A Novel Algorithm for Differentiating Authorized Users from Fraudsters via Mobile Keypad Input Patterns During Password Updates

Boumedyen Shannaq [1]

[1] *Department of Management Information Systems, College of Business,*
University of Buraimi, Sultanate of Oman

Abstract **- This study addresses the challenge of distinguishing legitimate users from imposters during password updates by developing the TKIP-RUB (Transforming Keypad Input Patterns to Recognize User Behavior) algorithm. A literature review revealed that existing algorithms, including the EPSB method, achieve limited accuracy in recognizing user behavior based on mobile keypad input. The research aimed to enhance authentication systems by answering the question: Can transforming historical input patterns improve the accuracy and reliability of user recognition? The hypothesis posited that the proposed algorithm would significantly outperform existing methods in accuracy and precision. To evaluate this, an experimental study was conducted using login attempts from 143 users who updated their passwords, resulting in a dataset of 629 records (486 training, 143 testing). The TKIP-RUB algorithm was integrated into a mobile authentication system to analyze user behavior and generate predictive patterns. Results showed that while the EPSB algorithm achieved an accuracy of 9.091%, the TKIP-RUB algorithm reached 53.147%, representing a fivefold improvement. This demonstrates the superior effectiveness of the TKIP-RUB algorithm in enhancing recognition rates, security, and positive predictive precision.**

This work underscores the importance of advanced algorithmic techniques in mitigating risks of unauthorized access and provides recommendations for future password creation strategies.

DOI: 10.18421/TEM141-68
https://doi.org/10.18421/TEM141-68

Corresponding author: Boumedyen Shannaq,
Department of Management Information Systems, College of Business, University of Buraimi,
Sultanate of Oman
Email: boumedyen@uob.edu.om

Received: 03 August 2024.
Revised: 30 December 2024.
Accepted: 21 January 2025.
Published: 27 February 2025.

Keywords **- Password patterns, mobile keypad input, security authentication, user behavior recognition, algorithm development.**

1. Introduction

The usage of mobile devices for sensitive data has expanded, posing a challenge to traditional password authentication methods like shoulder surfing and phishing [1], [2]. It is possible to improve mobile authentication security and stop unwanted access by forecasting user password trends [3], [4].

Many users consider mobile or portable devices an essential part of their lives, performing most tasks using these devices to access required services [5]. Typically, to access any service, users must type text or numbers using mobile keypad input, including creating new passwords for accessing sensitive data in systems such as email, bank accounts, and social media.

This work argues that the mobile keypad input distribution structure significantly influences users' password selection behavior. The proximity of certain letters on the keypad helps users to remember their created passwords. Based on this assumption, the current passwords are analyzed and compared to the mobile keypad input distribution structure. In the following sections, the development of the prediction and recognition model using user mobile keypad input behavior is explained in detail.

As the use of mobile devices to access sensitive data increases, authentication method security has emerged as a major problem [6]. Conventional password-based authentication methods are frequently susceptible to phishing, guessing, and shoulder surfing, among other types of assaults [7], [8], and [9].

Users frequently make passwords that are simple to figure out, which erodes security even further [10], [11].

Predicting and identifying user password patterns is an urgent way to improve the security of mobile authentication systems and stop unwanted access [12], [13], [14]. However, amidst the convenience of mobile technology lies a pressing concern: Password security. Weak passwords pose a significant threat to privacy, leaving systems vulnerable to exploitation by malicious actors [15]. [16] Hackers adeptly manipulate these vulnerabilities, assuming the guise of authorized users to gain illicit access. In response to this challenge, the research endeavors to fortify password security through innovative means. By leveraging numeric sequences entered via a mobile keypad and analyzing users' password histories, the meaningful patterns that enhance security measures are extracted. This proactive approach not only mitigates the risk of unauthorized access but also reinforces the integrity of user accounts, ensuring robust protection in an ever-evolving digital landscape. Even with the growing importance of mobile devices and the appearance of distinct typing behaviors similar to biometric signatures, there is a noticeable lack of research in the literature on the use of these patterns to improve security. While some research focuses on authentic users and imposters, only [17] has looked at this particular security route. Their study presented an authentication technique, but its testing and optimization were hampered by modest database sizes and user acceptability surveys. Moreover, the limitations of existing security measures and the algorithm's inefficiency with different password patterns and durations highlight a large vacuum in creative yet workable alternatives for strong security. This emphasizes the necessity of the newly proposed TKIP-RUB algorithm to improve the "Confidence Range (CR)" method, which seeks to generate distinct user profiles and monitor password behaviors in-depth, providing a more practical and efficient means of boosting security.

The contribution of this work intends to examine user behavior in creating password combinations on portable keypads, the study will analyze recurring themes and combinations, identify factors influencing password pattern selection, and develop predictive algorithms for future password patterns. These algorithms will be tested for accuracy in practical situations and integrated into current mobile authentication systems to enhance security and prevent unauthorized access. Additionally, the study aims to provide security recommendations for creating robust passwords, advising users on making safer passwords and avoiding common patterns.

2. Literature Review

The ubiquitous presence of mobile devices in daily lives have fundamentally altered how people interact with technology [18], [19], and [20].

Users seamlessly integrate mobile keypads into their routines, relying on them for essential tasks and even during moments of relaxation. This symbiotic relationship underscores the indispensability of mobile devices, shaping user behavior and preferences over time [21], [22]. Within this digital ecosystem, patterns emerge as users navigate their keyboards, revealing distinctive typing behaviors characterized by frequently selected letters.

These behavioral nuances, akin to biometric signatures, offer valuable insights into user interactions, presenting an opportunity to anticipate future behaviors. In the exploration of existing literature, prior studies related to recognizing unauthorized user from the authorized user based on the user password patterns that delve into security application systems designed to distinguish genuine users from impostors, were encountered. In existing literature, no research except the one by [15] investigates this security path, and introduces an algorithm focused on authentication as a core model for system access control. Passwords, a key mechanism for identifying authorized users, faces challenges such as spoofing and man-in-the-middle attacks. Unauthorized users with the correct password can access and change data, causing significant losses. Hackers also attempt to penetrate systems through password prediction.

