Passowrd policy characteristics and keystroke biometric authentication

Simon Parkinson¹ | Saad Khan¹ | Andrew Crampton¹ | Qing Xu² | Weizhi Xie¹ | Na Liu¹ | Kyle Dakin¹

¹Department of Computer Science, School of Computing and Engineering, University of Huddersfield, Huddersfield, UK
²College of Intelligence and Computing, Computer Science and Technology, Tianjin University, Tianjin, China

Correspondence
Simon Parkinson, Department of Computer Science, School of Computing and Engineering, University of Huddersfield, Huddersfield, UK.
Email: s.parkinson@hud.ac.uk

Abstract
Behavioural biometrics have the potential to provide an additional or alternative authentication mechanism to those involving a shared secret (i.e. a password). Keystroke timings are the focus of this study, where key press and release timings are acquired whilst monitoring a user typing a known phrase. Many studies exist in keystroke biometrics, but there is an absence of literature aiming to understand the relationship between characteristics of password policies and the potential of keystroke biometrics. Furthermore, benchmark datasets used in keystroke biometric research do not enable useful insights into the relationship between their capability and password policy. Herein, substitutions of uppercase, numeric, special characters, and their combination of passwords derived from English words are considered. Timings for 42 participants for the same 40 passwords are acquired. A matching system using the Manhattan distance measure with seven different feature sets is implemented, culminating in an Equal Error Rate of between 6% and 11% and accuracy values between 89% and 94%, demonstrating comparable accuracy to other threshold-based systems. Further analysis suggests that the best feature sets are those containing all timings and trigraph press to press. Evidence also suggests that phrases containing fewer characters have greater accuracy, except for those with special character substitutions.

1 INTRODUCTION

The use of digital resources throughout modern society has mandated the need to adequately restrict access to authorised users. Restricting access requires a robust and reliable means of performing user authentication. In most restriction mechanisms, a user will state an identity and then provide information that can verify that they have indeed claimed their correct identity. The most common implementation is a password system, where the password is kept secret and used to authenticate a user. Although password-based systems provide a good first-line defence, there is a continuing struggle between the complexity of the password to implement strong security and the user's desire for ease-of-access. To withstand brute-force attacks, a strong password policy is often enforced involving a combination of lower and upper case, numeric, and special characters that should be frequently changed [1]. However, passwords generated to this specification become hard to remember, resulting in the user finding ways to increase memorability, which includes performing systematic variation (i.e. increasing a number at the end) or simply writing the password down so that it does not need to be remembered. It is widely acknowledged that there is a trade-off to be achieved between memorability and security, and it is challenging to create a policy to guide the user to good practice without being overly restrictive and making it difficult to remember [2,3]. Evidence suggests that there is good and increasing use of passwords using substitutions [4].

The challenges of password-based authentication systems have inevitably resulted in the exploration of approaches that have strong security properties whilst reducing the burden of remembering on the user. Biometric systems provide an alternative approach to authentication whereby physiological, behavioural and chemical characteristics are sensed and used to authenticate a user in replacement (or in addition) to a password. Biometric systems have always been heavily researched,
and there is a wealth of literature on the different types and their applications, as presented by these extensive surveys [5,6]. For example, in terms of physiological, fingerprints, and facial features are examples. In terms of behavioural, voice, and keystroke dynamics are two examples. For chemical biometrics, DNA and odour are two examples. Recent advancements also include using behavioural biometrics extracted from wearable fitness trackers [7], and also using it for clinical purposes, such as stress detection [8]. Biometric systems impact on their users in different ways, depending on whether they are invasive or passive for the participant. For example, facial recognition is passive as it can be captured from video or image of the user without them taking special action, whereas a fingerprint system is invasive as the user needs to physically connect with a sensing device.

Herein, we revisit the topic of keystroke biometrics for authentication. Keystroke timings can be acquired passively from the user whilst they type their password. The use case is that the keystroke information is acquired as the user enters their password, thus providing a two-factor approach and helping to prevent against their account being compromised by a brute-force attack or being used should their password be acquired by an adversary. This is a well-established biometric area and its first use can be traced back to analysts in World War II aiming to track operators sending Morse Code messages via radio communication [9]. There have since been many studies on keystroke biometrics, focusing on the extraction of different features, using different phrase lengths, and different learning algorithms [10–12].

Although many studies do exist providing useful insight, there is a lack of empirical analysis investigating how keystroke biometric systems could be affected by different password policy characteristics (uppercase, numeric, special character, and length) that are often introduced to increased password security. The majority of key works in this area use datasets acquired from typing either the same phrase or from monitoring all user typing input, consisting of free-text input and no single password phrases. Furthermore, the phrases extracted in these datasets would not meet the majority of password policies, where a uppercase, numeric, or special symbol substitution are required. As we are investigating the use case of keystroke biometric authentication as an additional authentication layer on the user’s password, it is necessary to gain timings from phrases containing characteristics that are aligned with common password policies. There is a need to understand how keystroke systems perform based on different password characteristics, to ensure that a balance can be achieved whereby the combined use of password and keystroke provide security beyond that of each single component.

To investigate the relationship between password policy characteristic and keystroke biometrics, an empirical study has been performed, involving acquiring timings from 42 participants, before systematically analysing using a distance measure and threshold approach. The following are the main contributions provided in this work:

- The design of a rigorous keystroke biometric data collection exercise with varying password policy characteristics (length and character substitution), resulting in a publicly available dataset containing found sets timings for 42 individuals for 40 passwords.
- Equal Error Rates are achieved between 6% and 11% with accuracy between 89% and 94% for all feature sets. Empirical observations reveal the impact of the number of characters and types of character substitutions, providing new knowledge on their relationship.

The remainder is organised as follows: In Section 2, related work is presented and discussed. Section 3 details the considered features and the methodology followed in this research. Section 4 discusses acquired data and details enrolment template creation. Section 5 details the matching mechanism used in this research followed by the process for empirical testing. Section 6 details the results and considers the impact of different password policy components, and finally, a conclusion and area for future work are provided.

2 | RELATED WORK

Behavioural biometrics are considered as a security strengthening addition to the existing authentication schemes. The domain of behavioural biometrics is an active research discipline and is continuously evolving in capability. For example, a recent study proposed a new smartphone authentication system, called AnswerAuth, which is based on the way a user unlocks their phone and brings it towards their ear [13]. The evaluation was performed on a data set of 85 users, containing 10,200 patterns, and achieved 98.98% accuracy using a Random Forrest classifier. Another recent work investigated the feasibility of using gaze-based behavioural indices to perform user classification [14]. The developed solution was tested on nine participants and obtained 97% accuracy. Many research studies also propose the fusion of multiple behavioural biometrics (i.e. Multi-modal techniques) to gain better performance, such as integrating eye movement and mouse dynamics achieved 92.9% accuracy for a short subject registration time of 20 s [15].

Previous behavioural biometric studies are predominantly publicly available datasets, which contain a large number of samples (repetitions) for a large number of users. In keystroke biometrics, these datasets are often acquired from monitoring users typing free text input. For example, the Aalto University Dataset containing key timings for 168K participants over a 3-month time period [16]. In collecting the data, users type 15 sentences from a large sentence pool. The sentences are in English and are not provided as a single phrase, such as a password. Furthermore, the data has no substitutions. Therefore, there is an absence of data sets where the participants are typing phrases with different types of character substitutions in a controlled manner. This is because the large datasets have often been acquired passively through key logging from willing
participants when typing phrases longer than passwords. The lack of a suitable dataset could be a motivating reason as to why the relationship between password policy and keystroke capabilities has received less attention. The research presented herein is positioned to provide knowledge on this important topic.

