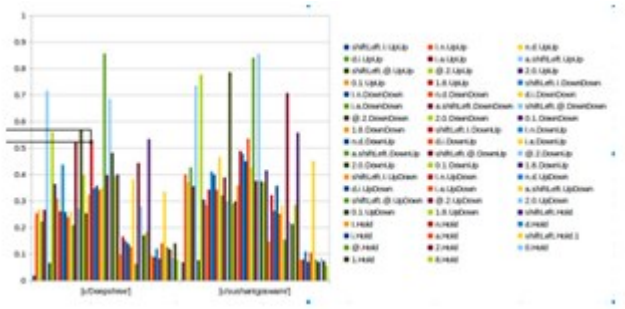

Fig. 1. Conventional features timing plot of two people inputting the identical phrase at various times

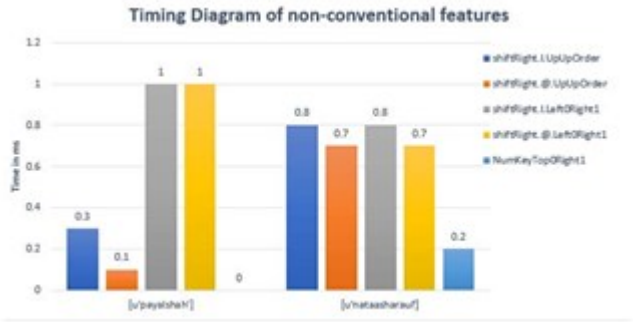

Fig. 2. Non-conventional features timing plot of two people

at different times entering the same sentence. in order to better identify the individual, which was tested on a limited group of users [19]. Our study makes use of non-traditional characteristics, which are described in detail ahead of time for reference and to facilitate their application to the Change Password use case. A. Elements of Unusual Characteristics When typing capital letters or odd symbols like punctuation marks, everyone utilizes the keyboard in their own unique way, particularly when using the (left/right) shift key. Pressing CapsLock or the Left Shift Key and Right Shift Key simultaneously allows the user to type in all capital letters. Users also have separate links to utilize the Left ShiftKey and the Right ShiftKey when entering punctuation marks. It has come to our attention that users often exhibit repetitive patterns while utilizing these particular keys. These keys are regarded non-conventional features, thus it's important to record the user's behavior in certain situations in order to understand how they utilize them. How often a user presses each shift key may be monitored using this function. • Keyboard Shortcuts: This feature keeps track of the user's keyboard shortcuts for entering capital letters [20]. • The (Top/Right) Numeric Key: The majority of keyboards include two numeric keypads, one on the right side and one above the lettering. Users show a preference for the number key sets when entering numerical data. The top panel's number keys are used more often by those who are used to using them. The right panel users are no different. Very few people are seen to be able to utilize them easily. One of the non-traditional aspects to be collected for typing is the user's pattern of keystrokes, which might be revealed via their use of number keys. It's recommended to release the first key before the second: Here we have yet another intriguing routine. As said before, when the user uses the non-conventional factor—ShiftKey—to press two keys simultaneously, they must release them in a certain order or all at once. The user's key-release pattern is the main emphasis of this functionality. This isn't a timing feature, albeit the release time is taken into account. As seen in Figure 3, this becomes important when the user presses the Shift key. For example, this occurs when a user pushes Shift followed by another key, holds down Shift, and then lets off of the other key [20]. • Deliberately releasing both keys simultaneously As seen in Figure 4, this becomes important when the user presses the Shift key. This occurs, for example, when a user pushes the Shift key, then presses and releases another key, then finally releases the Shift key [20].

