International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:15, No:1, 2021

A Second Look at Gesture-Based Passwords: Usability and Vulnerability to Shoulder-Surfing Attacks

Lakshmidevi Sreeramareddy, Komalpreet Kaur, Nane Pothier

***Abstract*—**For security purposes, it is important to detect passwords entered by unauthorized users. With traditional alphanumeric passwords, if the content of a password is acquired and correctly entered by an intruder, it is impossible to differentiate the password entered by the intruder from those entered by the authorized user because the password entries contain precisely the same character set. However, no two entries for the gesture-based passwords, even those entered by the person who created the password, will be identical. There are always variations between entries, such as the shape and length of each stroke, the location of each stroke, and the speed of drawing. It is possible that passwords entered by the unauthorized user contain higher levels of variations when compared with those entered by the authorized user (the creator). The difference in the levels of variations may provide cues to detect unauthorized entries. To test this hypothesis, we designed an empirical study, collected and analyzed the data with the help of machine-learning algorithms. The results of the study are significant.

***Keywords*—**Authentication, gesture-based passwords, machine learning algorithms, shoulder-surfing attacks, usability.

I. Introduction

FOR many computer applications, text-passwords are the method of authentication. It is well known that these passwords are vulnerable to attacks by third–parties. Users tend to choose easy to remember passwords [1], which are in turn easier for attackers to obtain with fewer trails. A graphical password plays the same role as a text-password but can include handwritten drawings or designs, with or without textual elements.

A graphical password is accessible to speakers of any language because it is alphabet independent. Users do not have to remember a string of letters. Password authentication requires humans to recall and recognize the passwords immediately or over short intervals of time. Paivio and Yuille [2] demonstrated experimentally that the users' recall ability after 72 hours of retention interval–drops by 50% for alphanumeric passwords, but only 20% for pictures.

Effective user authentication applications are crucial in maintaining information security. In response to the growing number of threats to information security, a wide variety of authentication mechanisms have been developed [3]. Many existing authentication methods [4] (e.g., alphanumeric passwords) with random characters of the string (e.g., %&10#78dgl*) passwords are difficult to remember, especially for individuals with cognitive disabilities that affect memory (e.g., Down syndrome, mild traumatic brain injury, and dementia).

The gesture-based password approach aims to reduce the memory load of the authentication process. This approach uses only the mouse/touchscreen for input and requires no additional device. For their password, users draw an image using the mouse. In addition to the image drawn, this method considers measures such as confidence score, password length and size, pauses between strokes, stroke start and end angles, and speed to authenticate the user.

The design of the present study extends that of the previous study [5], which included a 20-second timer to mimic the shoulder-surfing effect. We suspect 20 seconds is too long to reveal the underlying impact of time. In addition, the image of the passwords was presented when the user tried to replicate the password, which is not possible in reality for a shoulder surfer. In this second study, we are re-examining the issue by presenting the password image for shorter amounts of time (5, 10, 15 seconds) and removing the image before the participants start replicating the passwords. In addition, we wanted to see if there is any difference in password behavioral features based on type of password such as simple (for example, a circle) versus complex (for example, a password drawn using native language). Taking into account the passwords content from [5], we divided the password categories into two groups. We asked participants to choose a password from a group. More information about this process will be discussed in the application design and results section of this paper.

In this paper, we report a user study aiming to evaluate the usability of the proposed gesture-based authentication method. We also collected preliminary data from different timers that helps can contribute to an understanding how the proposed method function in the case of shoulder surfing attacks.

We developed a gesture-based password application for data collection. Two modes of data collection were used: Creation mode and Replication mode. In creation mode (Session 1), users were asked to create six different passwords and reenter each password five times. In replication mode, users saw a password image created by some other user for a fixed duration of time. Three different duration timers, 5

L. Sreeramareddy is with the Computer Science Department, Salem State University, Salem, MA 01970 USA (phone: 978-542-2379; fax: e-mail: lsreeramareddy@salemstate.edu).

K. Kaur and N. Pothier are with the Computer Science Department, Salem State University, Salem, MA 01970 USA (e-mail: kkaur@salemstate.edu, nane.pothier@gmail.com).

seconds (Session 2), 10 seconds (Session 3), and 15 seconds (Session 4) were used to mimic the shoulder-surfing attack. After the timer expired, the password image was removed, and users were asked to replicate the password. There were 74, 57, 50, and 44 users who participated in Session 1, Session 2, Session 3, and Session 4, respectively.

In this study, machine learning algorithms have been applied to determine whether the person is a genuine user or an imposter based on the password entered. Five different machine learning algorithms were deployed to compare the performance in user authentication: Decision Trees, Linear Discriminant Analysis [5], Naïve Bayes Classifier [5], Support Vector Machines (SVMs) with Gaussian Radial Basis Kernel function [6] , and K-Nearest Neighbor [7].