In terms of the development and progression of key dynamic biometric systems, one of the earliest studies was performed on 50 subjects, where each subject chose their own password of 6–15 characters in length [17]. In the study, the researchers achieved 91% accuracy through the use of Multi-layer Perceptron (MLP) and Probabilistic Neural Network (PNN) algorithms. The achievement of this study is significant as it used representative password phrases. Subsequent studies were able to improve in terms of accuracy. For example, one study [18] also applied MLPs on 20,400 data samples using 34 features and achieved an Equal Error Rate (EER) of 4.45%. The dataset was acquired from 51 subjects by typing \texttt{.tie5Roaml} as the password, which contains a special character, upper and lowercase characters as well as a number. Although these results are promising, the single phrase password, not closely representing a word, does not provide an insight into how the EER might change should the password change in terms of length and substitution. Another research work developed a new, highly scalable keystroke-based biometric system called, TypeNet [19]. The TypeNet performs user authentication using Siamese Recurrent Neural Network (RNN). The evaluation of TypeNet was performed on one of the largest keystroke dataset, containing 136 million keystrokes from 168,000 users, and achieved 9.53%–3.33% EER.

Other studies have continued to build upon earlier success, using the same dataset. For example, another study [20] used a combination of Support Vector Machine (SVM) and Deep Learning algorithms to train keystrokes (key press/release events) from the \texttt{.tie5Roaml} password. The data was acquired from 51 subjects, where each subject performed 400 repetitions of a password. The solution demonstrated an accuracy of 92.60%. Another paper used SVMs to identify users based on keystroke biometrics [21]. A set of 155 features were extracted from the \texttt{.tie5Roaml} password using different combinations of timings. Using a data set gathered from 94 subjects, the developed solution demonstrated over 97% accuracy.

Research has also been undertaken using different datasets. In one work, the authors collected 4126 data samples from 22 subjects when typing the numeric password of 766420 [22]. In another study [23], a database of 7600 sequences was acquired over the duration of 4 months from 300 users. The sequences are users' first name and surname typed together. The features used in this study are hold time, release-press latency, and press-press latency. A total of 31 machine learning algorithms were used for evaluation. The solution has achieved EER as low as 5.32%. A different study [24] used \texttt{.tie5Roahnbg}, \texttt{.aeibo2246@}, \texttt{.nzkla29zab.}, and \texttt{.aeR5t.tlnh} passwords to acquire 30,000 samples from 1000 voluntary subjects. The dataset was used to train several classifiers, demonstrating accuracy of >98% and an ability to manage large datasets. This study shows that the characteristics of passwords (e.g. length, types of letters, entropy, etc.) impact on the feature extraction process, and consequently the accuracy and performance of the overall authentication system. Another study introduced the use of a Partially Observable Hidden Markov Model (POHMM) [25]. The authors used five publicly available keystroke datasets to evaluate the POHMM. The POHMM consistently achieved higher performance and better accuracy over existing solutions in terms of user identification and (continuous) verification.

Keystroke authentication systems have also been developed for mobile devices. One such work used the \texttt{.tie5Roaml} password to acquire a keystroke dataset from 42 subjects [26]. Each subject typed the same password for 60 times on an Android device. Following this, keystroke dynamics were used to extract 71 made up from different timing combinations. After testing multiple algorithms, it was concluded that the Random Forest obtained the highest accuracy of 82.53%. A recent study proposed the consolidation of 2-graph and 3-graph time-based features and achieved 4.19% false acceptance rate and 4.59% false-positive rate [27]. The keystroke data was acquired from 152 subjects using a digit-only password, containing 17 characters. The password was typed on mobile touchscreen 10 times by each subject. A different study built a solution to handle typing errors whilst entering passwords [28]. This included collecting timings from 10 subjects making intentional mistakes while typing the password of [Mohammed63] on a mobile device. As compared with random forest classifier that showed 12% EER, the proposed solution presented 9% EER and is based on time-based keystroke features.

A recent study [29] proposed the fusion of keystroke dynamics with gait patterns to perform continuous user authentication on the mobile devices. The data acquisition occurs while the user walks and inputs text, simultaneously. Based on an evaluation dataset collected from 20 participants, the developed solution achieved 99.11% accuracy using the multilayer perceptron classifier. Similarly, another study [30] demonstrates that using touch dynamics in addition to PIN-based authentication system significantly increases the level of security. In case of PIN number being fully compromised, the probability of a successful impersonation attempt is reduced from 100% to only 9.9%.

There are many other applications that have used similarity measures as a pose to machine learning techniques. This works by scoring a new sample against an enrolled template to determine if it is a positive match, within a predefined threshold [31]. Similarity measures include examples such as Gaussian mixture model-based matching [32] and distance measuring [31]. These techniques also demonstrate 90% accuracy, but there is the added challenge of determining decision boundaries. However, although their accuracy might be slightly lower than machine learning approaches, they have advantages in terms of research as they require no training phrase, resulting in them requiring less computation overheads. Furthermore, from a research perspective, using a similarity measure approach can help with understanding any relationships in the underlying datasets that could be hidden by the generalised approach of machine learning mechanisms.
The absence of any research and available dataset to in the area of understanding characteristics of password policies and keystroke dynamic systems motivates the design and collection of a suitable dataset. Furthermore, it is evident that a vast range of matching mechanisms can be used in keystroke dynamic systems with varying accuracy, but more traditional distance measuring approaches can still achieve good accuracy, which is significant considering their ease-of-use and transparent inner-workings.

3 | METHOD

To gain a comprehensive understanding of how password policy affects the performance of keystroke biometric systems, we have performed the following process:

- Established a set of 40 passwords of varying length constructed from English dictionary words. Substitutions of different types are then made to some of the passwords. These variations are uppercase, numeric, special symbol, and combination of all three;
- Collect timings for all 40 passwords for 42 participants. Each participant will type each password 4 times;
- Implement a distance-based scoring technique and threshold approach to compare test data sets with enrolled templates;
- Use three of a participant’s timings to process into an enrolled sample. The fourth sample is reserved for testing purposes, where each participant’s test sample is compared against all enrolled templates to see whether it matches the correct participant or not. The experiment is repeated with multiple threshold values; and finally
- We analyse the results to identify accuracy characteristics of the system in regard to the different password characteristics and feature sets.

The remainder of this section provides more detail on the above outlined process.

3.1 | Passwords

The passwords are all generated by selecting English dictionary words of varying complexity using the following criteria, which is guided by a survey into user practice in passwords [4]. In the study of six million passwords, 75% were between 8 and 10 characters in length. Furthermore, in excess of 40% are either native English words either in their unchanged form or with substitution. However, this study uses passwords generated from a system in Asia (acquired from the Chinese Software Developer Network) and therefore has a high portion (67%) of non-native English speakers. Other studies focusing on systems with a higher portion of English-speaking users report percentages [33]. In the study with six million passwords, 12.4% are alphabet only, 39% include at least one number, and 3.2% include the use of a symbol. Note that the password policy of the Chinese Software Developer Network did not require the use of a symbol. For these reasons, and in the interest of developing a representative password set, the following specification is followed to generate passwords. In total, 40 passwords of varying complexity are used to acquire timings from the user.