A total of 30 individuals were analyzed by recording their typing speed while they typed "India@2018" 15 times. As shown in Figure 1, it has been verified that two users entering the same password could have distinct non-traditional factor-based input patterns. Section B: Policy Framework Requirement Section II explains that timing parameters for user authentication are the primary focus of static keyboard dynamics research. People have a distinct way of pressing keys on a keyboard, as mentioned in [19]. If you know how to utilize shortcut keys like Shift, Caps Lock, or a certain sequence of numbers, you can unlock even more features. Because they are associated with the kind of key press and release rather than time collected, these aspects are referred to as non-conventional [19]. In static keystroke dynamics, there are restrictions on the length of the username and password, making it difficult to catch the usage of these non-conventional elements. In contrast, continuous keystroke dynamics make it easy to record such features. We have developed regulations and declarations to collect additional data about the user's unconventional feature use. The user is presented with these assertions and the password at random. This allows for the collection of extra data on the user's usage of non-traditional features. Part C: Guidelines for Encrypting Data and Passwords The minimum length of a password is eight characters, and it must include both capital and lowercase letters, digits, and special characters (! @,#,$,%,&, etc.) for security purposes. However, we have included an extra stipulation that the length must not exceed "10" characters for this experiment. The five rules listed below were developed to ensure that the intended password setting process captured any non-standard characteristics. • Rule No. 1: Any combination of capital and lowercase letters: hitting the Shift key, together with the CapsLock key, will produce uppercase characters, whereas hitting the Shift key alone will produce special case characters. The goal of this policy is to determine whether the user prefers to use CapsLock or the Shift key to type capital letters. You should also specify which of the two Shift keys—the left or right one—the user usually uses. Because of this policy, it is easier to record when a user presses the Shift or CapsLock keys. Policy 2: A combination of capital letters: There are two options for users who encounter strings of continuous capital letters: either utilize a CAPS lock to input just uppercase letters or keep the shift key held while typing. Habits often dictate the decision. The policy is therefore centered on the user's usage of CapsLock. Policy 3 emphasizes the usage of the Shift/Caps key by alternating between lowercase and uppercase letters.

INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT

ISSN 2454-9940

www.ijasem.org

Vol 19, Issue 2, 2025

TABLE I TEST CASES FOR FRAMING THE TEXT USING THE POLICY

Test Cases	Examples	Users showing the same pattern behaviour
Case 1: Keeping the uppercase character and the special case character together (Any order)	@W, N@, H@, &S@, @P	86%
Case 2: Several uppercase characters together	CAST, SERVICES, MADE, BEN	86%
Case 3: Alternate upper and lower case characters	MaDe, bAnKiNg	92%
Case 4: Consecutive Numbers	4357, 009	93%
Case 5: Half upper-case Half lower-case	BENchmark	93%

someone using it. For some people, the only keys they need to write capital letters are shift (left/right). However, there are those who would rather use the capital letters while typing. Num Lock, often known as the top vertical number panel of the keyboard, is made explicit in Policy 4-Consecutive Numbers. Users' actions when entering numerical values via the Num Lock key or the top panel may be recorded. • Policy 5: Lowercase and Half Half lowercase: This setting highlights the user's usage of the Ctrl key. To type all capital letters, some users may choose to utilize just the capital letters key. Although Shift (left/right) keys may be more convenient for certain people while typing capital letters. In order to capture the different combinations of non-conventional qualities that users would use to build their passwords, the following three texts were crafted using the principles indicated above. Section D. Developed Test Use Cases Fifty people were survey to get a better idea of how they write by having them type the same three phrases five times in a row. Table I summarizes examples and percentages of users that type the same way when choosing a password, regardless of the content written. While typing, it was noticed that users' behavior with keys like Shift (left/right), CapsLock (top/right side of the keyboard), and Number (bottom/side of the keyboard) is quite consistent. Table I displays the outcomes. This proves that the regulations are clear and may assist in detecting unusual characteristics when users input passwords. Table II displays the assessment findings together with the performance matrix derived from the data obtained from the users while they typed the text. The findings show that changing the user's password does not affect their unconventional behavior with regard to the Shift/Caps keys. No matter whether it's a password, a statement, or just plain old typing, people have a habit of using the same shift (left/right) and numbers (top/right) keys. The suggested Static Keystroke Dynamics Authentication (SKDA) Model was therefore developed with the results' observations in mind.