This paper introduces the "Confidence Range" algorithm, which monitors password activities (time, error, style) to detect suspicious behavior, creating a unique "Electronic Personal Synthesis Behavior (EPSB)" for authorized users. [15] Tried to support their research by developing surveys to measure user acceptance instead of improving the algorithm , [23], [24]. Additionally, the database used in their research was not large enough to effectively test the proposed algorithm. However, this proposed mechanism reveals a weakness in its reliance on consistent usage of similar patterns whenever authorized users update their passwords. To effectively differentiate between authorized and unauthorized users who may have exploited the password, at least 60% of the same user patterns small letters, capital letters, numbers, and symbols must be repeated. Yet, many security systems prohibit password updates if 60% of the patterns are repeated, even if they appear in different positions [25].

Cypress Data Defense [26] documents six main password security threats which administrators can mitigate through the combination of authentication protocols with robust password systems under a policy of regular rotations.

The authors in [27] and [28] recommend essential password policies which stress the combination of complexity with periodic changes and strict administrator controls for security system advancement.

Consequently, it becomes evident that the CR approach is not compatible with current security policies and encounters significant difficulties in recognizing user patterns when there is less than 60% repetition. Additionally, the CR approach falters when adjusting password length, as revealed by the experiments indicating inefficiency, especially when new passwords deviate significantly from old patterns. This impracticality extends to real-world scenarios, as the algorithm demands substantial time and resources for processing. Furthermore, its effectiveness hinges on the availability of extensive user records, posing a challenge in situations with limited data availability.

There is a noticeable lack of research in the literature about the use of these patterns to improve security, even with the growing importance of mobile devices and the advent of distinctive typing tendencies similar to biometric signatures. Only [20] has looked at this particular security approach, whereas other researches concentrate on using password patterns to differentiate between real users and imposters. Although their work established an authentication technique, it was hampered in its ability to be effectively tested and improved by user approval surveys and a tiny database. Furthermore, there is a clear need for creative, workable solutions for strong security given the limitations of present security methods and the algorithm's inefficiency with different password patterns and durations.

This highlights the necessity of the proposed TKIP-RUB algorithm to improve the CR algorithm, which seeks to generate distinct user profiles and monitor password actions in a complete manner, providing a more efficient and harmonious method of improving security.

3. Research Methodology

The research methods involve a series of systematic steps to advance understanding and application of user authentication algorithms. First, a comprehensive review of recent literature is conducted to grasp the current state of knowledge and to understand the CR algorithm and EPSB. Next, a smart application is developed to re-implement the CR-EPSB algorithm, ensuring it is tailored for modern usage. Following this, the strengths and weaknesses of the existing CR-EPSB algorithm are analyzed to identify areas for improvement. Building on these insights, a new algorithm is created that takes into account the distribution structure of letters and numbers on mobile keypads, aiming to enhance security and efficiency. The smart application is then updated to incorporate this new algorithm, referred to as TKIP-RUB (Transforming Keypad Input Patterns to Recognize User Behavior). The final phase involves rigorous testing and evaluation of both the CR-EPSB and TKIP-RUB algorithms to distinguish authorized users from impostors during password updates, based on patterns from their previous passwords. The proposed methodology ensures a robust approach to improving user authentication systems. Figure 1 illustrates the methodological steps used to achieve the objectives of this study.

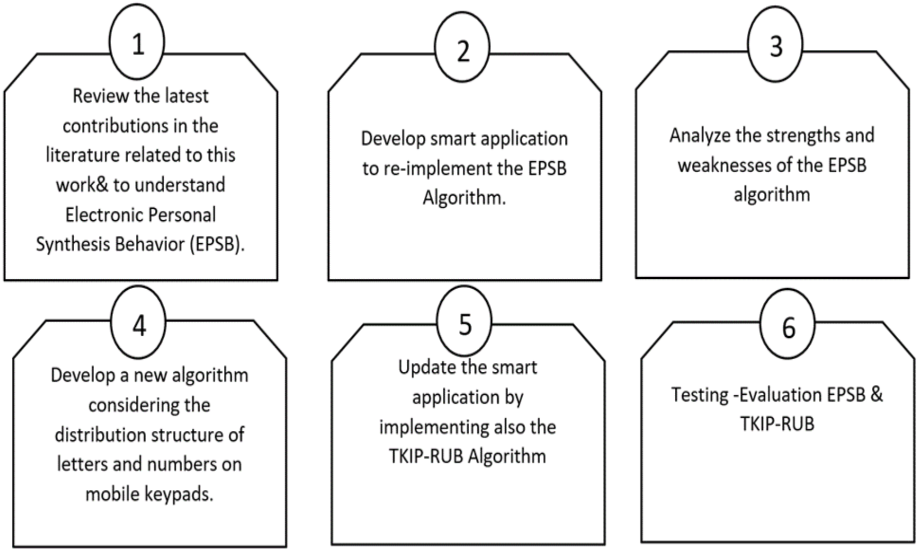

1
Review the latest contributions in the literature related to this work& to understand Electronic Personal Synthesis Behavior (EPSB).
2
Develop smart application to re-implement the EPSB Algorithm.
3
Analyze the strengths and weaknesses of the EPSB algorithm
4
Develop a new algorithm considering the distribution structure of letters and numbers on mobile keypads.
5
Update the smart application by implementing also the TKIP-RUB Algorithm
6
Testing -Evaluation EPSB & TKIP-RUB

Figure 1. Methodology

3.1. EPSB Algorithm

The CR-EPSB Algorithm is revolving around the concept of "Confidence Range", established by the range of minimum and maximum values of factors like mean, median, and mode. Specifically, the proposed algorithm undergoes the following steps when handling the selected password of genuine users:

- The password is analyzed to generate six distinct patterns, encompassing capital letters, lowercase letters, a combination of both, numbers, symbols, and the password's length.
- Each pattern's length is calculated.
- The algorithm determines the minimum and maximum values for mean, median, and mode for each pattern.
- These steps are reiterated whenever genuine users update their passwords.
- A Confidence Range is generated for each user.

Figure 2 illustrates the CR-EPSB Algorithm steps used to recognize the authorized user from unauthorized users.