II. Related Research

Authentication methods using text-based passwords prompt users to choose complex strings or even random strings for better security [8]. These complex passwords are hard for people to remember. Therefore, these passwords become inaccessible for people with limited memory recall ability. Many researchers have predicted that drawing or pictures are retained in the memory longer than text because of the 'Picture Effect' and the principle that "One picture is better than a thousand words" [9]. Thus, gesture-based passwords are one potential alternative for people with cognitive disabilities. Gesture-based passwords may be recognition or recall-based. Recognition-based passwords require a remembered sequence of clicks. One example of a recall-based password is the DAS (Draw A Secret) scheme [11], which requires users to cross certain cells in a grid space. Another alternative is our stroke-based approach that allows users to draw freely on the canvas.

The most common user authentication method entails alphanumeric usernames and passwords. This has significant drawbacks; for example, the user tends to pick passwords that can be easy to guess. On the other hand, if a password is hard to guess, then it is often hard to remember. To address this problem, researchers have developed an authentication method that uses pictures. People can remember images better [9]. This authentication method is called graphical/gesture-based passwords.

There are many limitations of Alphanumeric passwords such as password space and difficulty remembering passwords when passwords include longer length of the characters [10]. Graphical passwords have been proposed in the recent decades. Some benefits of graphical passwords include their ease of use. Graphical passwords tend to have smaller password space, so the chances of occurring brute force attacks are high. By drawing nature of the gesture-based passwords on Canvas at the time of creation and login, results in shoulder-surfing attacks [10].

There is a Gesture-based password method that is closely related to the proposed: DAS scheme [11]. In this method, a password is drawn in a gird which then translates to sequence of (x, y) coordinates that passes through the gird [10]. The password space of this method is theoretically proven to be larger. However, as is the case with text-based passwords, users tend to underutilize the password space made available. In addition, 29% of the passwords created were invalid because they followed the grid lines or cross-grid corners criteria [12].

Two gesture-based password prototypes have been developed that take one or multiple strokes as passwords. The Passdoodles method [13] considers the shape of the stroke and the speed of movement. The gesture-based touchpad system [14] considers the shape of the strokes and the length of pause between strokes. Neither system has been formally evaluated via empirical user studies.

More recently, researchers have explored the possibility of using behavioral measures to enhance the authentication method. For example, [15] used pressure, coordinates, size, speed, and time as additional layers to authenticate users on a touch-screen smartphone. Sae-Bae et al. [16] examined fingertip dynamics (e.g., all fingers moving or partial fingers moving) as an additional authentication component and tested the idea using an iPad. Both studies suggest that the use of behavioral factors is promising [17]. However, both studies required pre-defined shapes or, gestures, or strokes created in relation to specific locations on the screen.

III. Application Design

In this study, we developed a gesture-based application to allow multiple strokes in the password. Each password can contain four strokes. The number of strokes allowed in this study was determined by the limits of web applications' performance. We developed four versions of the application for multi-stroke application. This study is of multi-stroke password application with four study sessions starting from study objectives, participants, tasks, and results. We used MATLAB for analyzing the data. We report this study's ML (Machine Learning) results using MATLAB classifiers to examine the shoulder-surfing effect of gesture-based password for the four sessions.

A. Definition of Password Groups

From our previous study [18], we know that the majority of the password content varied among the following categories:

1) Mathematical Shapes
2) Numbers
3) Letters
4) Letters or characters in other languages
5) Animals and other related living things
6) Trees, flowers and other plants
7) Everyday objects (books, forks, spoons)
8) Vegetables, fruits and other food
9) Human faces and emotions and human body parts
10) Buildings
11) Scenery
12) Vehicles and transportation

From the above categories, we have formed two groups to start with. Each group has its own categories. The groups of passwords content include:

1) Group 1: A password from the below categories

a. Mathematical Shapes
b. Numbers
c. Letters
d. Letters or characters in other languages
2) Group 2: A password from the following categories
a. Animals and other related living things
b. Trees, flowers and other plants
c. Everyday objects (books, forks, spoons)
d. Vegetables, fruits and other food
e. Human faces and emotions and human body parts

B. Webpage for Password Creation

The first application allows users to create passwords with multiple strokes. The main goal of this application is to collect the data and measure the behavioral features. Fig. 1 shows the user interface of this application. With respect to the algorithm, we adopted the $N [19] Multistroke Recognizer to compute the confidence score and also implemented algorithms to compute the behavioral features of the gesture-based password to support multi-strokes within a password. All other logging features were computed by modifying the related algorithms.