PROPOSED STATIC KEYSTROKE DYNAMICS AUTHENTICATION(SKDA) MODEL

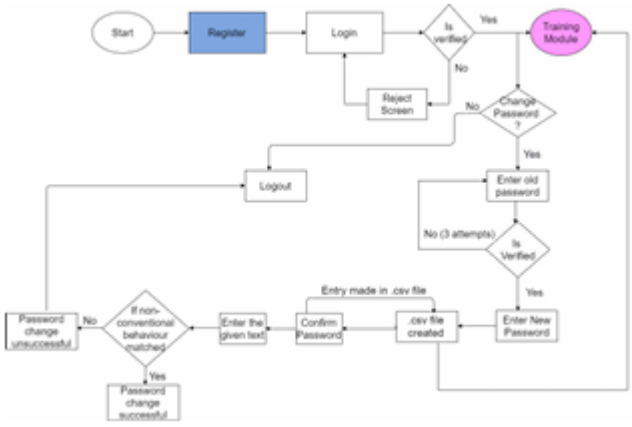

Fig. 5. SKDA Model to authenticate the user

The SKDA Model records the amount of time each user spends typing and the keys they utilize. From the recorded timing characteristics, we derive timing features like hold time and flight/latency time for each user. Once non-standard elements like the Number Key's use (top or right area of the keyboard), Shift Key's use (left or right side), and CapsLock Key's use have been recorded, a unique pattern may be derived from the collected data. Because the necessary timing features are unavailable when the user changes the password, it is impossible to remark on the user's authenticity based on keyboard use (as specified in Section III-B). In the interim between the development of the timing-based model and its completion, the non-conventional elements suggested in Section IV may prove to be invaluable. Figure 5 shows the proposed SKDA Model, which uses non-conventional characteristics to authenticate the user when they update their password. The suggested strategy requires the user to log in using the system's default password. When the user registers, we record the exact moment they press and release each key. After signing up, users are only need to log in once. If their credentials are same, training will get data on

INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT

ISSN 2454-9940

www.ijasem.org

Vol 19, Issue 2, 2025

how quickly they typed their password. If the authentication process is unsuccessful, the user will be declined and requested to log in once again. A. Case for Changing Passwords As mentioned before, the user's non-standard keystroke profile has already been generated. The policy outlined in Section IV-C will now be used to produce randomly selected text, which the user will be requested to input when entering a new password. Once the user inputs the text, the authentication process may be completed by comparing the new password with the non-conventional characteristics. The training phase will now receive this freshly generated file including the new password in order to capture its temporal aspects. Module B: Data Collection The model requires three separate inputs of the username and password (which may be changed once training is complete) before it can work properly. Data pertaining to the user's keystrokes while entering a password is recorded. We accept these three submissions as genuine. It will affect the whole dataset if the initial three entries aren't close to optimal. The consistency check must be finished before any entries following the third entry are saved. The k-nearest neighbor technique is used to verify for consistency and prevent outliers from reaching the data. In consistency check, a training sample is only deemed real if the average distance value is smaller than the closest kth distance from the future training dataset. The information is sent to the training module after the fifth input (see Section V-C). C. Instructional Guide The data files of each user are fed into the training model. Figure 7 shows the training process, which involves adding n/4 false entries to n data entries for each user using the user data entries collected during registration and login after the consistency check is completed, as detailed in Section V-B. Division D: Experimenting Python, the Django framework, and the web technologies of Javascript, jQuery, and Bootstrap were used to implement the SKDA model that was presented. Nobody could log in except the individual who used the

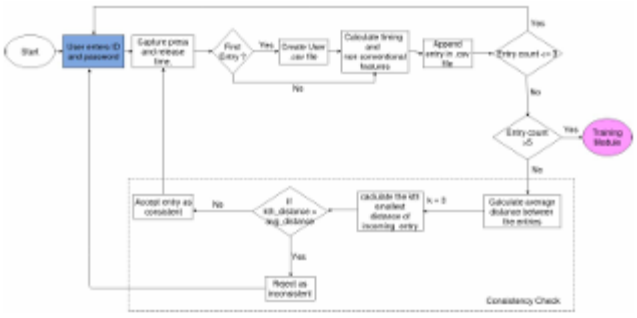

Fig. 6. Data Collection Module

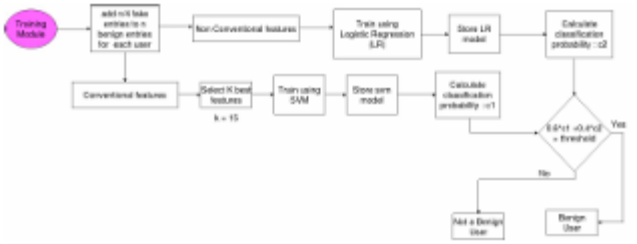

Fig. 7. Training Module

Figures 9 and 10 further show that when the user employs non-standard features, the typing pattern changes. The results of testing the suggested model on 155 users, who were observed inputting the default password and changing it, are detailed in.

