Dynamic Memorable Password Generation Algorithm (DyMP-Gen)

Baker Al Smadi
Computer Science and Digital Technologies
Grambling State University
Grambling, LA, USA
Bakir_smadi@hotmail.com

Dileon Saint-Jean
Computer Science and Data Science
American Public University System
Computer Science and Digital Tech
Grambling State University
Grambling, LA, USA
dileon.saintjean@gmail.com

Majd Tahat
Cyberspace Engineering
Louisiana Tech University
Ruston, LA, USA
Latech.tht@gmail.com

Hosam Alamleh
Computer Science
University of North Carolina Wilmington
Wilmington, North Carolina
hosam.amleh@gmail.com

Syed Raza
Computer Science and Data Science
American Public University System
Charles Town, West Virginia
syed.raza@mycampus.apus.edu

Ali Abdullah S. AlQahtani
Computer Systems Technology
North Carolina A&T State University
Greensboro, North Carolina
alqahtani.aasa@gmail.com

***Abstract*—The increasing frequency and sophistication of cyberattacks necessitate a reevaluation of traditional password practices. This paper introduces DyMP-Gen (Dynamic Memorable Password Generation) Algorithm, a novel algorithm designed to enhance password security while prioritizing usability. DyMP-Gen allows users to generate strong, unique passwords that evolve over time, mitigating the risks associated with static, password reuse, and leaked passwords. By combining a user pre-selected static component with dynamically generated elements, a password based on current time and date is created. DyMP-Gen ensures that passwords are both memorable and resilient to cracking attempts. The algorithm's simplicity and offline functionality further enhance its usability, eliminating the need for additional devices or software. Through its innovative approach, DyMP-Gen offers a promising solution to the pervasive challenges of password security.**

***Index Terms*—Passwords, Authentication, Dynamic, Security, Password Generation**

I. Introduction

Passwords have been in existence for a while now, with each advancement in technology bringing a new, more complex requirement for passwords. The rapid expansion of web-based technologies has caused a significant increase in the number of accounts a typical user has. A global study by NordPass indicated that the average user has approximately 100 passwords [1], [7]. Password and account breaches have similarly skyrocketed over the last decade. Reports indicate that almost 1 in 4 individuals were affected by a data breach in the last 18 months [2], [11]. Additionally, more than 60% of people admit to reusing passwords and more than 1 in 4 indicate that their oldest password is on average 4 years old [3], [4], [6]. A study by security.org shows that more than 66% of people use the same password across multiple accounts, and of those, about half share passwords with others [5], [12]. The same study shows that users typically use relatively unsafe methods for creating passwords. 57% of those surveyed indicate that making only slight changes to old passwords was how their new passwords are created. (See Figure 1)

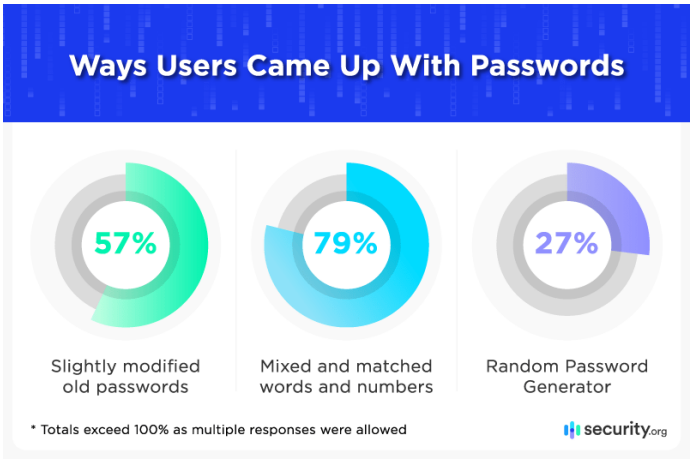

Fig. 1. Ways Users Came up with Passwords [5]

While users may not be able to protect against data breaches that affect service providers, better password habits would help prevent unauthorized access to accounts as passwords are the first line of defense to most modern services. Many prior articles have investigated how people perceive data breaches and the need to change passwords, along with their responses to such events. Individuals are also likely to slightly modify old passwords following a breach [6], [7], [13]. A study at a university by Usenix revealed that by tweaking leaked password and username combinations over a 10-year period, 32% of new accounts were compromised, despite the complexity