- Varying character lengths of 6, 8, 10, and 12. There will be two words for each character length;
- Two more English words for each length will be selected and one uppercase letter substitution will be performed in each word;
- Two more English words for each length will be selected and one numeric letter substitution will be performed in each word;
- Two more English words for each length will be selected and one special character substitution will be performed in each word; and
- Two more English words for each length will be selected and one uppercase, one numeric, and one special character substitution will be performed in each word.

The full combination of passwords is included in Appendix 9.1. As seen in the list of passwords, uppercase substitutions are used at different locations, whereas numeric and special character substitutions are used to replace characters where the shape of the number or special character is sufficient to act as a substitution for the normal letter. The specific special characters used in this experiment are: !, â€‹, |, $, \, #, and @. Each participant will be asked for the password a total of four times. The entire list will circulate three times in the same order, and finally, in the fourth iteration the ordering of the passwords will be randomised to the same order for all participants.

3.2 | Acquired timings

A software tool has been developed in C# to (1) present the password to the user, (2) acquire and record timestamps as they type the password, and (3) perform validation to ensure the user types the phrase correctly, and if not, ask them to repeat it. However, as the participants will perform their experiment unsupervised, there is the possibility that they can exit the software and not complete the exercise. The software tool records millisecond timings for the key down and key up actions as the password is displayed to the participant, until they select a ‘next’ button or press the carriage return key to inform the software they have finished. Figure 1 displays a graphical illustration of the individual key down and key up timestamps whilst the participant is typing the action password. There are in total 12 timestamps recorded from the participant typing the six-letter word. Each character will generate a timestamp when a key is pressed ($t_{VI}$) and one when the key is released ($t_{AI}$).
provide a full series of samples; however, upon analysis, it is evident that in some instances recordings are missing due to the user stopping and restarting the test. Due to incompleteness of only having one completed sample per password, some datasets are incomplete. In total 87% of all password and participant combinations have four samples. For example, participant 24 is missing responses (password 20, 23, 30, and 31). There are also many other instances where a user has provided two or three samples per password. This has occurred when a user exits the data collection exercise. However, as the work here presents an empirical analysis, the data is still used during the analysis as it can be used to gain an understanding of the technique's capabilities. There is also one instance (participant 34) where more samples have been provided than necessary. This is because the user has restarted the data collection exercise, resulting in too many timings being acquired.

Table A1 provides the number of times a participant provided repeated datasets for each password. Any number in the ‘0’ row means that there are instances where they submitted no response at all. The table counts up to six repetitions, which was the maximum acquired. As evident in the table, there are participants who provided only one series of timings for each password. These are participant 17, 21, 22, 23, 28, 37. The vast majority of participants provide the required four timings. Similarly, Table A2 provides the number of repetitions for each password, per participant. For most passwords, it is evident that 30 or more users provide four or more samples. However, this is not the case for passwords 6, 11, 14, 15 where 28–29 users provide four or more samples. Interestingly, these passwords have either no substitution or a single uppercase character. Another interesting finding from the table is that there are not more users providing a lower number of samples for passwords including all substitution types. This could suggest that the participant is taking more time and care with the difficult task.

To demonstrate variation in a user's timings, Figure 2a provides the timings for participant number 2 when typing the phrase action. The up and down timings for each key are displayed relative to the previous timestamp \( t_{\text{prev}} - t_{\text{next}} \). There are some important observations that can be established from looking at this example. Attempt number three has a delay between \( t_{\text{a}} \) and \( t_{\text{v}} \), which is not typical when considering the other three samples. Similarly, the difference between \( t_{\text{a}} \) and \( t_{\text{v}} \) is higher than with the other three samples. Both these timings are release to press (r-p) timings (seek timings) and would be ignored if this feature was not utilised. Furthermore, intuitively the forth sample is the quickest sample, demonstrating that the participant is more familiar with the phrase.
establish a representative dataset to enrol a user, it is necessary to remove this outlier. In this example, we consider that of \( t_{iv} \) demonstrated in Figure 2a as it contains an outlier. We use the m-score measure, which is a variation on the well-known z-score, and is used as a means to identify outliers. The reason behind using the m-score is its suitability for small sample sizes, which is pertinent considering in most instances there are four attempts at each password by each participant [34,35].

The median is calculated for each password character up and down action, which for example is \( \bar{X}_{iv} \). We then calculate the Median Absolute Deviation (MAD) for each character action \( MAD_{iv} \). We then calculate an M-score for each \( t_{iv} \) by the following:

\[
M^i_{iv} = 0.6745 \frac{t_{iv} - \bar{X}_{iv}}{MAD_{iv}}.
\]

The Median Absolute Deviation (MAD) will converge to the median of half the normal distribution, which is the 75% percentile of a normal distribution, and hence the value of 0.6745. For further information on equation for calculating the MAD, the readers are directed to early work by Iglewicz and Hoaglin [34].

We then replace \( t_{iv} \) with \( \bar{X}_{iv} \) if the calculated \( M^i_{iv} \) value is greater than 3.5. Figure 2b shows an updated series of attempt timings for participant 2 when typing action. It is evident that the \( t_{iv} \) outlier has been removed. It is however noticeable that there is still a large degree of variation in \( t_{iv} \). This is to be expected as there is a large degree of variation amongst all attempts, with no majority being closely aligned. As with all biometric systems, a template is created and stored for future matching. In this approach, the mean of the first three samples (outliers removed) is used. The purpose for only using the first three is to keep the fourth sample ‘unseen’ for empirical testing (Section 5). In some instances, the participants did not supply four timings per password. Full information on the number of times a participant provided repetitions for each password can be seen in Tables A1 and A2. Where a user has provided less than the four samples and providing the total number of samples for the password is greater than or equal to 2, the last sample is reserved for testing used during template generation. If only one sample has been provided, the same is used to create a template and test set. The mean of each up and down key timing is calculated by:

\[
\mu_{t_{iv}} = \frac{1}{j} \sum_{n=1}^{j} t_{iv}^n,
\]

where \( j \) is the number of \( t_{iv} \) samples. Furthermore, \( t_{iv}^n \) is the individual key down timings for the character ‘i’. In the example, we have \( t_{iv}^1, t_{iv}^2, \) and \( t_{iv}^3 \). Figure 2c demonstrates the template for the continuing timing set of participant 2 typing action. Furthermore, the figure also includes templates for the first five participants to illustrate their difference.

5 | EXPERIMENT

In this section, an experiment is conducted to determine how effectively key timing information can be used as a biometric. This specifically involves investigating:

- An approach to match and score a set of timings (test set) against a template set and determine whether or not it is a match
- Accuracy information when using the fourth samples of each participant to match against all enrolled templates
- The impact on performance of using the seven different feature sets, as discussed in Section 3.3.