RESULTS

Using the performance indicators False Rejection Rate (FRR) and False Acceptance Rate (FAR), the suggested SKDA model was assessed. The primary goal of this study was to determine the effect of using non-traditional aspects of Static Keystroke Dynamics, as described in Section IV, on the detection of real and false users during credential checks during login and when the

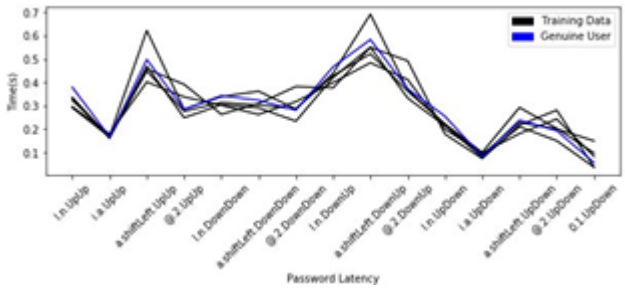

Fig. 9. Login Attempt of Genuine user variations along with its trained dataset of 15 contributing features

INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT

ISSN 2454-9940

www.ijasem.org

Vol 19, Issue 2, 2025

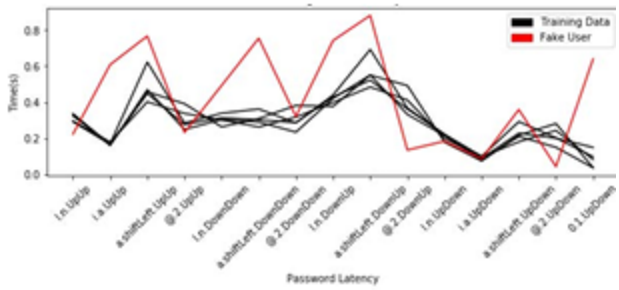

Fig. 10. The user wishes to update their login credential. There are false user variants and a real user training dataset showing login attempts. The results of evaluating the suggested model on both the default and changed password typing rhythms of the user are shown in

Table III. TABLE III DEFAULT AND CHANGEDPASSWORDNON-CONVENTIONAL FEATURES PERFORMANCE METRICS

Performance Parameters	Default Password Non-conventional features	Change Password Non-conventional features
FAR	0.22	0.21
FRR	0.05	0.04
ERR	0.13	0.12

Table IV shows that when compared to previous research, our findings reveal that the SKDA model uses a relatively small number of characters (10–15 characters). Time and non-traditional aspects were both taken into account by our SKDA model on a short string of text (ten to fifteen characters), which is often thought of as the length of a password (i.e., static keystroke dynamics). Achieving FAR of around 0.02% and FRR of roughly 0.3% was a success, as indicated in Table IV.

CONCLUSION AND FUTURE SCOPE

In static keystroke dynamics, the model that is created from the temporal characteristics acquired when the user enters their login and password is used for authentication. Nevertheless, information pertaining to key presses and releases is not recorded when passwords are changed.

Study	No. of Partici-pants	No. of Char-acters used	Features	FAR	FRR
Arwa Alsul-tan et al. [21]	30	1000	Non-conventional features both	0.0011	0.28
Kathryn Hempstalk et al. [22]	10	10800-30000	Non-conventional features	0.113	0.331
Blaine Ayotte et al. [23]	-	-	Timing Fea-tures	0.029	0.490
Ahmed A. et al. [24]	53	11000	Timing fea-tures	0.0152	4.82
Saira Zahid et al. [25]	25	12500	Timing fea-tures	0.292	0.308
A. Alsultan et al. [20]	25	7200	Timing and Non-conventional features both	0.009	0.215
SKDA Model	155	10-15	Timing and Non-conventional features both	0.02	0.3

a new password that you may use. The user's new password is then mapped to the available model for static keystroke dynamics. This study proposes and demonstrates the usage of non-conventional characteristics for password changing. The article goes on to talk about rules that may be used to record unconventional aspects like shift key use, NumLock key usage, CapsLock key usage, and key release pattern when using shift keys. For user authentication, the SKDA model employs non-traditional elements in conjunction with timing features. An experiment was conducted to test the idea with 155 individuals. The results showed a false alarm rate (FAR) of around 0.02% and a false positive rate (FRR) of about 0.3%. In the future, we want to investigate other kinds of keyboards and determine the device-specific keystroke dynamics.

REFERENCES

[1]. R. Joyce and G. Gupta, “Identity authentication based on keystroke latencies,” in Communications of the ACM, 33(2), 168-176, 1990.

[2]. S. Mondal and P. Bours, “Continuous authentication in a real world settings,” in Advances in Pattern Recognition, Eighth International Conference, pp. 1-6, 2015.

[3]. F. Monrose and A. D. Rubin, “Keystroke dynamics as a biometric for authentication,” in Future Generation computer systems, 16(4), 351-359, Elsevier Science, 2000.

[4]. R. Abinaya and A. Sigappi, “Biometric identification of a genuine user/imposter from keystroke dynamics dataset,” in

International Jour nal of ChemTech Research, Vol.11 No.08, pp 147-160, 2018.