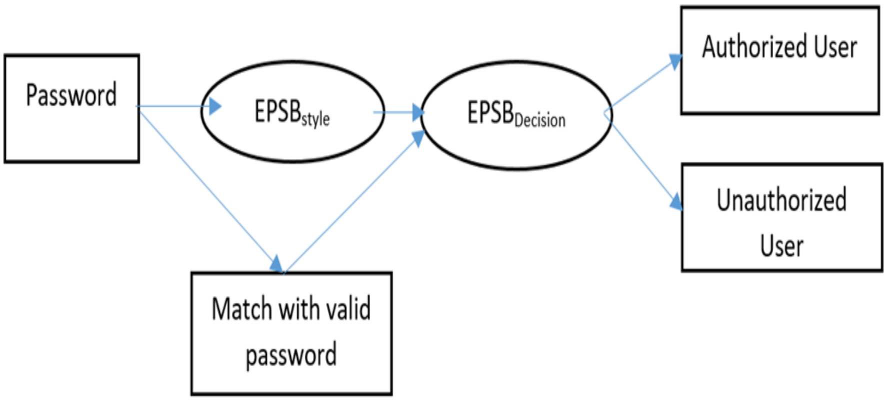


Figure 2. CR- EPSB algorithm [15]

3.2. The Proposed TKIP-RUB Algorithm

Mobile devices often feature a 10-button pad, also known as a "digital" or "12-key pad."

Despite the name, these pads typically contain twelve buttons, facilitating the input of numbers and alphanumeric characters.

While using the mobile keypad to enter numbers is straightforward, letters are also present on most buttons, although their arrangement may differ across regions.

Text input on mobile physical keyboards may require switching between input modes, with each button responsible for typing a sequence of letters, numbers, and symbols. For instance, a button may have multiple alphabetic characters assigned to it, accessed through repeated presses.

Some text input mode may be required to complete individual fields, and the user can select and switch between them. In text mode, each button is responsible for typing a sequence of letters, numbers and symbols, usually not listed on the key. For example, the button has the signs "A", "B", "C" and "2". Thus, after one press, the letter "A" will appear in the field, after two quick presses - "B", after three - "C". This is called a triple press, since the maximum number of alphabetic characters on the button is three.

By pressing the button the fourth time, the number "2" appears.

Thus, pressing the key successively produces the signs "A", "B", "C", "2", "@" and "?", then they are repeated.

Almost all panels and virtual input systems on remote controls and touchscreen smartphones support an input system that does not require multiple presses of a single button. In this case, for each character, the user presses the button only once, and the algorithm substitutes the most probable characters, taking into account other pressed keys. For example, the user presses the buttons "666" for "O", "6" for "M" ,"2: for "A" , "66" for N and as he moves, the letters "O", then "M" ,"A",N", and finally the desired word - "OMAN" are displayed in the input field. Figure 3 and Table 1 present the standard structure and letter distribution across various substitution numbers. They illustrate the frequency and patterns of letter usage in mobile keypad inputs, highlighting the letters most commonly associated with specific replacement numbers. The graphic highlights the unique typing patterns that may be examined to improve mobile authentication systems' security protocols.

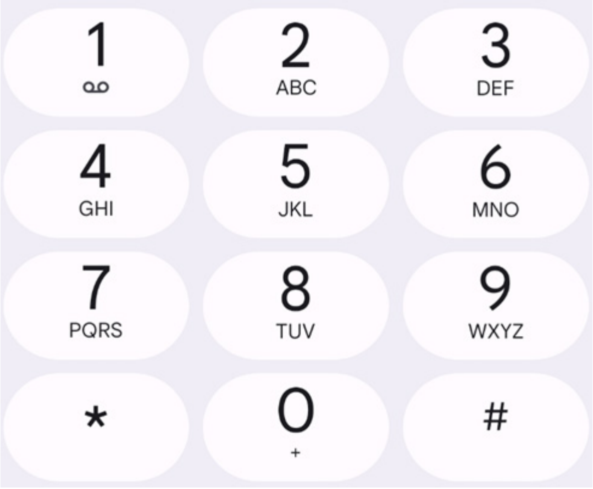


Figure 3. Structure of mobile keypad inputs

3.3. Procedures TKIP-RUB Algorithm

The procedures outlining how the given keypad mappings and algorithm function are used:

- To begin, map every letter of the alphabet to the matching row of digits on a mobile keypad using a dictionary.
- Map Letters to Numbers: Based on conventional mobile keypads, utilize the dictionary to translate each letter of the input text into the corresponding number sequence: “A" corresponds to "2", "B" to "22", and "C" to "222".
- Proceed in the same manner with each letter.
- Input Text Conversion: Using the dictionary, replace each letter in the given input text with the corresponding numeric sequence.
- Create Numeric Sequences: Put the numeric sequences together to create a continuous string of numbers that, in keypad patterns, reflects the input text.
- Utilize Numerical Patterns: Based on the keypad mappings, utilize these numeric sequences for further processing, such as pattern recognition, to evaluate or authenticate user input.

3.4. Experiment and Implementation

In the experiment, a dataset was created from user authentication attempts. It included 143 users who updated their passwords, totaling 629 records (486 in the training dataset and 143 in the test dataset. Table 1 illustrates the code sample representing the mobile keypad array, referred to as the encoding process in this work.

Table 1. Code sample representing the mobile keypad array

```
private Dictionary<char,
string> keypad Mappings
= new Dictionary<char,
string>()
    {
        {'A', "2"}, {'B',
"22"}, {'C', "222"},
        {'D', "3"}, {'E',
"33"}, {'F', "333"},
        {'G', "4"}, {'H',
"44"}, {'I', "444"},
        {'J', "5"}, {'K',
"55"}, {'L', "555"},
        {'M', "6"}, {'N',
"66"}, {'O', "666"},
        {'P', "7"}, {'Q',
"77"}, {'R', "777"}, {'S',
          "7777"},
        {'T', "8"}, {'U',
"88"}, {'V', "888"},
        {'W', "9"}, {'X',
"99"}, {'Y', "999"}, {'Z',
        "9999"}      };
```

The evaluation of the TKIP-RUB and CR-EPSB algorithms focuses on distinguishing legitimate users from imposters during password updates by analyzing patterns derived from their previous passwords.