5.1 | Matching

In this work, the Manhattan distance measure is used to score how well a testing sample matches that of an enrolled template [36] function is used. A distance measure and threshold approach has been adopted as previous research findings detail that both these mechanisms are part of systems that are capable of generating results comparable with applying general learning mechanisms (e.g. deep learning) [10,11,37]. Although general learning techniques have good potential, they often do not perform well on smaller datasets, are open to adversarial
attacks [38], their results can be difficult to interpret [39], and they are computationally expensive to use. Herein, the aim is to gain an understanding of the relationship between password characteristics and accuracy. This understanding could be easily lost when applying a general learning mechanism and therefore, we adopt a systematic and easy to interpret approach. The Manhattan measure is used over other available distance measures (e.g. Euclidean) as it has demonstrated to be more suitable to data with a higher number of dimensions [40]. This is calculated by:

$$d_{k,l} = \sum_{i=1}^{n} |t_i - e_i|$$

The testing set is denoted as $T = t_1, t_2, \ldots, t_n$ and the enrolled sample as $E = e_1, e_2, \ldots, e_n$ where both $t_i$ and $e_i$ are individual timings. $n$ is the size of both test and enrolled set. The Manhattan distance is measured for each testing set against all participant samples and recorded in $d_{k,l}$, where $k$ is the participant number and $l$ is the password template being used for comparison.

A decision function is established to determine whether the $d_{k,l}$ value for a specific participant and password is sufficient to determine a valid match. In this research, we use a threshold, $t$, which is used as a decision function. If the $d_{k,l} < t$ then the test sample matches the specific template, otherwise a non-match is returned. However, determining the correct threshold value is not trivial as it will vary between password and feature set. For example, in Figure 3 $d_{k,l}$ scores can be seen when measuring the test set acquired from participant 1 for action against each enrolled participant’s template for action. Furthermore, measurements for templates generated using each feature set are included. It is expected that $d_{1,l}$ (1 on x-axis) is to be the lowest as that is the correct match; however, as evident from the figure, there are instances where $d_{k,l}$ scores within other, incorrect passwords of similar low scores. It is therefore clear that determining an effective threshold is challenging and will inevitably result in false rejections and false acceptances. In the remainder of this section, empirical analysis is performed for all seven feature sets with a changing threshold value ($t$).

5.2 | Process

In this empirical analysis, templates are generated for each participant and password combination, as detailed in Section 4.1. As previously described, the last of the user's samples is removed and stored for testing. This results in the creation of an enrolled set, $E$, and a test set, $T$. In the majority of instances this results in three samples being used to create an individual participant and password template, $e$, and one sample being used as the test set, $t$. During testing, each test set will be compared against each enrolled template to determine if it is a suitable match. In our datasets, we have 1492 enrolled samples and 1670 test sets (generated from 4324 individual samples), which is approximately a one-third enrolled/testing split. The reason that the number of enrolled samples is smaller than the number of test sets is due to the fact that in some instances the participants did not supply four timings per password, as previously discussed in Section 4.1. As we are comparing each test set to each enrolment set, the experiment involves a total of 6,229,100 authentication decisions. In this analysis, the decision threshold, $t$, starts at 0 and is incremented by 0.1 until 100, where no further changes in performance were identified across all feature sets. This involves running the experiment on each future set a total of 1000 times. Furthermore, as we are performing this experiment to consider seven different features sets, almost seven trillion authentication decisions need to be made when performing the entire experiment. This takes around 2 h to complete on a computer with a 3.6 GHz i7 CPU with 16 GB of available RAM.

5.3 | Evaluation metrics

In this experiment, we are systematically comparing each test sample with all enrolled templates. It is therefore possible to establish the following measures:

- **True Positive Rate ($tpr$)**: number of instances where a valid correct template is matched to the correct test set;
- **False Positive Rate ($fpr = 1 - tnr$)**: the number of instances where an incorrect valid template match has been identified using the test set. This is also known as the False Acceptance Rate ($frr$);
- **True Negative Rate ($tnr$)**: the number of instances where an invalid match is returned for an incorrect template match;
- **False Negative Rate ($fnr$)**: the number of instances where an invalid match is returned for a correct template match. Also known as the False Rejection Rate ($frr$);
- **Accuracy**: the rate of correct decisions made. More specifically, $\text{Accuracy} = \frac{tpr}{tpr + fnr}$ and $\frac{tnr}{fpr + frr}$;
- **Equal Error Rate ($eer$)**: the point where $far = frr$.

6 | RESULTS AND DISCUSSION

Figure 4a presents the Receiver Operator Curve (ROC) showing the different $far$ and False Rejection Rate ($frr$) value with a changing threshold value. Figure 4b shows the same information, only focusing in on the lower section where $far$ values are less than 0.4 and $frr$ values are less than 0.2 to allow the reader to more easily see the difference in the feature set. From analysing Figure 4a and 4b, it is immediately evident that although there are differences between the results for each feature set, the results overall are not too dissimilar between feature sets, with variation often within 10%. The feature sets demonstrate that the best results are with the trigraph press to press ($\text{trigraph p-p}$) feature set, with the feature set containing all up and down timings ($\text{fp}$) following very closely behind. The feature set demonstrating the worst results is press to press ($\text{p-p}$), followed by release to release ($\text{r-r}$) and release to
press \((r\cdot p)\). The next two best are trigraph release to release \((trigraph \ r\cdot r)\) and press to release \((p\cdot r)\).

Table 1 presents the performance measures for each feature set where the \(far\) and \(frr\) become equal to within 1\% and represents the eer. The table also demonstrates the same orderings as illustrated in Figures 4a,b. The best eer value is observed to be 10\% for both the feature set containing all timings \((f-t)\) and trigraph press to press \((trigraph \ p\cdot p)\). Table 1 also provides the threshold value at which the eer was achieved. A general observation from these threshold values is that the higher the value, the lower the eer value. These results are promising and demonstrate the potential of a dynamic biometric system.

In the remainder of this section, the difference between password length, as well as the substitution of uppercase, special and numeric characters is further investigated and discussed. Individual ROC illustrations are used (Appendix 9.3) for each of the 40 passwords (Appendix 9.1).

Table 2 provides both the number of times a feature set has been identified as best for a participant's test data set and the threshold and accuracy values for the best overall feature set. The best overall values are calculated by calculating accuracy values for testing all participant test data sets. The table also shows the number of participants where each feature set produces the best performance for each password. This has been established by identifying the feature set resulting in the lowest eer, which is the point at which the \(far\) and \(frr\) intersect. However, it should be noted that they are not exactly equal (i.e., \(far = frr\)) because a stepsize of 0.1 was used to reduce the computation overheads when performing the experiment. In Table 2, only five feature sets are included, and this is because for all participant data sets, both \(p\cdot r\) and \(r\cdot r\) are never identified as being the best. More specifically, they never produce an eer that is lower than any other feature set when taking a participant's test set and comparing against all enrolled samples.

Before proceeding to analyse relationships between password features and key timing gestures, an immediate observation evident in Table 2 is that full timings set \((f-t)\) and trigraph press to press \((trigraph \ p\cdot p)\) are identified as the best feature sets for all passwords. However, this was to be expected as both these feature sets have the best overall performance, as provided in Table 1. In total, \(f-t\) is identified to be the best feature set for 32 passwords and \(trigraph \ p\cdot p\) for 8. Another general observation is that the best identified threshold values for \(trigraph \ p\cdot p\) (average 27) are lower than for \(f-t\) (average 49). The reason for this is because the \(trigraph \ p\cdot p\) will contain fewer timings for comparison and will result in a lower distance measure. Another general observation identified during the experimentation is that release to press \((r\cdot p)\) was the worst performing feature set and was only selected as the best for five users for passwords of length 6 with no substitutions.