[5]. A. Andrey, Vyazigin, Y. Nadezhda, Tupikina, and V. E. Sypin, "Software tool for determining of the keystroke dynamics parameters of personal computer user," in International Conference on Micro/Nanotechnologies and electron devices EDM,978-1-7281-1753-9/19/$31.00, IEEE, 2019.

[6]. P. H. Pisani and A. C. Lorena, "A systematic review on keystroke dynamics," in Journal of the Brazilian Computer Society, 2013.

[7]. M. L. Ali, J. V. Monaco, C. C. Tappert, and M. Qiu, "Keystroke bio metric systems for user authentication," in Journal of Signal Processing Systems,DOI: 10.1007/s11265-016-1114-9, Springer, 2016.

[8]. K. Shekhawat and D. P. Bhatt, "Recent advances and applications of keystroke dynamics," in International Conference on Computa tional Intelligence and Knowledge Economy (ICCIKE),978-1-7281 3778-0/19/$31.00, IEEE, 2019.

[9]. D. R. Gentner et al., "A glossary of terms including a classification of typing errors," in in Cognitive aspects of skilled typewriting, ed: Springer, pp. 39-43, 1983.

[10]. J. Roth, X. Liu, A. Ross, and D. Metaxas, "Investigating the discrimi native power of keystroke sound," in IEEE Transactions on Information Forensics and Security, vol. 10, pp. 333-345, 2015.

[11]. Y. Zhong and Y. Deng, "A survey on keystroke dynamics biometrics:

[17]. P. S. Teh, A. B. J. Teoh, C. Tee, and T. S. Ong, "A multiple layer fusion approach on keystroke dynamics," in Pattern Analysis and Applications, 14:23–36, DOI 10.1007/s10044-009-0167-9, 2011. [18] A. A. Ahmed and I. Traore, "Biometric recognition based on free-text keystroke dynamics," in IEEE TRANSACTIONS ON CYBERNETICS, VOL. 44, NO. 4, APRIL, 2014.

[18]. R. Nataasha, R. Shankarmani, and P. Joshi, "Non-conventional fac tors for keystroke dynamics as a support factor for authenticating user," in International Journal of Innovative Technology and Ex ploring Engineering (IJITEE), ISSN: 2278-3075, Volume-9 Issue-4, DOI:10.35940/ijitee.D1194.029420, 2020.

approaches, advances, and evaluations," in Recent Advances in User Authentication Using Keystroke Dynamics Biometrics. Science Gate Publishing, pp. 1-22, 2015.

[12]. P. S. Dowland and S. M. Fumell, "A long-term trial of keystroke profiling using digraph, trigraph and keyword latencies," in Security and Protection in Information Processing Systems, ed: Springer, pp. 275-289, 2004.

[13]. P. Lozhnikov, E. Buraya, A. Sulavko, and A. Eremenko, "Methods of generating key sequences based on keystroke dynamics," in Dynam ics of Systems, Mechanisms and Machines (Dynamics) IEEE, DOI: 10.1109/Dynamics.2016.7819038, 2016.

[14]. A. Alsultan and K. Warwick, "User-friendly free-text keystroke dynamics authentication for practical applications," in Interna tional Conference on Systems, Man, and Cybernetics (IEEE), DOI: 10.1109/SMC.2013.793, 2013.

[15]. K. S. Killourhy and R. A. Maxion, "Comparing anomaly-detection algorithms for keystroke dynamics," in Proc. IEEE/IFIP International Conference on Dependable Systems Networks (DSN), 125–134, 2009.

[16]. R. Nataasha, R. Shankarmani, and P. Joshi, "A comprehensive review of keystroke dynamics-based authentication mechanism," in Proceedings of Advances in Intelligent Systems and Computing, vol 1059. Springer, DOI: https://doi.org/10.1007/978-981-15-0324-5 13, 2019.

[19]. A. Alsultan, K. Warwick, and H. Wei, "Improving the performance of free-text keystroke dynamics authentication by fusion," in Applied Soft Computing, 2017.

[20]. [21] A. Arwa, W. Kevin, and W. Hong, "Non-conventional keystroke dynam ics for user authentication," in Pattern Recognition Letters, vol. 89, pp. 53-59, 2017.

[21]. [22] I. H. W. K. Hempstalk, E. Frank, "One-class classification by combining density and class probability estimation," in The European Conference on Machine and Learning and Principles and Practice of Knowledge Discovery in Database, 505–519, 2005.