The dataset spanned from September 2020 to July 2023 and included 143 employees who were active during this period. Table 2 illustrates the distribution of user password update frequency, with 629 records dispersed among the 143 users. For instance, 103 users changed their passwords 4 times, as extracted from 412 records in the security log database. Additionally, 23 users changed their passwords 5 times during this period, as found in 115 records of the security log database. Finally, 17 users changed their passwords 6 times, according to 102 records in the security log database.

Table 2. Distribution of password changes

Number of User	Frequency of Password Changes	Security log Records
103	4	412
23	5	115
17	6	102

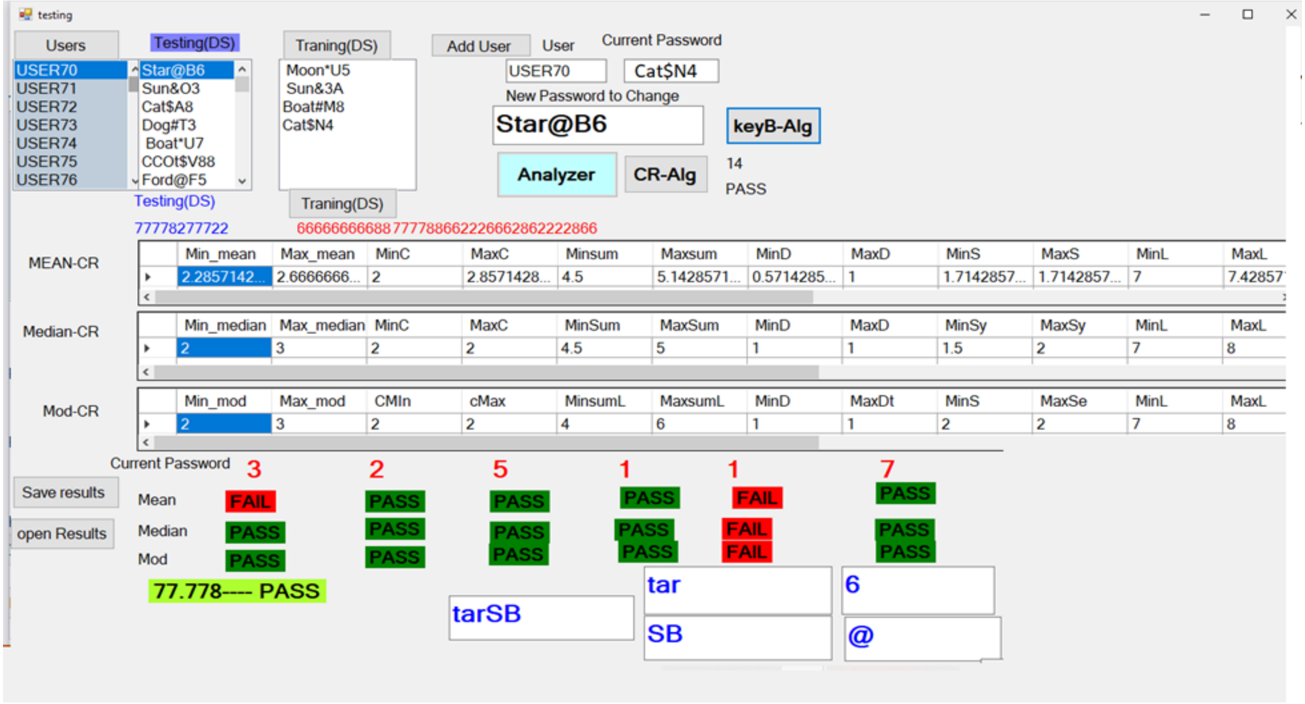
To create the test collection, the most recently updated passwords of 143 users were excluded and stored as a new dataset. The final datasets are as follows:

- **Training Dataset**: 428 records.
- **Testing Dataset**: 143 records.

The dataset was divided into two parts: The training dataset, consisting of 428 records, and the testing dataset, consisting of 143 records. The training dataset was uploaded to the server and used to submit all user records to the developed smart application. This application generates a user pattern profile for each individual based on the following password analysis algorithm:

- Process all user passwords from the training dataset.
- Normalize each password by removing numbers and symbols.
- Convert all letters to uppercase.
- Use the substitution matrix (refer to Table 1 for the mobile keypad array) to map each letter to its corresponding numerical pattern.
- Store the resulting pattern for each user.
- Update the user's variable by appending the new series number generated in step 5.
- Repeat steps 1 through 6 for each user until all 143 users are processed.

Figure 4 illustrates a sample implementation for User 70. The training dataset was utilized in the initial experiment to generate unique user patterns based on their historical input passwords. The EPSB and TKIP-RUB algorithms were also implemented in the code. This is the way the application works: User 70 updated their password five times. Initially, User 70 used the password "Moon*U5," which was then updated to "Sun@3A," followed by "Boat#M8,", then "Oog#ABC" and later "Cat$N4." These passwords were included in the training dataset. The most recent password, "Star@B6," was excluded from the training dataset and used for testing.



Users: USER70, USER71, USER72, USER73, USER74, USER75, USER76

Testing(DS): Star@B6, Sun&O3, Cat$A8, Dog#T3, Boat*U7, CCOt$V88, Ford@F5

Traning(DS): Moon*U5, Sun&3A, Boat#M8, Cat$N4

User: USER70

Current Password: Cat$N4

New Password to Change: Star@B6

14
PASS

Testing(DS)
77778277722

66666666688777788662226662862222866

MEAN-CR

Min_mean	Max_mean	MinC	MaxC	Minsum	Maxsum	MinD	MaxD	MinS	MaxS	MinL	MaxL
2.2857142...	2.6666666...	2	2.8571428...	4.5	5.1428571...	0.5714285...	1	1.7142857...	1.7142857...	7	7.42857

Median-CR

Min_median	Max_median	MinC	MaxC	MinSum	MaxSum	MinD	MaxD	MinSy	MaxSy	MinL	MaxL
2	3	2	2	4.5	5	1	1	1.5	2	7	8

Mod-CR

Min_mod	Max_mod	CMIn	cMax	MinsumL	MaxsumL	MinD	MaxDt	MinS	MaxSe	MinL	MaxL
2	3	2	2	4	6	1	1	2	2	7	8

Current Password	3	2	5	1	1	7
Mean	FAIL	PASS	PASS	PASS	FAIL	PASS
Median	PASS	PASS	PASS	PASS	FAIL	PASS
Mod	PASS	PASS	PASS	PASS	FAIL	PASS