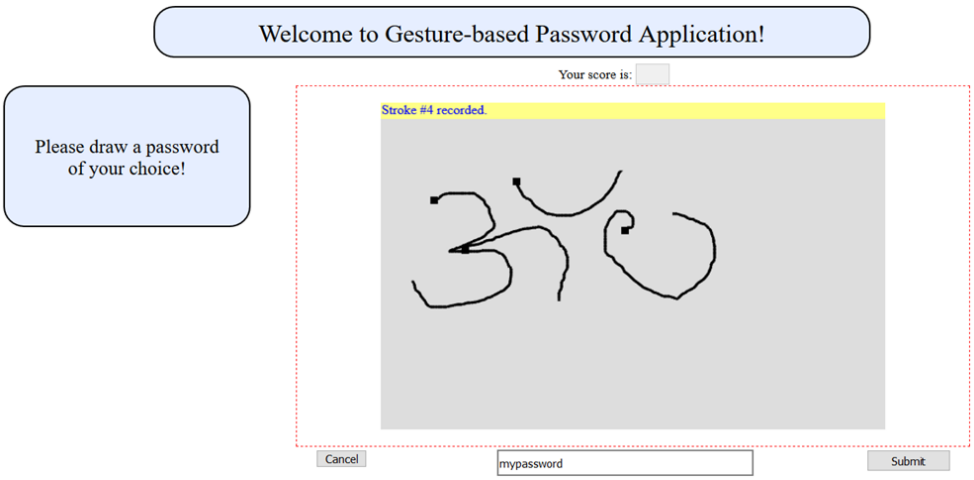

Fig. 1 User Interface for password creation session

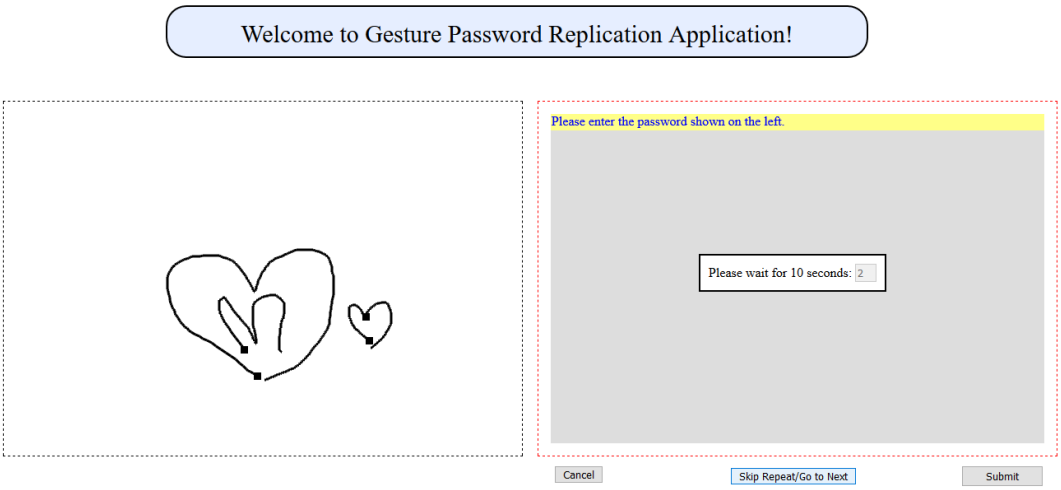

Fig. 2 User Interface for password replication session

C. The Web Page for Password Replication

The second application goal was to present participants with the password that had been created by other participants. The passwords selected to present to the participants were randomly picked from a pool of passwords generated by an algorithm that was implemented for this purpose. We modified the user interface for this application in order to present the password image for replication purposes. Since the users were replicating passwords that were created already, the researcher had to compute the confidence score of the replicated password after the session was completed. Because the template matching $N recognizer requires stroke points in an array of points, we were not able to compute the score at the run time. The main goal for this application was to study the

security of the gesture-based password, meaning how well the users could replicate the passwords that were created by other users. This session was a password replication with 5-second timer. The password image was presented to users for 5 seconds. The users memorized the password presented and replicated it five times. The second goal for this application was to study the memorability/recallability of the passwords, meaning how well users remembered the password that was presented for 5 seconds rather than having the password on the screen until they replicated it. Fig. 2 shows the user interface for this session.

The third and fourth websites are identical to the second website except the timer in which password image was presented to the users were 10 seconds and 15 seconds, respectively.

IV. Methodology

A. Participants

225 participants took part in the study. There were 74, 57, 50, and 44 users who participated in Session 1, Session 2, Session 3, and Session 4, respectively. 95 participants were female and 130 were male. The average age of the participants was 20.55 (Stdev. = 2.5). On average, participants had 12.55 years of computer experience (stdev. = 4.0) and 11.72 years of internet experience (stdev. = 3.8). Participants spent approximately 34.20 hours per week on the Internet (stdev. = 32.33).