979-8-3315-2096-0/25/$31.00 ©2025 IEEE

required of new passwords [8]. A study by NordPass also indicated that with US and UK users, more than 30% of them consider resetting or changing passwords to be as stressful as retiring [9]. Similarly, a survey by YouGov found that 28% of US adults use the same password for all their accounts [10]. In services that do not enforce the use of MFA, a reimagination of the way passwords are created and used is necessary. In this work, we present a novel method for the creation and protection of passwords. The Dynamic Memorable Password Generation algorithm (DyMP-Gen) is capable of providing updated passwords over time, with little to no input from a user. This algorithm allows for the creation of passwords that meet four of the common characteristics of strong passwords: lengthy, complex, unique and difficult to guess, leading to a very high entropy password that is easy for a user to remember and use. DyMP-Gen is also variable and depends on multiple factors that vary to confirm authentication. This algorithm is further explored in the following sections.

II. Related Work

The prevalence of cyber attacks and the ensuing data breaches underscore the importance of password security [14]. Passwords acting as the first line of defense for most services are susceptible to various attack vectors like brute force attacks, dictionary attacks, and probabilistic context free grammar attacks [16]. The challenge is not only in the complexity of these attacks, but also in the inherent difficulties that users face in creating and managing strong passwords [17].

Studies have shown that user often resort to insecure practices such as password reuse, and usage of easy to guess passwords due to the burden of having to remember too many complex passwords [18]–[20]. This literature review evaluates the use of password generation and management, examining existing techniques, their limitations, and the potential of dynamic and memorable password generation to enhance overall password security. It will analyze prior research, identify prevailing gaps, and present our novel method, DyMP-Gen, which seeks to redefine the way passwords are created and utilized.

A. User Behavior and Password Practices

Evidence consistently reveals a very concerning trend in user password practices. Studies show a widespread prevalence of weak and easy to guess passwords, that will typically use personal information, dictionary words and simple patterns [7], [13], [21]. Tanni et al. (2022) conducted a study and found that a significant portion of participants (56%) included personal information in their passwords, while 69% reused passwords across multiple accounts. These findings are consistent with prior studies highlighting the prevalence of this issue.

Several factors contribute to these insecure practices. The foremost challenge lies in the inherent difficulty of remembering complex passwords, especially as the number of online accounts per user continues to grow [21]. Users often prioritize memorability over security, opting for easily recallable passwords that are susceptible to cracking attempts. Additionally, the cognitive burden of managing numerous unique passwords can lead to password fatigue, further incentivizing the reuse of passwords or the creation of easily guessed variations [21].

These user-centric challenges underscore the need for password generation methods that not only prioritize security but also address the usability and memorability concerns of users. The ideal solution would empower users to create strong, unique passwords for each account without compromising their ability to recall them effortlessly. By aligning password generation with human cognitive capabilities, we can encourage the adoption of secure practices and contribute to a more resilient cybersecurity environment.

B. Traditional Password Generation Methods

Traditional password generation methods, while serving as the foundation for authentication systems, often have the challenge of balancing security and usability. Random password generators, for instance, excel in producing high-entropy passwords that are inherently resistant to cracking attempts. However, their reliance on arbitrary character combinations renders them notoriously difficult for users to memorize, leading to the adoption of insecure practices like writing down passwords or reusing them across multiple accounts [23].

On the other hand, mnemonic-based approaches, like those explored by Song et al. (2019), attempt to enhance memorability by associating passwords with meaningful phrases or patterns. While these methods can improve user recall, they may introduce vulnerabilities if the underlying mnemonics are easy to guess or predictable. The effectiveness of mnemonic strategies is heavily contingent on the complexity and uniqueness of the chosen mnemonics, placing an additional cognitive burden on users, thereby complicating the process.

Password cracking techniques are ever evolving; thus, it necessitates the development of password generation methods that can withstand these advanced threats [23]. Traditional methods, with their static and often predictable nature, may fall short in providing adequate protection against such attacks.

The limitations of traditional password generation methods highlight the pressing need for more innovative and adaptive approaches. The concept of dynamic and memorable password generation offers a potential solution. By generating passwords that evolve over time and are inherently easy to recall, this approach aims to bridge the gap between security and usability, empowering users to create and manage strong passwords effortlessly.