Next, the impact of the password characteristics and capabilities of the proposed system are examined. Appendix 9.3 presents 20 individual ROC graphs presenting the results grouped by password characteristic which are in pairs. More specifically, the two passwords generated with the different characteristics are grouped together. For example, two passwords of six characters in length with no substitutions \(\textit{action}\) and \(\textit{return}\) are presented together in Figure A1.

In this discussion, we do not cross-reference with each ROC figure to keep the text easy to read; however, the authors do recommend the reader to review the ROC figures through the discussion as an illustrative aid. The ROC figures are clearly labelled as to their associated password. For example, Figure A1a–d presents the ROC for password lengths of 6, 8, 10, and 12, respectively. Furthermore, the ROC figures are grouped in five rows based on characteristics. The first (as four figures previously cross-referenced) have no substitution,
followed by a row including uppercase substitution, and so forth (Figures A2–A5). The reader can use these ROC figures throughout this section to aid understanding. In each figure, the ROC is provided for the seven different feature sets for the two passwords of each length.

Table 2 demonstrates that for passwords with no substitutions, the feature set identified as being best for the most participant test datasets are full timings set (f‐t) and trigraph press to press (trigraph p‐p). It is evident that trigraph p‐p has been identified as being best for only passwords with a length of 6 (action and return), and f-t for passwords of lengths 8, 10, and 12. Interestingly, it is also evident that both action and return have the largest variety of features identified as being best in comparison with all other passwords, irrespective of length and substitutions. In terms of accuracy, there is a gradual decrease from 94% to 91%. This is also evident in the eer where a decrease from 6% to 9%

In terms of uppercase substitutions, each of the passwords has a single uppercase character substitution made in different locations. Interestingly, the results for all phrases generate similar accuracy and eer rates of around 90% and 9%–10% (only 0.7% variation), respectively. Similar to instances with no uppercase substitution, f-t has also been identified as the best feature set for five out of eight of the passwords, and trigraph p‐p has been identified as best for three of the passwords, which are of 6 and 10 characters in length. An interesting observation is that those where trigraph p‐p has been identified as best, the position of the uppercase substitution is either the first or last character.

Following the same process, numeric substitutions have been made where all apart from two of the passwords lengths of 6 and 10 (cr1sis and addressin9) have the best performance with the trigraph p‐p features set. In terms of accuracy and eer rate, that values are quite consistent, showing a slight decrease (<1%) as password length increases.

Similar to instances with either no or uppercase substitution, f-t has also been identified as the best feature set for six out of eight of the passwords, and trigraph p‐p has been identified as best for two of the passwords. An interesting observation is that for only two participants are any other features sets identified as the best. This is the trigraph r-r set for the password of brok3n.

In terms of special character substitutions, f-t has been identified as best for the majority of password instances (7 out of 8) and trigraph p‐p is identified as the best for one password of 6 characters in length (fri!end). No other features have been identified as best for any participant data sets, apart from p‐p.

| Feature set              | Threshold | tpr  | tnr  | fpr  | fnr  | accuracy |
|--------------------------|-----------|------|------|------|------|----------|
| Full timings             | 52.9      | 89.33| 89.66| 10.33| 10.66| 89.66    |
| Press to press           | 28.1      | 88.99| 89.32| 10.67| 11.00| 89.32    |
| Press to release         | 33.8      | 86.91| 87.27| 12.72| 13.08| 87.27    |
| Release to press         | 24.6      | 87.79| 88.02| 11.97| 12.20| 88.02    |
| Release to release       | 38        | 87.52| 87.97| 12.02| 12.47| 87.96    |
| Trigraph press to press  | 30.8      | 89.16| 89.49| 10.50| 10.83| 89.49    |
| Trigraph release to release | 40.9  | 88.47| 88.92| 11.07| 11.52| 88.92    |

**Figure 4** Receiver Operator Curve for each feature set with varying threshold values
| Password         | ft | p-p | r-p | tri-p-p | tri-r-p | Best Feature Set | Threshold | Accuracy | tpr  | tnr  | fpr (far) | fnr (frr) |
|------------------|----|-----|-----|---------|---------|------------------|-----------|----------|------|------|----------|----------|
| action           | 15 | 9   | 1   | 16      | 0       | tri-p-p          | 27.3      | 94.40    | 94.40| 5.60| 5.56     |
| return           | 8  | 11  | 4   | 17      | 2       | tri-p-p          | 12.8      | 93.05    | 93.06| 6.95| 6.94     |
| bacteria         | 33 | 8   | 0   | 1       | 0       | ft               | 32.5      | 93.16    | 92.52| 6.84| 7.48     |
| football         | 30 | 1   | 0   | 7       | 2       | ft               | 31.3      | 91.06    | 91.37| 8.94| 8.63     |
| calculated       | 30 | 0   | 0   | 0       | 12      | ft               | 40.8      | 91.54    | 91.28| 8.46| 8.72     |
| automotive       | 42 | 0   | 0   | 0       | 0       | ft               | 42.2      | 91.41    | 91.26| 8.59| 8.74     |
| professional     | 42 | 0   | 0   | 0       | 0       | ft               | 44.6      | 91.01    | 91.18| 8.99| 8.82     |
| technologies     | 22 | 0   | 0   | 18      | 1       | ft               | 46.1      | 90.85    | 90.71| 9.15| 9.29     |
| Average values with no substitution: |     |     |     |         |         |                 | 92.06     | 91.98    | 92.06| 7.94| 8.02     |
| Filter           | 13 | 0   | 0   | 18      | 11      | tri-p-p          | 26.4      | 90.78    | 90.73| 9.22| 9.27     |
| docTor           | 32 | 0   | 0   | 9       | 1       | ft               | 45.5      | 90.65    | 90.75| 9.35| 9.25     |
| coMputer         | 38 | 0   | 0   | 4       | 0       | ft               | 46.1      | 90.35    | 90.46| 9.65| 9.54     |
| clickNing        | 29 | 0   | 0   | 11      | 2       | ft               | 26.7      | 90.20    | 90.23| 9.80| 9.77     |
| condiDions       | 33 | 0   | 0   | 9       | 0       | ft               | 45.8      | 90.39    | 90.32| 9.61| 9.68     |
| Conference       | 11 | 0   | 0   | 31      | 0       | tri-p-p          | 26.8      | 90.08    | 90.15| 9.92| 9.85     |
| disappointeD     | 7  | 0   | 0   | 35      | 0       | tri-p-p          | 46.5      | 90.13    | 89.98| 9.87| 10.02    |
| inflaMmation     | 26 | 0   | 0   | 16      | 0       | ft               | 27.4      | 90.14    | 90.23| 9.86| 9.77     |
| Average values with uppercase substitution: |     |     |     |         |         |                 | 90.34     | 90.36    | 90.34| 6.66| 6.94     |
| brok3n           | 30 | 0   | 0   | 10      | 2       | ft               | 27.7      | 90.02    | 89.88| 9.98| 10.12    |
| cr1sis           | 10 | 0   | 0   | 32      | 0       | tri-p-p          | 48.3      | 89.67    | 89.75| 10.33| 10.25    |
| deliv3ry         | 38 | 0   | 0   | 4       | 0       | ft               | 27.7      | 89.91    | 89.98| 10.09| 10.02    |
| annflying        | 25 | 0   | 0   | 16      | 0       | ft               | 48.3      | 89.68    | 89.68| 10.31| 10.32    |
| underst0od       | 29 | 0   | 0   | 13      | 0       | ft               | 48.6      | 89.69    | 89.57| 10.30| 10.43    |
| addressin9       | 15 | 0   | 0   | 27      | 0       | tri-p-p          | 48.9      | 89.63    | 89.63| 10.37| 10.37    |
| headqu4rters     | 30 | 0   | 0   | 11      | 0       | ft               | 49.4      | 89.53    | 89.51| 10.47| 10.49    |
| pr3scription      | 24 | 0   | 0   | 18      | 0       | ft               | 49.2      | 89.53    | 89.61| 10.47| 10.39    |
| Average values with numeric substitution: |     |     |     |         |         |                 | 89.71     | 89.70    | 89.71| 10.29| 10.30    |
| fr3nd            | 20 | 0   | 0   | 22      | 0       | tri-p-p          | 49.7      | 89.49    | 89.51| 10.51| 10.49    |
| gardÂën          | 24 | 0   | 0   | 18      | 0       | ft               | 50.3      | 89.41    | 89.31| 10.59| 10.69    |
| d|ameter          | 26 | 0   | 0   | 16      | 0       | ft               | 50.6      | 89.46    | 89.49| 10.54| 10.51    |
| receiVes         | 26 | 0   | 0   | 16      | 0       | ft               | 51.3      | 89.28    | 89.30| 10.72| 10.70    |
| university       | 22 | 0   | 0   | 20      | 0       | ft               | 51.2      | 89.09    | 89.12| 10.91| 10.88    |
| deviVering       | 22 | 0   | 0   | 17      | 0       | ft               | 51.7      | 89.05    | 89.05| 10.95| 10.95    |
| bre$htaking      | 25 | 4   | 0   | 12      | 0       | ft               | 52        | 89.26    | 89.19| 10.74| 10.81    |
| embarrassin2?    | 38 | 0   | 0   | 3       | 0       | ft               | 52.7      | 89.25    | 89.22| 10.75| 10.78    |
| Average values with symbol substitution: |     |     |     |         |         |                 | 89.29     | 89.27    | 89.29| 10.71| 10.73    |
| F@st3r           | 41 | 0   | 0   | 0       | 0       | ft               | 53.6      | 89.18    | 89.17| 10.82| 10.83    |
| pO!c3           | 40 | 0   | 0   | 0       | 0       | ft               | 54.2      | 89.24    | 89.29| 10.76| 10.71    |
| sc!enC3          | 42 | 0   | 0   | 0       | 0       | ft               | 53.8      | 89.40    | 89.35| 10.60| 10.65    |
for two passwords, one of length eight and one of length 10. These passwords are de-ivering and bre$t$aking. Accuracy and eer values remain consistent with a less than 1% decrease for different lengths.