B. Tasks

Within-subject design was adopted, and each participant completed four sessions. In the first session (the creation session), the participants created six gesture passwords and re-entered each password five times using the mouse. Among the six, the first two passwords were selected by the researchers from group 1 and group 2. The remaining four were the participants' own choice.

In the second, third and fourth sessions, participants were presented with images of six passwords created by other people and re-entered each password 10 times. The images were presented one at a time.

In all sessions except the first, the password image was presented on the left side of the screen for the duration of the timers–5, 10 and 15 seconds. After the timer expired, the image was removed. Participants were then instructed to begin the re-entry process. In all three time conditions, the image was presented on the screen for timer-specific seconds only (replication with 5, 10, and 15 second timers). The order of the three conditions was counter-balanced to control for the learning effect.

C. Data Processing

The study was conducted online and all participants completed the four sessions either at home or on campus. Participants were required to complete the four sessions on four different days within one week. Each session took approximately 20-40 minutes. At the end of the study, participants completed an online demographic questionnaire and a satisfaction survey. Data collected from all four sessions were organized into three sets: Dataset A, Dataset B, and Dataset C. Dataset A consists of session 1 (creation) and session 2 (replication with 5-seccond timer). Dataset B consists of session 1 (creation) and session 3 (replication with 10-second timer). Dataset C consists of session 1 (creation) and session 4 (replication with 15-second timer). For each password entered by the user, four features were extracted: password score, password length, password speed, and password size. All four features extracted from the dataset were normalized before being used for training and testing the model.

D. ML Models

Three different models A, B, and, C were trained corresponding to datasets A, B, and C to determine the impact of variation of password presentation time. Five different ML algorithms- Decision Tree, Linear Discriminant Analysis, Naïve Bayes Classifier, SVM with Gaussian radial Basis function, and K-Nearest Neighbor algorithms were used to train different models. The reason for selecting five different machine algorithms was to compare the performance of different algorithms for gesture-based password evaluation. Three different datasets (A, B, and C) were used with five different ML algorithms, resulting in 15 different models.

Since the dataset was not big, 10-cross validation was used to evaluate the performance of the models.

V. Results

We used MATLAB software for ML analysis. Various trained models were evaluated using Classification Accuracy. Classification Accuracy gives the percentage of correct classifications. Fig. 3 shows the classification accuracies obtained for various trained models.

In Fig. 3, the blue color bar corresponds to Dataset A, the red color bar corresponds to dataset B, and the yellow bar color bar corresponds to dataset C. As can be seen in Fig. 3, SVMs with Gaussian Radial Basis function outperform other ML algorithms.

In order to determine the variation of password presentation time (5 seconds, 10 seconds, and 15 seconds), further analysis was carried out. Confusion Matrix metric was computed to evaluate the effect on the timers. Confusion Matrix gives a better visual representation than the Classification Accuracies, since it shows the number of true positives, true negatives, false positives, and false negatives.

A. Replication with 5-second Timer

Dataset A was formed by combining the data generated during the creation phase (session 1) and replication with 5-sec timer (session 2). Fig. 4 shows the Confusion matrix dataset A using SVM with Gaussian Radial Basis function Algorithm.

As can be seen from Fig. 4, the model trained with SVM with Gaussian Radial Basis function was able to classify both the classes, i.e. creator and intruder, very accurately. The dataset A consisted of 3535 samples from both creators and

intruders. From the confusion matrix one can see that 14% of the un-authorized users were incorrectly classified as authorized users.