C. Password Cracking Techniques

The way that passwords are cracked has evolved over time, with more complex methods being used. Brute-force attacks, which systematically try all possible combinations of characters, remain a prevalent threat, especially for shorter or less complex passwords [23]. Dictionary attacks, leveraging pre-compiled lists of common words and phrases, exploit the tendency of users to choose easily guessable passwords [22].

More recently, probabilistic context-free grammar (PCFG) attacks have gained prominence, utilizing statistical models to generate password guesses based on patterns and structures observed in real-world password leaks [24]. Zhang et al. (2023) highlight the effectiveness of PCFG-based attacks, particularly in targeted scenarios where attackers possess additional information about the user, such as leaked passwords from other services. The ability of these attacks to exploit patterns in human-generated passwords underscores the need for password generation methods that can produce unpredictable and complex combinations.

As password cracking techniques become more complex, the need for strong and resilient password generation algorithms that can withstand these attacks become more important. The adversarial perspective must be considered when designing such algorithms, anticipating potential attack vectors and incorporating the means to mitigate them, and present a more secure environment.

Credit card fraud and password cracking are interconnected threats that undermine the security of online transactions and user authentication systems. Credit card fraud, as highlighted by Smadi et al. [29], exploits vulnerabilities in payment systems, prompting the need for secure and fraud-proof online payment frameworks using advanced cryptographic measures and real-time verification. Similarly, password cracking techniques, such as brute force and dictionary attacks, target weak or reused credentials, granting unauthorized access to sensitive financial data. Smadi et al. [30] further address these risks by proposing a fraud detection framework that identifies anomalies across diverse consumer behaviors using machine learning, showcasing how adaptive security strategies can combat evolving threats. Together, robust payment systems and secure authentication practices are essential to safeguard financial transactions from fraud and unauthorized access.

D. Dynamic and Memorable Password Generation

Due to the increased number and velocity of threats, along with user challenges in creating passwords, dynamic password generation presents a method for increasing the security of password policies, and passwords in general. It transcends the limits of current traditional methods, by creating passwords that evolve over time, thereby mitigating the risks associated with static passwords, but still remaining easy to recall. Such an approach will promote user adoption and compliance.

Several researchers have explored the feasibility of dynamic password generation. The DyMP-Gen algorithm, proposed in this paper, exemplifies this approach by creating passwords that are both strong and memorable through a combination of user-selected static components and dynamically generated numeric elements based on multiple variables. Other notable approaches include PassImg, which leverages image hashing and metadata to generate consistent passwords across devices without server-side storage [25], and PassMan, which employs a similar concept but relies on user-defined phrases and hints for password generation [26]. Tremblay-Savard et al. (2019) also proposed a unique algorithm that generates strong passwords based on user-provided information, aiming to enhance memorability [27]. Additionally, Sameer et al. (2024) introduced an innovative approach using LoRA fine-tuned Large Language Models (LLMs) to generate natural language-like strong passwords, further pushing the boundaries of memorability and security [28].

III. The Proposed System

This paper proposed a system that maintains dynamically updated passwords overtime. The main goal is to overcome the user's weak passwords and having the same password for all user's accounts. The proposed system will relatively enhance the security of users password compared with the regular passwords system. This system aims to minimize the user's interaction with any external devices or software to generate his/her password.

IV. Methodology

This section dives deeper into the proposed algorithm. The user's password is formed by combining three parts:

1) **Static part:** An alphabetic word chosen by the user, containing at least one uppercase letter and four or more lowercase letters.
2) **Calculated number:** A simple number generated using a pre-defined formula based on multiple variables.
3) **Concatenated variables:** The result of a simple arithmetic operation using various variables.

Several variables are used to generate the user's password, including:

1) **Current time:** This is broken down into two sub-variables:
   - Hours
   - Minutes
2) **Date:** This is broken down into three sub-variables:
   - Days
   - Months
   - Years
3) **Generated variables:** Based on the current time and date. See the following tables for detailed information.