As demonstrated in Table 2, passwords containing all three types of substitution are interesting because the $f_t$ feature set has been identified as best for all participant/password combinations. Furthermore, it is also interesting that this is the only substitution type whereby the accuracy values remain within 0.5% variance, yet the eer shows a slight improvement.

In terms of the average eer for passwords with different types of substitution, the best are those with no substitution, followed by uppercase, numeric, special character, and finally, a combination or all three. However, it is worth noting that the difference is small, decreasing from 91% accuracy and eer 8% for with no substitution to 89% accuracy and eer 11% (rounded to 0 decimal places).

Table 3 provides results on an aggregated level for the different password lengths. This includes a count of how many times trigraph $p$-$p$ and $f_t$ are identified as the best feature set for a password. Furthermore, the table includes average accuracy values for all passwords with that length, irrespective of substitution type. A general observation here are that $f_t$ is the dominant feature set, except for with passwords of 6 characters in length. Another observation is that the eer is gradually worsening as the password length is increasing; however, this is small and less than 1%. The results discussed so far focus on identifying the feature set with the lowest overall eer for all participant/password combinations. In Figure 5, the number of times each of the five feature sets are identified as the best for matching a test password against the enrolled template. The table is interesting as it demonstrates that there is good consistency with $f_t$ being the best feature set, followed by trigraph $p$-$p$ second for all participants. Any variation is often in the balance of $f_t$ and trigraph $p$-$p$, but it is clear that $f_t$ is always the best.

### 6.1 Summary of key findings

In summary to the previous section where results are presented and discussed, the below list provides a summary of the key findings from this study:

- Improved accuracy is achieved where no substitutions are present in the password (2% improvement). Furthermore, where no substitutions are present, the accuracy is best for passwords of a shorter length. More specifically, 94% for 6 characters in length, decreasing to 90% for those 12 characters in length. Where substitutions have been made, length has little significance with all substitution types having around 1% variation in accuracy and eer. This demonstrates the length is an important factor when utilising passwords with no substitutions;
- Full-timing ($f_t$) feature set is identified as being the best for the greatest number of participants, accounting for 69% of all participant/password combinations. Trigraph press to press (trigraph $p$-$p$) is the second best, accounting for 27%.
- The other three features identified as best for some participant password combinations account for 4% of the total. In addition, press to release and release to press are never identified as the best feature set for any participant/password combination. Considering that $f_t$ has been identified as best in the majority of instances, there is no advantage to converting the data into different feature sets;
- Passwords containing all three types of substitutions demonstrate a small decrease in performance from any individual substitutions instance. Furthermore, when compared with those with special character substitutions, the eer has a slight improvement; and
- There is a small decrease in accuracy as length increases (1%) across all feature sets, except where no substitutions are made. This demonstrates that length has little significance in relation to accuracy when a substitution of any type is introduced.

### 7 AVAILABILITY

The datasets used in this research are made available at: https://selene.hud.ac.uk/scomsp2/keydynamics.zip.

### 8 CONCLUSION

A systematic study of the key timings for 42 individuals for 40 different phases with varying length and with different
substitutions is presented. The presented approach involves removing outliers and creating a template for each user and phrase instance, which resulted in the refinement of individual samples. One of the user's samples is reserved from being included in the template as it is used for testing. The processed enrolled templates and test data sets are then used to produce different feature sets (e.g., press to press). The Manhattan distance measure is then used to quantify how close a test sample matches any single template. A threshold mechanism is then applied to determine whether a test sample's Manhattan distance measure is suitable to determine a match or a rejection.

Experimentation is performed to determine the threshold and feature set that yields the best performance. An eer of between 10% and 11% and accuracy values of 89%-90% were observed for all feature sets, with the lower being with the feature set containing all timings and trigraph press-to-press. Further empirical analysis was then performed to gain an understanding of how phrase size and substitutions impact on accuracy. Key observations have been determined and provide a useful insight for those designing keystroke biometric systems, as well as doing further research into their capabilities. Our study can help inform their experimental design. Of our key findings, it is interesting to have identified that the capabilities of a key dynamic system are at their best with no substitution and the password is shorter in length (6). This means that implementing a keystroke biometric system, alongside a strong password policy, might have a negative impact on the biometric system.

Although this study provides useful insights into password policy and its impact on biometrics, there are limitations to the study. The first limitation is in the number of participants and their similar technical competence. The second being in the number of phrases asked and the potential for more repetitive samples to be acquired to further investigate repeatability. Limitations are also present in the participant provides the timings sequentially on the same computer hardware, and gaining timings using different hardware over a prolonged period of time would help provide a more representative dataset. However, notwithstanding these limitations, this study has merits and the data is supportive of the findings. In future work, the authors intend to address the aforementioned as well as consider the use of different matching mechanisms.