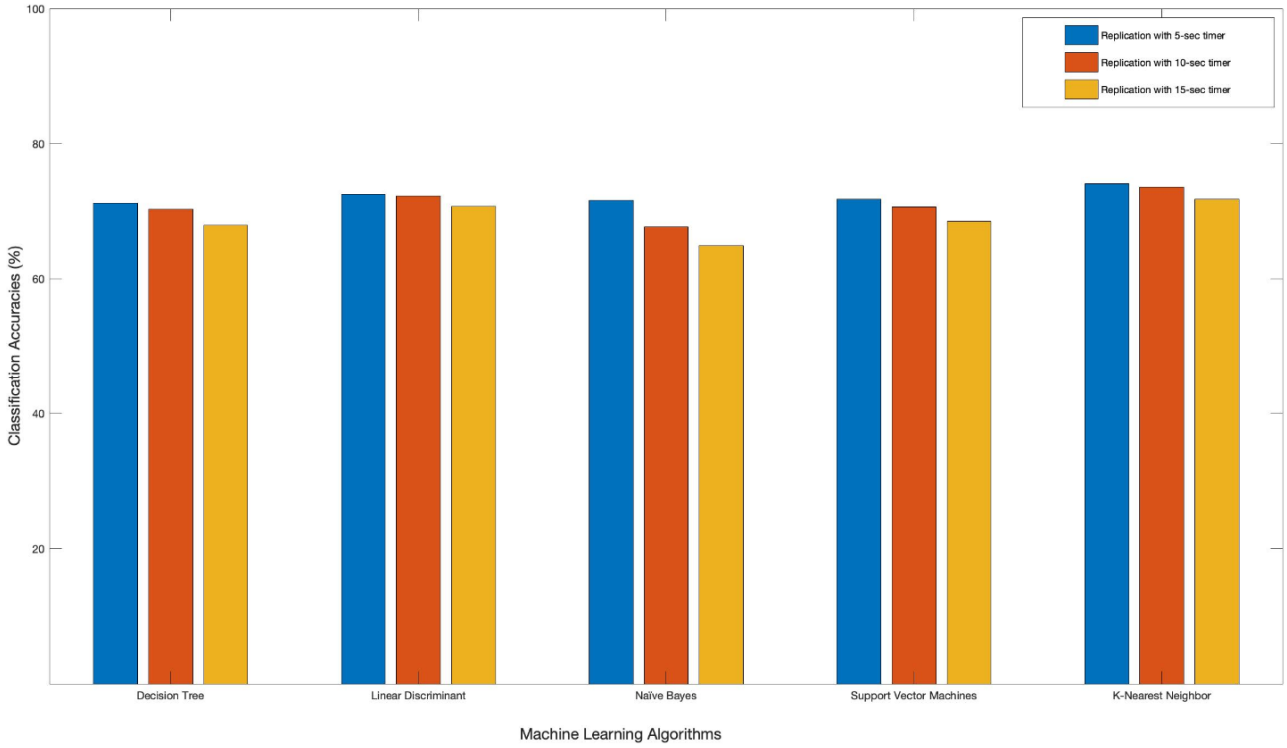

Fig. 3 Classification accuracies for various ML Algorithms for datasets A, B, and C

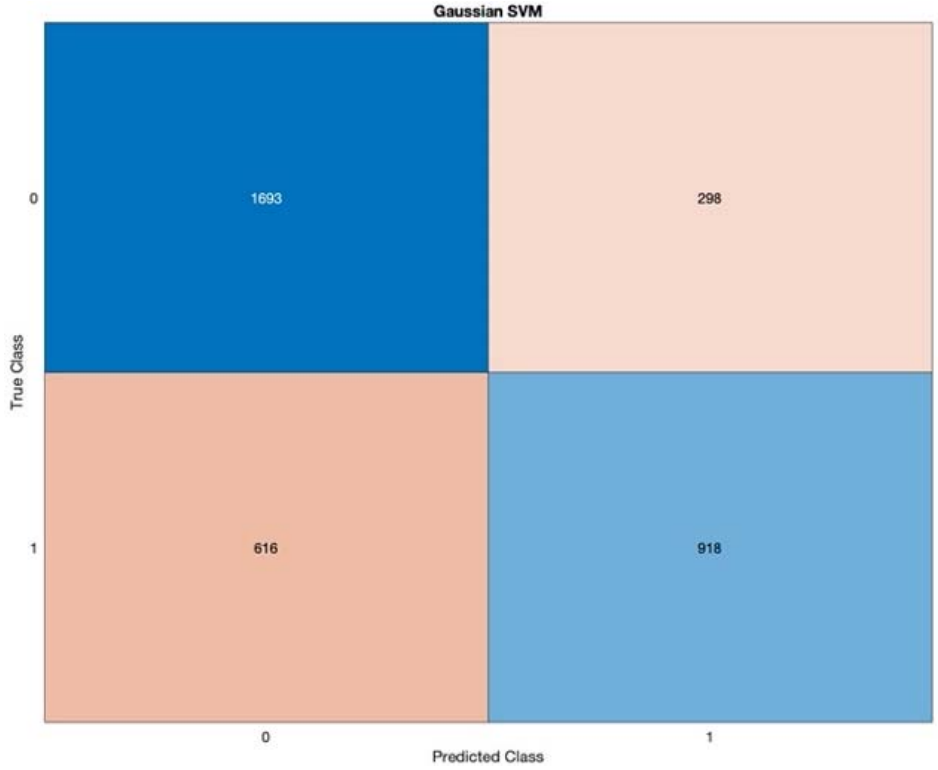

Fig. 4 Confusion matrix for Dataset A using SVM with Gaussian Radial Basis function

B. Replication with 10-second Timer

Dataset B was formed by combining the data generated during the creation phase (session 1) and replication with 10-sec timer (session 3). Fig. 5 shows the Confusion matrix dataset B using SVM with Gaussian Radial Basis function algorithm.