TABLE I
Interval Variables

Minute Intervals (MI)	Values	Hour Intervals (HI)	Values	Days Number (DN)	Values
1–5	1	1–3	1	1–10	5
6–10	2	3–6	2	11–20	15
11–15	3	6–9	3	21–31	25
16–20	4	9–12	4		
21–25	5				
26–30	6				
31–35	7				
36–40	8				
41–45	9				
46–50	10				
51–55	11				
56–60	12				

The user should be able to tell his/her password based on the pre-selected static word that involves an uppercase letter and at least four other lowercase letters combined with the dynamic numbers based on the above variables.

TABLE II
BINARY VARIABLES

Day or Night Binary (DNB)	Values	Months or Days Binary (MB or DB)	Values
AM	1	ODD	1=$
PM	2	EVEN	2=#

The user should go through a short survey that generates the user's simple formula to calculate the 2nd part of the password. This formula should maintain a unique password at least every 5 minutes. The generated password ensures that the user's password is updated in the server's database. The user's password can be generated offline and without even the need for any electronic devices due to the low complexity and simplicity.

The user has to set up the formula and remember the generated formula instead of remembering the full password. The formula will be generated by the user through a short survey as follows:

1) Set up the static alphabetic password
2) Select the desired (easy to remember) variables to use to generate the formula.
3) Concatenate the static password with the generated number through the created formula.
4) Match the calculated password with the generated password on the server side to grant access to the user's account.

During the setup phase, the user must select the appropriate time zone to ensure synchronization between the server and the user's system, preventing any discrepancies. This step allows both the server and the user to generate identical passwords, regardless of time zone differences. (See Figure 2)

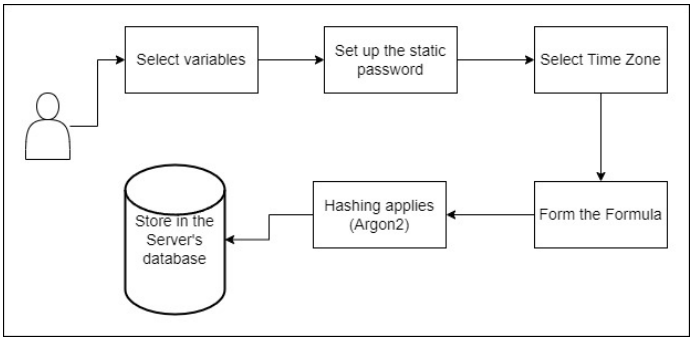

Fig. 2. DyMP-Gen Algorithm

V. SYSTEM ANALYSIS AND EXPERIMENTS

In this section, we will assume several user scenarios in terms of different formulas to generate strong passwords that meet all platforms' standards.

Use Case 1

A user is desired to use the word *"Tiger"* as a static word, and the following formula has been pre-generated:

$$\text{MI} + + \text{HI} + + \text{DN} + + (\text{MI} + \text{HI}) + + \text{MB}$$

In this case, if the current time and date is 2:06 pm, September/12/2024, then the dynamic password will be quickly calculated as:

$$2 + + 1 + + 15 + + (2 + 1) + + \$ = 10312\$$$

Then the dynamic password will be **"Tiger21153$"**.

Use Case 2

A user is desired to use the word *"Toyota"* as a static word, and the following formula has been pre-generated:

$$\text{Year} + + (\text{HI} \times \text{HI}) + + (\text{MN} + \text{MI}) + + \text{DB}$$

In this case, if the current time and date is 9:56 pm, January/05/2024, then the dynamic password will be quickly calculated as:

$$2024 + + (4 \times 4) + + (1 + 12) + + \text{odd}\,\$ = 20241613\$$$

Then the dynamic password will be **"Toyota20241613$"**.

Use Case 3

A user is desired to use the word *"Lovelyfamily"* as a static word, and the following formula has been pre-generated:

$$\text{DI} + + \text{year} + (\text{MN} \times \text{HI}) + + (\text{MI} \times \text{DNB}) + + \text{MB}$$

In this case, if the current time and date is 6:31 am, June/22/2024, then the dynamic password will be quickly calculated as:

$$25 + + 2024 + (6 \times 3) + + (7 \times 1) + + \# = 2224187\#$$

Then the dynamic password will be **"Lovelyfamily2224187#"**.

The above use cases are designed to show different complexities, and all of them are still easy for the brain to come up with. The main objective of our system is to balance security with simplicity. Thus, the user doesn't need any external tools or electronic devices to generate the password. On the other hand, this system offers a dynamic password-generation technique that enhances the user's security.