77.778---- PASS

tarSB

tar

SB

6

@

Figure 4. Screenshoot of developed application for analyzing User 70

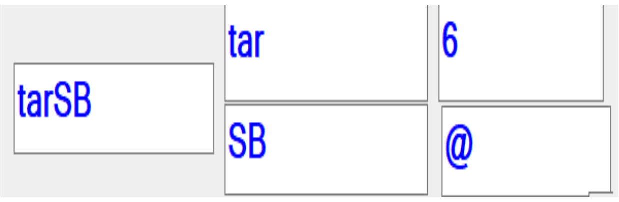
Scenario: The developed application calculates the patterns generated by the EPSB and TKIP-RUB algorithms for User 70. When User 70 enters the password "Star@B6"—which has not been seen before, the application must determine whether this password belongs to an authorized user or an unauthorized user. The goal is to test which algorithm is more effective at recognizing if the entered password is associated with the authorized user. The EPSB calculation for User 70 is as follows: Figure 4 shows that the application starts by allowing the user to enter their username and current password.

For this example, User 70 inputs "USR70" as the username and "Oog#ABC" as the current password.

User 70 then attempts to update their password to "Star@B6." After clicking the "Analyze" button, Figure 5 illustrates the generated EPSB styles for User 70's password "Star@B6," segmented into distinct categories: lowercase letters, uppercase letters, a combination of both, numbers, symbols, and the overall length of the password. Figure 6 displays User 70's behavior profile, including the "MEAN-CR," "Median-CR," and "Mod-CR" values.

The application shows the minimum, and maximum values for each factor based on the six styles of the EPSB algorithm. Table 3 provides an overview of the various types of EPSB styles. The proposed application calculates the occurrence of each of the six styles found in "Star@B6."

Specifically, it identifies small letters "tar" = 3, capital letters "SB" = 2, a mix of capital and small letters "STARB" = 5, numbers "6" = 1, symbols "@" = 1, and the total length of the password = 7. When the "CR-Alg" button is clicked, the EPSB algorithm is applied, and the accumulated calculations for User 70 are displayed, as shown in Figure 5, which is extracted from Figure 4.



tarSB

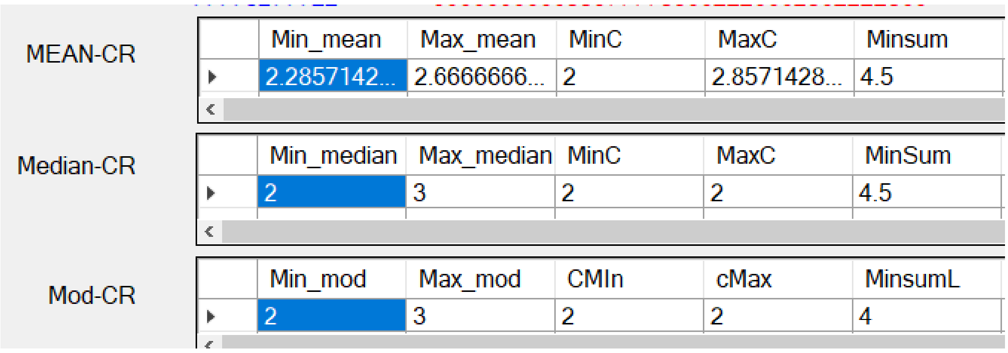
tar

6

SB

@

Figure 5. EPSB styles for User 70 for Password "Star@B6"



MEAN-CR

	Min_mean	Max_mean	MinC	MaxC	Minsum
▸	2.2857142...	2.6666666...	2	2.8571428...	4.5

Median-CR

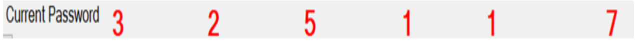
	Min_median	Max_median	MinC	MaxC	MinSum
▸	2	3	2	2	4.5

Mod-CR

	Min_mod	Max_mod	CMIn	cMax	MinsumL
▸	2	3	2	2	4

Figure 6. Calculation of EPSB styles (User 70's behavior profile)

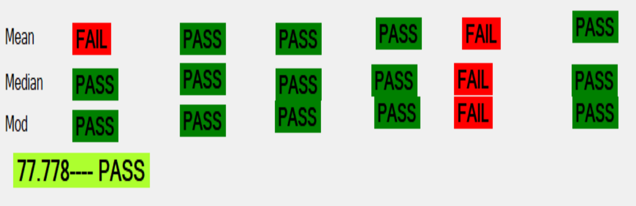
The next step the application will also calculate the CR-EPSB for the "Star@B6" password entered for renewing the current password, for this password the CR-EPSB is presented in Figure 7.



Current Password 3 2 5 1 1 7

Figure 7. CR-EPSB for "Star@B6" password

Now the application will compare the accumulated previous CR of User 70, as shown in Figure 6, with the CR of "Star@B6" presented in Figure 7. The final step is for the application to display the results, as shown in Figure 8.



Mean	FAIL	PASS	PASS	PASS	FAIL	PASS
Median	PASS	PASS	PASS	PASS	FAIL	PASS
Mod	PASS	PASS	PASS	PASS	FAIL	PASS

77.778---- PASS

Figure 8. Results of CR-EPSB User 70 and CR of Star@B6"

In the Mean measure, the system shows a 'Fail' status for the comparison of small letters. The CR of the user profile for small letters has a minimum of 2.857 and a maximum of 2.666. For "Star@B6" to match this indicator, the CR for the small letter mean indicator should be within the range of 2.857 to 2.666. The algorithm shows a 'Fail' status because the CR for small letters of "Star@B6" is 3, which is not within the range of [2.857 to 2.666]. The Capital Letters, Sum of Small Letters + Capital Letters, and Numerals indicators were matching, resulting in a 'PASS'.

The symbols indicator did not match, resulting in a 'Fail', and the final parameter length matched, resulting in a 'PASS'. The same matching process is calculated for the Median and Mode indicators. Table 4 demonstrates the comparison by the Mean indicator, Table 5 demonstrates the comparison by the Median indicator, and Table 6 demonstrates the comparison by the Mode indicator.