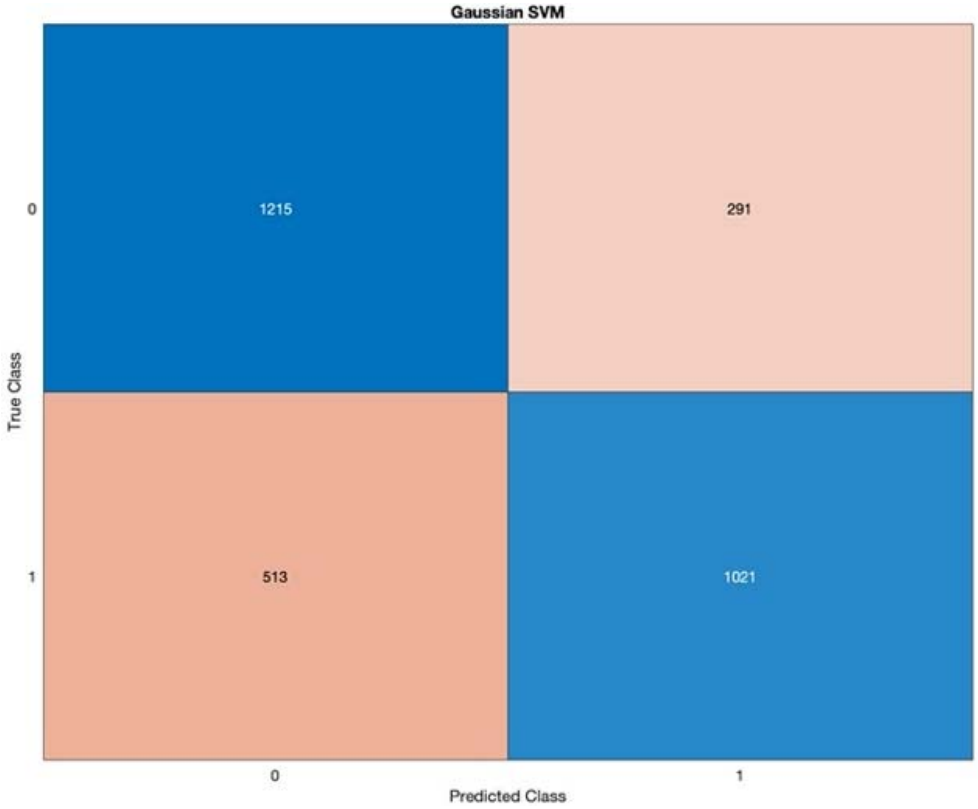

Fig. 5 Confusion matrix for Dataset B using SVM with Gaussian Radial Basis function

Dataset B consisted of 3040 samples from both creators and intruders. The confusion matrix reveals that 19% of un-authorized users were incorrectly classified as authorized users. As can be seen from Fig. 5, the model is still able to distinguish between the creator and intruder, but the accuracy has dropped compared to Dataset A.

C. Replication with 15-second Timer

Dataset C was formed by combining the data generated during the creation phase (session 1) and replication with 15-sec timer (session 4). Fig. 6 shows the Confusion Matrix dataset C using SVM with Gaussian Radial Basis function algorithm.

Dataset C consisted of 2769 samples from both creators and intruders. From the confusion matrix it is clear that 33% of un-authorized users were incorrectly classified as authorized users. As can be seen from Fig. 6, the model is still able to distinguish between the creator and intruder, but the accuracy has further dropped compared to Dataset A and Dataset B.

Figs. 3-6 indicate that as the presentation time for the password or the time for shoulder surfing attack increases from 5 seconds to 15 seconds, the performance of the model drops. In addition, we were interested to see if there is a difference in the confidence scores [19] of the passwords with three different timers (5, 10, 15). As Fig. 7 shows, there is a significant difference between the confidence score of creators vs. intruder with timers.

VI. Discussions and Future Work

A. Usability

The results of the study show that users were able to easily learn the gesture-based password method. Participants felt that it was easy to create the gesture-based passwords. They could enter the passwords in a fairly short amount of time (typically less than 20 seconds), and they did not feel frustrated when creating and re-entering gesture passwords.

B. Difference between Conditions

As the researchers expected, the confidence scores of passwords re-entered by creators are significantly higher than those of the passwords re-entered by intruders. There is a statically significant difference between the confidence scores of the passwords replicated with 5 second and 10 seconds. Also, significant difference was observed in other features, such as length, size and speed of gesture-based passwords.