This paper proposed a system that has to be compatible with the server system. Both the user and the server sides have to follow the same algorithm to avoid any synchronization problems.

VI. SECURITY ANALYSIS

The main goal of this system is to overcome the issue of using the same password in all accounts or platforms. Herein, this system offers a dynamic password system.

A. Compatibility

The proposed system doesn't need any external interaction such as password manager applications or specific applications to generate the password, or even any device/tool. Also, this system doesn't need any online operations on the user's side since the password will be dynamically updated on the server's side. Hence, the user will be able to guess the password based on the pre-generated formula offline. One of the best features of the proposed system is the ease of use anytime, anywhere.

B. Defending against Possible Cracking Attacks

This system is designed to mitigate the possible man-in-the-middle attacks since the operations are done at the server's side and there's no need to share the password with any party so it can be compromised.

The system setup will be completed once, and then there's no need to update the password on the server side because it's designed to be dynamically changed over time.

One of the possible attacks is the dictionary attack, which is also considered to be avoided in the system design. The system always generates strong and complex passwords that meet all platform standards. On the other hand, the generated password is salted through the combination of dynamic variables.

C. Multi-Factor Authentication (MFA)

The proposed system also uses an MFA technique since it depends on several factors to dynamically generate the passwords, such as time, date, daytime/nighttime along with a prior static password chosen by the user.

Limiting the time for every generated password also will take a role to minimize the risk of compromising the user's password since the generated password could be changed every five minutes.

D. Flexibility

In this proposed system, at the setup stage, the user has the ability to form the formula and control the complexity of the generated formula. Moreover, the user can select any variable from the proposed variables based on his/her convenience, which makes remembering the formula easier and faster for the user.

E. Password Strength Analysis

To evaluate the strength of the generated passwords using this algorithm, the entropy was calculated as $E = \log_2(R^L)$. The most basic password generated with this algorithm boasts an entropy of 71 bits. In general, passwords with an entropy of 60 bits or more are considered good. A comparison of other methods for creating passwords is outlined in the table below contrasted with DyMP-Gen. (See table III)

TABLE III
EXAMPLE PASSWORDS AND THEIR ENTROPY (IN BITS)

Example Password	Entropy (bits)
Common Passwords	
123456	19.93
P@$sword	51
John1999	47
DyMP-Gen Passwords	
Tiger21153$	67
Toyota20241613$	92
Lovelyfamily2224187#	128

The results above clearly show that passwords generated by DyMP-Gen have significantly higher entropy compared to common passwords. This heightened entropy translates to a stronger, more secure password, thereby making it a good option.

VII. DISCUSSION

Comparing our system with other approaches. DyMP-Gen balances both security and usability. In terms of ease of use, our system doesn't require any electrical devices or password manager applications. Our system approach is to focus on users' memory to remember a dynamic password based on multiple pre-determined variables. Moreover, DyMP-Gen doesn't need internet access to perform. This paper proposes a system that needs other platforms to adopt our approach to operate on both sides. However, DyMP-Gen doesn't require any additional equipment since it's ready to be implemented on any platform's server. DyMP-Gen helps users to use unique passwords every time they try to log in to their account and this, in turn, enhances the password's security.

VIII. CONCLUSION

In this paper, we proposed a dynamic password system that significantly enhances security by frequently updating passwords. The system combines a user-chosen static component with dynamic elements based on real-time variables, such as current time and date, to generate unique passwords at regular intervals. This approach mitigates the risks associated with password reuse and strengthens overall security.

The design emphasizes simplicity and ease of use. Users create a straightforward formula through a brief setup survey, which allows them to generate their password offline without requiring additional devices. This setup process includes selecting a static password and defining variables that are easy to remember, ensuring that users can manage their passwords with minimal complexity.

Furthermore, the system accounts for potential time zone differences, ensuring that passwords are synchronized between the user and server for consistent authentication. Overall, this dynamic password system offers a practical and scalable solution that balances robust security with user convenience, addressing key challenges in modern password management.

IX. FUTURE WORK

As a future work, we are in the process of building a multi-factor authentication system that needs special software on the user's side. This application should generate a user's password with high complexity. We expect that the generated passwords are hard to crack and very secure. The proposed system will require more user engagement in generating passwords and a host to run our software to generate the user's password every time.