3.5. Running the TKIP-RUB algorithms

When the user clicks the "KeyB-Alg" button in Figure 4, the application will be transforming mobile keypad Input patterns to recognize user behavior as illustrated in Table 7. For example, "Moon*U5" is transformed into the sequence 666666088, where M is converted to 666, O to 666, N to 66, and U to 88. All numbers and symbols are ignored. This process is applied to all other passwords in the same manner.

To compare the accumulated sequence "66666887777886622266628622222866" (User 70) with the accumulated sequence "77778277722".

For "Star@B6," a similarity algorithm, based on the Longest Common Subsequence (LCS) method [36], is applied to identify 14 matching patterns between the sequences. This similarity surpasses the threshold for the similarity indicator, resulting in a "PASS" outcome. Figure 9 provides additional details about the developed application. In Figure 9, the application evaluates and analyzes User 71. The CR-EPSB algorithm failed to recognize User 71's entry, as they entered password "Sun&O3" was associated with User 71. However, the proposed TKIP-RUB algorithm successfully identified the entry, finding 18 similar patterns between User 71's profile and the entered password from the test data.

Table 3. Six parameters integrated with the EPSB algorithm

Parameter1	**Parameter2**	**Parameter3**	**Parameter4**	**Parameter5**	**Parameter6**
Small letters	Capital Letters	Sum of Small letters + Capital Letters	Numerals	Symbols	Length of the password

Table 4. Comparison by mean indicator

CR	**Parameter1**		**Parameter2**		**Parameter3**		**Parameter4**		**Parameter5**		**Parameter6**	
	Small letters		Capital Letters		Sum of Small letters + Capital Letters		Numerals		Symbols		Length of the password	
	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max
Mean for „User 70“	2.28	2.66	2.0	2.85	4.5	5.14	0.57	1.0	1.71	1.71	7.0	7.42
Mean for “Star@B6"	3.0	3.0	2.0	2.0	5.0	5.0	1.0	1.0	1.0	1.0	7.0	7.0
Results	Fail		Pass		Pass		Pass		Fail		Pass	

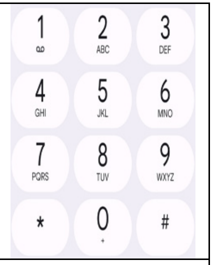
Table 5. Comparison by median indicator

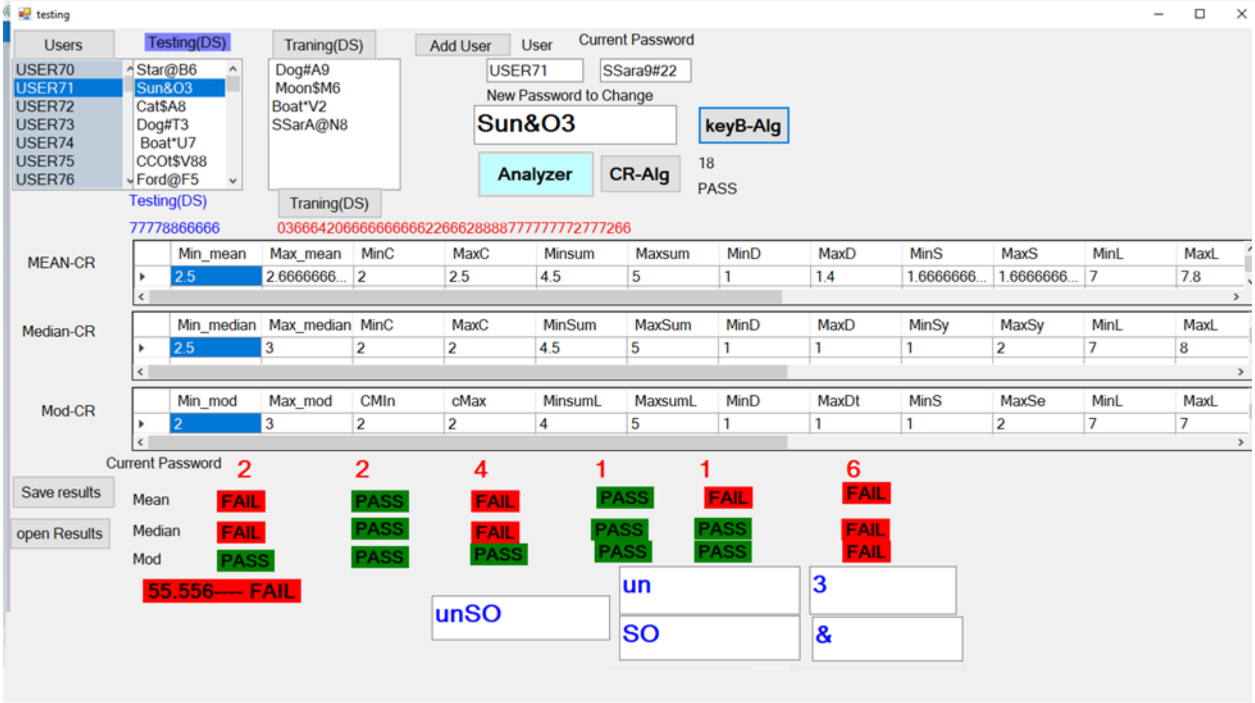
CR	**Parameter1**		**Parameter2**		**Parameter3**		**Parameter4**		**Parameter5**		**Parameter6**	
	Small letters		Capital Letters		Sum of Small letters + Capital Letters		Numerals		Symbols		Length of the password	
	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max
Median for „User 70“	2	3	2.0	2	4.5	5	1	1.0	1.5	2	7.0	8
Median for “Star@B6"	3.0	3.0	2.0	2.0	5.0	5.0	1.0	1.0	1.0	1.0	7.0	7.0
Results	Pass		Pass		Pass		Pass		Fail		Pass	

Table 6. Comparison by mode indicator

CR	**Parameter1**		**Parameter2**		**Parameter3**		**Parameter4**		**Parameter5**		**Parameter6**	
	Small letters		Capital Letters		Sum of Small letters + Capital Letters		Numerals		Symbols		Length of the password	
	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max
Mod for „User 70“	2	3	2.0	2	4	6	1	1	2	2	7.0	8
Mod for “Star@B6"	3.0	3.0	2.0	2.0	5.0	5.0	1.0	1.0	1.0	1.0	7.0	7.0
Results	Pass		Pass		Pass		Pass		Fail		Pass	