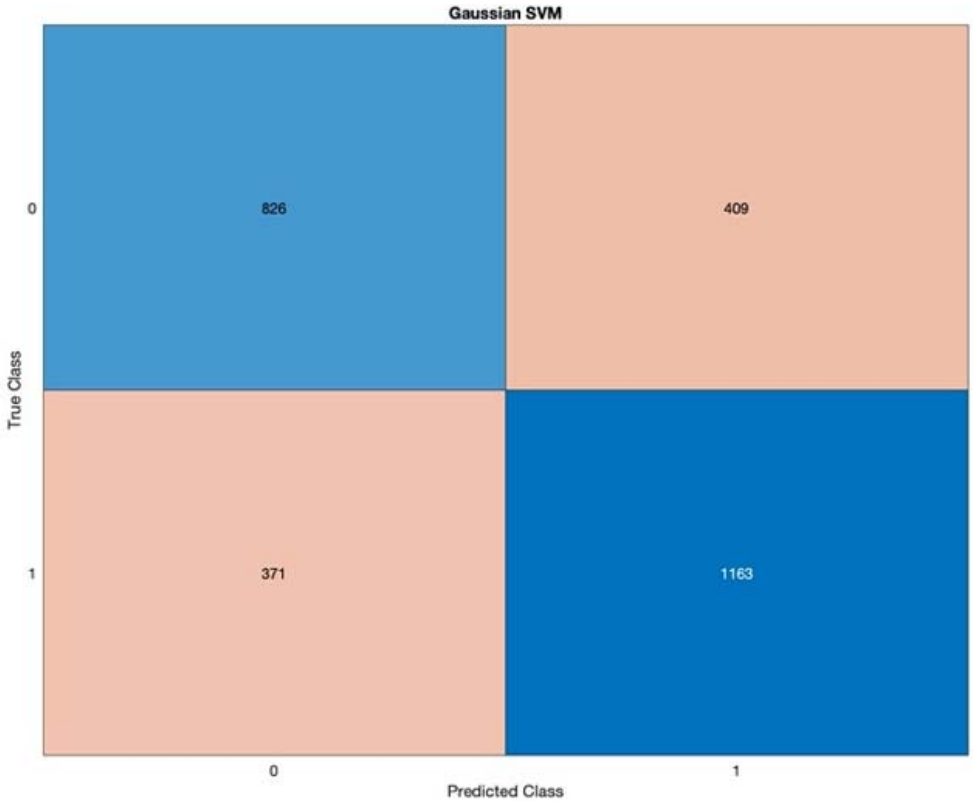

Fig. 6 Confusion matrix for Dataset C using SVM with Gaussian Radial Basis function

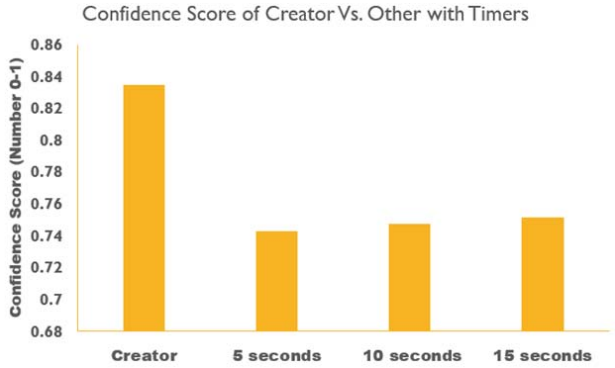

Fig. 7 Confidence score of passwords by creator vs. intruders with timers 5, 10, and 15 seconds

C. ML Results

This research study presents successful application of various ML algorithms in distinguishing between gesture-based password entered by authorized and unauthorized users. The longer is the duration of password exposure, the more difficult it is for the algorithm to distinguish between the intruder and non-intruder. As can be seen from Fig. 3, classification accuracies for all the ML algorithms used in this study drop as the exposure time increases from 5 seconds to 15 seconds. Of the ML algorithms used, SVM with Gaussian Radial Basis function outperformed the others. SVM with Gaussian Radial Basis function algorithm uses Gaussian radial basis function to transform the behavioral features (password score, password length, password speed, and password size) into a linear domain. The linearly transformed features are used for classification. Other ML algorithms do not transform the password features into non-linear domain, and thus the other ML algorithms do not perform well, have lower classification accuracy rate compared to the SVM with Gaussian Radial basis function. It can be concluded that password score, password length, password speed, and password size have non-linear relationship in determining if the password was entered by the authorized or un-authorized user.

In the future, we look forward to study the vulnerability and usability of gesture-based passwords by collecting even more detailed data, such as stroke level of passwords. Analyzing the stroke-level data using deep learning algorithms will further aid us to distinguish between un-authorized and authorized users of gesture-based passwords.