**TABLE 3** Comparison between password length

| Password Length | Number with f-t | Number with tri-p-p | accuracy | tpr  | tnr  | fpr (far) | fnr (frr) |
|-----------------|-----------------|---------------------|----------|------|------|----------|----------|
| 6               | 5               | 5                   | 90.59    | 90.59| 90.59| 9.41     | 9.41     |
| 8               | 10              | 0                   | 90.29    | 90.27| 90.29| 9.71     | 9.73     |
| 10              | 8               | 2                   | 89.99    | 89.93| 89.99| 10.01    | 10.07    |
| 12              | 9               | 1                   | 89.87    | 89.87| 89.87| 10.13    | 10.13    |

**FIGURE 5** Number of times each feature set is identified as best for each participant

**ORCID**

Simon Parkinson 🌐 https://orcid.org/0000-0002-1747-9914
REFERENCES

1. Dell’Amico, M., Michiardi, P., Roudier, Y.: Password strength: an empirical analysis. In: 2010 Proceedings IEEE INFOCOM, pp. 1–9. IEEE, San Diego (2010)
2. Yan, J., et al.: Password memorability and security: empirical results. IEEE Secur. Privacy. 2(5), 25–31 (2004)
3. Yildirim, M., Mackie, I.: Encouraging users to improve password security and memorability. Int. J. Inf. Secur. 18(6), 741–759 (2019)
4. Shen, C., et al.: User practice in password security: an empirical study of real-life passwords in the wild. Comput. Secur. 61, 130–141 (2016). [Online]. http://www.sciencedirect.com/science/article/pii/S0167404816300657
5. Daigan, S., Kumar, M.: A comprehensive survey on the biometric recognition systems based on physiological and behavioral modalities. Expert Syst. Appl. 143, 113114 (2020)
6. Fairhurst, M., Li, C., Da Costa-Abreu, M.: Predictive biometrics: a review and analysis of predicting personal characteristics from biometric data. IET Biom. 6(6), 369–378 (2017)
7. Khan, S., et al.: Biometric systems utilising health data from wearable devices: applications and future challenges in computer security. ACM Comput. Surv. (2020)
8. de Arriba-Pérez, F., et al.: Study of stress detection and proposal of stress-related features using commercial-off-the-shelf wrist wearables. J. Ambient Intell. Human. Comput. 10(12), 4925–4945 (2019)
9. Pasatiuk, P., Sacek, K.: A modified algorithm for user identification by his typing on the keyboard. Image Processing and Communications Challenges, 2, pp. 113–120. Springer, Heidelberg (2010)
10. Teh, P.S., Teoh, A.B.J., Yue, S.: A survey of keystroke dynamics biometrics. Sci. World J., 2013 (2013)
11. Banerjee, S.P., Woodard, D.L.: Biometric authentication and identification using keystroke dynamics: a survey. J. Pattern Recogn. Res. 7(1), 116–139 (2012)
12. Bhatt, S., Santhanam, T.: Keystroke dynamics for biometric authentication—a survey. In: 2013 International Conference on Pattern Recognition, Informatics and Mobile Engineering, pp. 17–23. IEEE (2013)
13. Burio, A., Crispo, B., Conti, M.: Answerauth: A bimodal behavioral biometric-based user authentication scheme for smartphones. J. Inf. Secur. Appl. 44, 89–103 (2019). [Online]. http://www.sciencedirect.com/science/article/pii/S2214212618304435
14. Krishnappa Babu, P.R., Lahiri, U.: Classification approach for understanding implications of emotions using eye-gaze. J. Ambient Intell. Human. Comput. 11(7), 2701–2713 (2020). [Online]. https://doi.org/10.1007/s12652-019-01329-8
15. Kasprzowik, P., Harezlak, K.: Fusion of eye movement and mouse dynamics for reliable behavioral biometrics. Pattern Anal. Appl. 21(1), 91–103 (2018)
16. Dhalak, V., et al.: Observations on typing from 136 million keystrokes. In: Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems, 1–12 (2018)
17. Revett, K., et al.: A machine learning approach to keystroke dynamics based user authentication. International Journal of Electronic Security and Digital Forensics. 1(5), 55–70 (2007)
18. Andrean, A., Jayabal, M., Thiruchelvam, V.: Keystroke dynamics based user authentication using deep multilayer perceptron. Int. J. Mach. Learn. Comput. 10(1) (2020)
19. Acién, A., et al.: Typenot: Scaling up keystroke biometrics (2020). arXiv preprint arXiv:2004.03627.
20. Mulino, Y., Ham, H., Darmawan, D.: Keystroke dynamic classification using machine learning for password authorization. Proc. Comput. Sci. 135, 564–569 (2018)
21. Krishnamoorthy, S., et al.: Identification of user behavioral biometrics for authentication using keystroke dynamics and machine learning. In: Proceedings of the 2018 2nd International Conference on Biometric Engineering and Applications, 50–57 (2018)
22. Kim, D.L., Lee, S., Shin, J.S.: A new feature scoring method in keystroke dynamics-based user authentications. IEEE Access. 8, 27901–27914 (2020)
23. Morales, A., et al.: Keystroke biometrics ongoing competition. IEEE Access. 4, 7736–7746 (2016)
24. Baynath, P., Sojaudah, K.S., Khan, M.H.-M.: Machine learning algorithm on keystroke dynamics pattern. In: 2018 IEEE Conference on Systems, Process and Control (ICSPC), pp. 11–16. IEEE (2018)
25. Monaco, J.V., Tappert, C.C.: The partially observable hidden markov model and its application to keystroke dynamics. Pattern Recogn. 76, 449–462 (2018)
26. Antal, M., Szabó, L.Z., László, I.: Keystroke dynamics on android platform. Proc. Technol. 19, 820–826 (2015)
27. Trojahn, M., Arndt, F., Ortmieier, F.: Authentication with keystroke dynamics on touchscreen keyboards-effect of different n-graph combinations. In: MOBILITY 2013, The third international conference on mobile services, resources, and users, 114–119 (2013)
28. Abshanketi, F., Traore, I., Ahmed, A.A.: Improving performance and usability in mobile keystroke dynamic biometric authentication. In: 2016 IEEE Security and Privacy Workshops (SPW), pp. 66–73. IEEE (2016)
29. Lamiøe, I., et al.: A continuous smartphone authentication method based on gait patterns and keystroke dynamics. J. Ambient Intell. Human. Comput. 10(11), 4417–4430 (2019)
30. Teh, P.S., et al.: “Strengthen user authentication on mobile devices by using user’s touch dynamics pattern. J. Ambient Intell. Human. Comput. 1–21 (2019)
31. Veenopag, P.C.: Applying empirical thresholding algorithm for a keystroke dynamics based authentication system. J. Inf. Commun. Technol., 18(4), 383–413 (2020)
32. Ceker, H., Upadhyaya, S.: Enhanced recognition of keystroke dynamics using Gaussian mixture models. In: MILCOM 2015-2015 IEEE Military Communications Conference, pp. 1305–1310. IEEE (2015)
33. Campbell, J., Bryant, K.: Password composition and security: an exploratory study of user practice. In: ACIS 2004 Proceedings, 80 (2004)
34. Iglewicz, B., Hoaglin, D.: Volume 16: how to detect and handle outliers. The ASQC basic references in quality control: statistical techniques, 16 (1993)
35. Czoarkin, C., et al.: Nist/sematech e-handbook of statistical methods. NIST/SEMATECH, July. Available online: http://www.itl.nist.gov/div898/handbook (2006)
36. Cooke, D., Clarke, G.: A basic course in statistics, 6ed. Arnold, London (2004)
37. Zhong, Y., Deng, Y.: A survey on keystroke dynamics biometrics: approaches, advances, and evaluations. In: Recent Advances in User Authentication Using Keystroke Dynamics Biometrics, pp. 1–22 (2015)
38. Paperon, N., et al.: The limitations of deep learning in adversarial settings. In: 2016 IEEE European symposium on security and privacy (EuroS&P), pp. 372–387. IEEE (2016)
39. Samek, W., Wiegand, T., Müller, K.: Explainable artificial intelligence: understanding, visualizing and interpreting deep learning models (2017). arXiv preprint arXiv:1708.08296
40. Aggarwal, C.C., Hinneburg, A., Keim, D.A.: On the surprising behavior of distance metrics in high dimensional space. In: International conference on database theory, pp. 420–434. Springer (2001)