Table 7. TKIP-RUB algorithms (analyzing User 70 passwords)

Password	**Transforming Mobile Keypad Input**	**Accumulated sequence (User 70)**	
Moon*U5	66666688	66666688777788662226662862222866	 1 ∞ 2 ABC 3 DEF 4 GHI 5 JKL 6 MNO 7 PQRS 8 TUV 9 WXYZ * 0 + #
Sun@3A	777788662		
Boat#M8	22666286		
Cat$N4	2222866		
		Accumulated sequence “Star@B6"	
Star@B6	77778277722	77778277722	



Users: USER70, USER71, USER72, USER73, USER74, USER75, USER76

Testing(DS): Star@B6, Sun&O3, Cat$A8, Dog#T3, Boat*U7, CCOt$V88, Ford@F5

Traning(DS): Dog#A9, Moon$M6, Boat*V2, SSarA@N8

User: USER71

Current Password: SSara9#22

New Password to Change: Sun&O3

18
PASS

Testing(DS): 77778866666

036664206666666666622666288887777777772777266

MEAN-CR

Min_mean	Max_mean	MinC	MaxC	Minsum	Maxsum	MinD	MaxD	MinS	MaxS	MinL	MaxL
2.5	2.6666666...	2	2.5	4.5	5	1	1.4	1.6666666...	1.6666666...	7	7.8

Median-CR

Min_median	Max_median	MinC	MaxC	MinSum	MaxSum	MinD	MaxD	MinSy	MaxSy	MinL	MaxL
2.5	3	2	2	4.5	5	1	1	1	2	7	8

Mod-CR

Min_mod	Max_mod	CMIn	cMax	MinsumL	MaxsumL	MinD	MaxDt	MinS	MaxSe	MinL	MaxL
2	3	2	2	4	5	1	1	1	2	7	7

Current Password	2	2	4	1	1	6
Mean	FAIL	PASS	FAIL	PASS	FAIL	FAIL
Median	FAIL	PASS	FAIL	PASS	PASS	FAIL
Mod	PASS	PASS	PASS	PASS	PASS	FAIL

55.556— FAIL

unSO

un

SO

3

&

Figure 9. Screen shoot for analyzing and testing process related to user 71

4. Evaluation

The effectiveness of the EPSB algorithm from [16] and the proposed TKIP-RUB algorithm was evaluated using the precision metric, defined as follows:

Precision (P) measures the proportion of correctly identified passwords that are considered successful.

Precision (P) = #(Relevant Passwords Matched [Pass]) / #(Total Test Items). Table 8 explains the precision calculation concept.

Table 8. Precision (P) in terms of TP, FP

Matched Password	Not Matched Password
Pass	True Positives (TP)
Fail	False Negatives (FN)

Where:

Relevant Passwords = number of all matched passwords = 143 for 143 users, P = TP / (TP + FP).

For the experiment, the test dataset containing 143 users was uploaded to the developed smart security application. This allowed us to compute the CR-EPSB algorithm from [20] and the TKIP-RUB algorithm proposed in this work, for all 143 users with 143 testing records. The performance metrics for two algorithms—TKIP-RUB and CR-EPSB—in terms of accuracy, false positives, and true positives are displayed in the Table 9.

The accuracy rate of the CR-EPSB Algorithm is 9.091%. This means that 9.091% of the positive predictions were accurate.

At 53.147%, the TKIP-RUB algorithm has a substantially better accuracy percentage. This indicates that 53.147% of the positive predictions were accurate.

Table 9. Precision for CR-EPSB and TKIP-RUB

Algorithms	TP	FP	TP +FP	TP/(TP + FP)	%
CR-EPSB	13	130	143	0.091	9.091
TKIP-RUB	76	67	143	0.531	53.147

The improvement percentage can be calculated using the formula:

$$\frac{P(TKIP-RUB\,)-P(EPSB)}{P(EPSB)} * 100 \tag{1}$$

$$= \frac{53.147-9.091}{9.091} * 100 = 484.615\%$$

Therefore, TKIP-RUB shows an improvement in recognition rate of approximately 484.615% compared to CR-EPSB algorithm.

The TKIP-RUB method is clearly superior to the CR-EPSB, as seen by the roughly 484.58% accuracy gain between the two algorithms. This indicates that the TKIP-RUB algorithm's accuracy is nearly five times greater than the original algorithm, demonstrating a significant improvement in performance.

The objective was to strengthen security measures by leveraging user-specific patterns derived from password histories.

This was achieved by analyzing and extracting significant patterns from user passwords, translating them into numeric sequences entered via a mobile keypad. Through experimentation, the effectiveness of the proposed approach in mitigating the risk of unauthorized access is demonstrated. Notably, this method aligns with real-world security policies, ensuring practical applicability. The results highlight the robustness of the approach in recognizing user patterns and fortifying system integrity. By seamlessly integrating keypad-input numerical patterns into existing authentication systems, this work contributes to the advancement of security protocols, offering a reliable solution for safeguarding user privacy and preventing unauthorized access.

5. Conclusion

The obtained results show that the precision of the TKIP-RUB algorithm is significantly higher than that of the previous method. More specifically, there was an approximate five times improvement in accuracy. This substantial increase suggests that, in comparison to the EPSB method, the TKIP-RUB algorithm is far more successful in accurately predicting positive cases. This paper has successfully presented a novel approach to enhancing security authentication and integration by transforming user password patterns into keypad-input numerical sequences.

However, the test collection results highlight the weaknesses of CR-EPSB in distinguishing genuine users from false ones, EPSB failed to recognize genuine users from false ones achieving a recognition rate of 90%. In contrast, TKIP-RUB failed to recognize genuine users from false ones achieving a recognition rate of 45%.

References:

[1]. Shakir, M., et al. (2024). The Influence of Mobile Information Systems Implementation on Enhancing Human Resource Performance Skills: An Applied Study in a Small Organization. *International Journal of Interactive Mobile Technologies*, *18*(13). Doi: 10.3991/ijim.v18i13.47027.

[2]. Shannaq, B. (2024). Unveiling the Nexus: Exploring TAM Components Influencing Professors' Satisfaction With Smartphone Integration in Lectures: A Case Study From Oman. *TEM Journal*, *13*(3), 2365-2375. Doi: 10.18421/TEM133-63.