VII. Conclusion

SVMs with Gaussian Radial Basis Kernel outperform other ML algorithms for gesture-based password authentication. Results confirm that the shorter the duration of the shoulder-surfing attack, the higher the authentication accuracy. In conclusion, behavioral features extracted from the gesture-based passwords lead to less vulnerable user authentication. However, this method is quite vulnerable to shoulder surfing. Design modifications are needed to improve its robustness

against shoulder surfing attacks.

REFERENCES

[1] X. Suo and G. S. Owen, "Graphical Passwords : A Survey," *21st Annu. Comput. Secur. Appl. Conf.*, no. Acsac, pp. 10 pp. – 472, 2005, doi: 10.1109/CSAC.2005.27.

[2] A. Paivio's and J. C. Yuille, "Imagery, Memory and Cognition (PLE: Memory)," in *Essays in Honor of Allan Paivio*, London: Psychology Press, 1983, pp. 65–139.

[3] P. Jadhao and L. Dole, "Survey on Authentication Password Techniques," *Int. J. Soft Comput. Eng.*, vol. 3, no. 2, pp. 67–68, 2013.

[4] S. Chiasson, A. Forget, E. Stobert, P. C. van Oorschot, and R. Biddle, "Multiple Password Interference in Text andClick-Based Graphical Passwords," in *Proceedings of the 2009 ACM Conference on Computer and Communications Security*, 2009, pp. 500–511, doi: https://doi.org/10.1145/1653662.1653722.

[5] A. Y. Ng and M. I. Jordan, "On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes.," *Adv. neural Inf. Process. Syst.*, 2002.

[6] S. Keerthi and Sathiya, "A fast iterative nearest point algorithm for support vector machine classifier design," *IEEE Trans. neural networks 11.1*, pp. 124–136, 2000.

[7] K. Chomboon, "An empirical study of distance metrics for k-nearest neighbor algorithm," *Proc. 3rd Int. Conf. Ind. Appl. Eng.*, 2015.

[8] D. C. Feldmeier and P. R. Karn, "UNIX Password Security - Ten Years Later," in *Proceedings of the 9th Annual International Cryptology Conference on Advances in Cryptology*, 1989, pp. 44–63.

[9] R. N.Shepard, "Recognition Memory for Words, Sentences, and Pictures," *J. Verbal Learning Verbal Behav.*, vol. 6, pp. 156–163, 1967, doi: https://doi.org/10.1016/S0022-5371(67)80067-7.

[10] L. Sreeramareddy, J. H. Feng, and A. Sears, "Poster : Preliminary Investigation of Gesture-Based Password : Integrating Additional User Behavioral Features," pp. 4–5.

[11] I. Jermyn *et al.*, "The Design and Analysis of Graphical Passwords," in *Proceedings of the 8th USENIX Security Symposium*, 1999.

[12] D. Nali and J. Thorpe, "Analyzing user choice in graphical passwords," *Sch. Comput. Sci. Carlet. Univ. Tech Rep TR0401*, pp. 1–6, 2004, doi: 10.1.1.85.998.

[13] C. Varenhorst, M. Van Kleek, and L. Rudolph, "Passdoodles ; a Lightweight Authentication Method under the direction of," 2004.

[14] D. Mejía V. and J. Doose, "Gesture Based Touchpad Security System."

[15] A. De Luca, A. Hang, F. Brudy, C. Lindner, and H. Hussmann, "Touch me once and i know it's you! Implicit authentication based on touch screen patterns," *Conf. Hum. Factors Comput. Syst. - Proc.*, no. January 2014, pp. 987–996, 2012, doi: 10.1145/2207676.2208544.

[16] N. Sae-Bae, K. Ahmed, K. Isbister, and N. Memon, "Biometric-rich gestures: A novel approach to authentication on multi-touch devices," *Conf. Hum. Factors Comput. Syst. - Proc.*, no. May, pp. 977–986, 2012, doi: 10.1145/2207676.2208543.

[17] L. Sreeramareddy, A. Janprasert, and J. Heidifeng, "Evaluating Gesture-Based Password And Impact of Input Devices," in *Proceedings of the International Conference on Security and Management (SAM)*, 2014.

[18] L. Sreeramareddy, S. Miao, and J. H. Feng, "Investigating gesture-based password: Usability and vulnerability to shoulder-surfing attacks," *Res. Adapt. Converg. Syst. RACS 2014*, pp. 230–235, 2014, doi: 10.1145/2663761.2664207.

[19] L. Anthony and J. O. Wobbrock, "A lightweight multistroke recognizer for user interface prototypes," *Proc. - Graph. Interface*, pp. 245–252, 2010.