How to cite this article: Parkinson S, Khan S, Crampton A, et al. Password policy characteristics and keystroke biometric authentication. IET Biome. 2021;10:163–178. https://doi.org/10.1049/bme2.12017
APPENDIX

PASSWORD LIST

1. action
2. return
3. bacteria
4. football
5. calculated
6. automotive
7. professional
8. technologies
9. Filter
10. docTor
11. coMputer
12. clickiNg
13. conDitions
14. Conference
15. disappointeD
16. inflaMmation
17. brok3n
18. cr1sis
19. deliv3r y
20. ann0ying
21. underst0od
22. addressin9
23. headqu4rters
24. pr3scription
25. frlend
26. gardÅ’en
27. d|ameter
28. rec$ives
29. universIty
30. de:i:vering
31. bre$htaking
32. embarrassin2
33. F@st3r
34. pO!c3
35. sc|enC3
36. he4Veny
37. |ndig3nOus
38. in5ul@tion
39. aSynchr0#ous
40. cat@s7Rophic

PARTICIPANT RESPONSES

Table A1 provides the number of times a participant provided 0-6 repetitions. Table A2 provides the number of times a participant provides 0-6 repetitions for each password.

ROC PLOTS

FIGURE A1 Receiver Operator Curve for passwords with varying lengths (6, 8, 10, and 12 characters)

(a) action and return  (b) bacteria and football  (c) calculated and automotive  (d) professional and technologies

FIGURE A2 Receiver Operator Curve for passwords with varying lengths and uppercase letter substitutions

(a) Filter and docTor  (b) coMputer and clickiNg  (c) conDitions and Conference  (d) disappointeD and inflaMmation

FIGURE A3 Receiver Operator Curve for passwords with varying lengths and numeric substitutions

(a) brok3n and cr1sis  (b) deliv3r y and ann0ying  (c) underst0od and addressin9  (d) headqu4rters and pr3scription
FIGURE A4  Receiver Operator Curve for passwords with varying lengths and special character substitutions

(a) fr'end and gardÂën  (b) d|ameter and rec$ives  (c) university and de$ivering  (d) bre$htaking and embarrassin?

FIGURE A5  Receiver Operator Curve for passwords with varying lengths, uppercase, numeric, and special character substitutions

(a) F@st3rand pOl!c3  (b) sc!enCe and he4Ven|y  (c) ndig3nOus and in5ul@tIon  (d) aSynchr0#ous and cat@s7Rophic
### Table A1
Number of times each participant provided 0-6 repetitions, grouped by participant.

| Participant | 0 | 1 | 2 | 3 | 4 | 5 | 6 |
|-------------|---|---|---|---|---|---|---|
| #            |   |   |   |   |   |   |   |
| 1            | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2            | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3            | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4            | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 5            | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 6            | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

### Table A2
Number of times each participant provided 0-6 repetitions, grouped by password.

| Password | Number of responses |
|----------|---------------------|
|          | 0 | 1 | 2 | 3 | 4 | 5 | 6 |
| 1        | 0 | 2 | 4 | 4 | 31 | 1 | 1 |
| 2        | 0 | 3 | 3 | 3 | 32 | 1 | 1 |
| 3        | 0 | 5 | 1 | 4 | 31 | 0 | 0 |
| 4        | 0 | 5 | 1 | 5 | 30 | 0 | 0 |
| 5        | 0 | 6 | 0 | 4 | 31 | 0 | 0 |
| 6        | 0 | 6 | 0 | 6 | 29 | 0 | 0 |
| 7        | 0 | 6 | 0 | 2 | 34 | 0 | 0 |
| 8        | 1 | 5 | 1 | 6 | 28 | 0 | 0 |
| 9        | 0 | 6 | 1 | 4 | 30 | 0 | 0 |
| 10       | 0 | 6 | 1 | 2 | 32 | 0 | 0 |
| 11       | 0 | 6 | 0 | 6 | 29 | 0 | 0 |
| 12       | 0 | 6 | 0 | 4 | 31 | 0 | 0 |
| 13       | 0 | 4 | 2 | 1 | 34 | 1 | 1 |
| 14       | 0 | 6 | 1 | 5 | 29 | 0 | 0 |
| 15       | 0 | 6 | 0 | 6 | 29 | 0 | 0 |
| 16       | 0 | 6 | 0 | 4 | 31 | 0 | 0 |
| 17       | 0 | 6 | 0 | 4 | 31 | 0 | 0 |
| 18       | 0 | 6 | 0 | 4 | 31 | 0 | 0 |
| 19       | 0 | 6 | 0 | 5 | 31 | 0 | 0 |
| 20       | 1 | 6 | 0 | 4 | 31 | 0 | 0 |
| 21       | 0 | 6 | 0 | 4 | 32 | 0 | 0 |
| 22       | 0 | 6 | 0 | 4 | 32 | 0 | 0 |
| 23       | 1 | 6 | 0 | 5 | 30 | 0 | 0 |
| 24       | 0 | 4 | 2 | 5 | 30 | 0 | 0 |
| 25       | 0 | 6 | 0 | 5 | 31 | 0 | 0 |
| 26       | 0 | 6 | 1 | 0 | 35 | 0 | 0 |
| 27       | 0 | 6 | 1 | 4 | 31 | 0 | 0 |
| 28       | 0 | 6 | 2 | 2 | 32 | 0 | 0 |
| 29       | 0 | 6 | 1 | 4 | 31 | 0 | 0 |
| 30       | 1 | 7 | 0 | 0 | 34 | 0 | 0 |
| 31       | 1 | 6 | 1 | 4 | 30 | 0 | 0 |
| 32       | 1 | 6 | 2 | 2 | 31 | 0 | 0 |
| 33       | 1 | 5 | 1 | 3 | 32 | 0 | 0 |
| 34       | 2 | 4 | 2 | 1 | 33 | 0 | 0 |
| 35       | 0 | 5 | 1 | 5 | 30 | 0 | 0 |
| 36       | 0 | 5 | 1 | 3 | 33 | 0 | 0 |
| 37       | 0 | 5 | 1 | 3 | 33 | 0 | 0 |
| 38       | 0 | 5 | 1 | 4 | 31 | 0 | 0 |
| 39       | 0 | 5 | 1 | 5 | 30 | 0 | 0 |
| 40       | 0 | 5 | 1 | 2 | 33 | 0 | 0 |