[3]. Ezugwu, A., et al. (2023). Password-based authentication and the experiences of end users. *Scientific African*, *21*, e01743. Doi: 10.1016/j.sciaf.2023.e01743.

[4]. Mostafa, A. M., et al. (2023). Strengthening cloud security: an innovative multi-factor multi-layer authentication framework for cloud user authentication. *Applied Sciences*, *13*(19), 10871. Doi: 10.3390/app131910871.

[5]. Teodorescu, C. A., Durnoi, A. N. C., & Vargas, V. M. (2023). The Rise of the Mobile Internet: Tracing the Evolution of Portable Devices. *Proceedings of the International Conference on Business Excellence*, *17*(1), 1645–1654. Doi: 10.2478/picbe-2023-0147.

[6]. Shuwandy, M. L., et al. (2024). Sensor-Based Authentication in Smartphone; a Systematic Review. *Journal of Engineering Research*. Doi: 10.1016/j.jer.2024.02.003.

[7]. Tsoukas, V., Gkogkidis, A., & Kakarountas, A. (2020). A Survey on Mobile User Perceptions of Sensitive Data and Authentication Methods. *Proceedings of the 24th Pan-Hellenic Conference on Informatics,* 346–349. Doi: 10.1145/3437120.3437337.

[8]. Aslan, Ö., et al. (2023). A comprehensive review of cyber security vulnerabilities, threats, attacks, and solutions. *Electronics*, *12*(6), 1333. Doi: 10.3390/electronics12061333.

[9]. Atoum, I. A., & Keshta, I. M. (2021). Big data management: Security and privacy concerns. *International Journal of Advanced and Applied Sciences*, *8*(5), 73-83. Doi: 10.21833/ijaas.2021.05.009.

[10]. Yıldırım, M., & Mackie, I. (2019). Encouraging users to improve password security and memorability. *International Journal of Information Security*, *18*, 741-759. Doi: 10.1007/s10207-019-00429-y.

[11]. George, A. S. (2024). Bridging the Digital Divide: Understanding the Human Impacts of Digital Transformation. *Partners Universal International Innovation Journal (PUIIJ),* 2(3), 1–34. Doi: 10.5281/zenodo.11287684.

[12]. Khan, M. A., Din, I. U., & Almogren, A. (2023). Securing access to internet of medical things using a graphical-password-based user authentication scheme. *Sustainability*, *15*(6), 5207. Doi: 10.3390/su15065207.

[13]. Mahfouz, A., et al. (2024). B2auth: A contextual fine-grained behavioral biometric authentication framework for real-world deployment. *Pervasive and Mobile Computing*, *99*, 101888. Doi: 10.1016/j.pmcj.2024.101888.

[14]. Jetpack. (2024). *How Weak Passwords Expose You to Serious Security Risks.* Jetpack. Retrieved from: https://jetpack.com/blog/weak-passwords/ [accessed: 10 July 2024].

[15]. Shakir, M., et al. (2016). Application of confidence range algorithm in recognizing user behavior through EPSB in cloud computing. *Journal of Theoretical and Applied Information Technology*, *94*, 416-427.

[16]. Albattah, W. (2018). Analysis of passwords: Towards understanding of strengths and weaknesses. *International Journal of Advanced And Applied Sciences*, *5*(11), 51-60. Doi: 10.21833/ijaas.2018.11.007

[17]. Shannaq, B. (2024). Improving security in intelligent systems: how effective are machine learning models with tf-idf vectorization for password-based user classification. *Journal of Theoretical and Applied Information Technology*, *102*(22).

[18]. Mobiversal. (2021). *The Impact of Mobile Technology in Our Lives.* Blog.mobiversal. Retrieved from: https://blog.mobiversal.com/the-impact-of-mobile-technology-in-our-daily-life.html [accessed: 12 July 2024].

[19]. Gladden, D. (2024). *The Effects of Smartphones on Social Lives: How They Affect Our Social Interactions and Attitudes*. ODU Digital Commons. Retrieved from: https://digitalcommons.odu.edu/ots_masters_projects/586/ [accessed: 13 July 2024].

[20]. Arai, K. (2022). *Advances in Information and Communication: Proceedings of the 2022 Future of Information and Communication Conference (FICC), Volume 2*. Springer Nature.

[21]. Szyjewski, G., & Fabisiak, L. (2018). A study on existing and actually used capabilities of mobile phones technologies. *Procedia Computer Science*, *126*, 1627-1636. Doi: 10.1016/j.procs.2018.08.136.

[22]. Future Today Institute. (2024). *2024 Tech Trends Report (*17th ed.). Future Today Institute. Retrieved from: https://futuretodayinstitute.com/wp-content/uploads/2024/03/TR2024_Full-Report_FINAL_LINKED.pdf [accessed: 16 July 2024].

[23]. Eldow, A., et al. (2019). Literature review of authentication layer for public cloud computing: a meta-analysis. *ARPN Journal of Theoretical and Applied Information Technology*. *14*(10).

[24]. Shakir, M., et al. (2021). Users Acceptance of Electronic Personal Synthesis Behavior (EPSB): An Exploratory Study. *Recent Advances in Technology Acceptance Models and Theories*, 509-520. Doi: 10.1007/978-3-030-64987-6_30.

[25]. Shakir, M., Hammood, M., & Muttar, A. K. (2018). Literature review of security issues in saas for public cloud computing: a meta-analysis. *International Journal of Engineering & Technology*, *7*(3), 1161-1171. Doi: 10.14419/ijet.v7i3.13075.

[26]. Cypress Data Defense. (2020). *6 Password Security Risks and How to Avoid Them.* Cypress Data Defense. Retrieved from: https://www.cypressdatadefense.com/blog/password-security-risks/ [accessed: 17 July 2024].

[27]. Jithukrishnan. (2022). *Top 10 password policy recommendations for system administrators in 2023.* Securden. Retrieved from: https://www.securden.com/blog/top-10-password-policies.html [accessed: 18 July 2024].

[28]. Shannaq, B., & Shakir, M. T. (2024). Enhancing Security through Multi-Factor User Behavior Identification: Moving Beyond the Use of the Longest Common Subsequence (LCS). *Informatica*, *48*(19).