ACKNOWLEDGMENT

The authors acknowledge the support of various individuals including the Dean of the College of Arts and Science at Grambling State University, support from the LIDSS Grant, Department of Education grant funding, and Title III resource support.

REFERENCES

[1] NordPass, "Global Password Security Report," 2023. [Online]. Available: https://nordpass.com/

[2] Identity Theft Resource Center, "2023 Data Breach Report," 2023. [Online]. Available: https://www.idtheftcenter.org/publication/2023-data-breach-report/

[3] Google/Harris Poll, "Online Security Survey," 2021. [Online]. Available: https://services.google.com/fh/files/blogs/google_security_infographic.pdf

[4] LastPass, "Psychology of Passwords," 2022. [Online]. Available: https://www.lastpass.com/resources/ebook/psychology-of-passwords-2022

[5] Security.org, "Password Habits Survey," 2023. [Online]. Available: https://www.security.org/resources/online-password-strategies/

[6] J. Bonneau, C. Herley, P. C. van Oorschot, and F. Stajano, "The quest to replace passwords: A framework for comparative evaluation of web authentication schemes," in 2012 IEEE Symposium on Security and Privacy, 2012, pp. 553-567.

[7] D. Florencio and C. Herley, "A large-scale study of web password habits," in Proceedings of the 22nd international conference on World Wide Web, 2013, pp. 657-664.

[8] D. Wang, H. Cheng, P. Wang, M. K. Reiter, and X. Zhang, "Fork me once…: Examining the security implications of social coding in GitHub," in 25th USENIX Security Symposium (USENIX Security 16), 2016, pp. 757-772.

[9] NordPass, "Global Password Security Report," 2021. [Online]. Available: https://nordpass.com/

[10] YouGov, "Password Security Survey," 2020. [Online]. Available: https://yougov.zendesk.com/hc/en-us/sections/20768750708637-Connections

[11] S. Romanosky, L. Ablon, and N. Kuehn, "Examining the costs and causes of cyber incidents," Journal of Cybersecurity, vol. 6, no. 1, p. tyaa019, 2020.

[12] A. Das, N. Saxena, and L. F. Cranor, "Why Johnny can't encrypt: A usability evaluation of PGP 5.0," in 19th USENIX Security Symposium (USENIX Security 10), 2014, pp. 317-332.

[13] A. Adams and M. A. Sasse, "Users are not the enemy: Why users compromise security mechanisms and how to take remedial measures," Communications of the ACM, vol. 42, no. 12, pp. 41-46, 1999.

[14] M. L. Mazurek et al., "Measuring password guessability for an entire university," in Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security, 2013, pp. 173–186.

[15] N. Perlroth, "Yahoo Says Hackers Stole Data on 500 Million Users in 2014," The New York Times, 2016.

[16] P. G. Kelley et al., "Guess again (and again and again): Measuring password strength by simulating password-cracking algorithms," in 2012 IEEE symposium on security and privacy, pp. 523–537, IEEE, 2012.

[17] C. Herley and P. Van Oorschot, "A research agenda acknowledging the persistence of passwords," IEEE Security & Privacy, vol. 10, pp. 28–36, 2012.

[18] M. Weir et al., "Testing metrics for password creation policies by attacking large sets of revealed passwords," in Proceedings of the 17th ACM conference on Computer and communications security, 2010, pp. 162–175.

[19] H. Bojinov et al., "Kamouflage: Loss-resistant password management," in European symposium on research in computer security, Springer, 2010, pp. 286-302.

[20] S. Komanduri et al., "Of passwords and people: measuring the effect of password-composition policies," in Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 2011, pp. 2595–2604.

[21] T. Islam Tanni et al., "Is My Password Strong Enough?: A Study on User Perception in The Developing World," EAI Endorsed Transactions on Creative Technologies, vol. 9, no. 30, pp. 1-12, 2022.

[22] A. Kanta, I. Coisel, and M. Scanlon, "A novel dictionary generation methodology for contextual-based password cracking," IEEE Access, vol. 10, pp. 59178-59188, 2022.

[23] S. Wang et al., "Pivot-V: An Optimized Algorithm for Password Generation in PCFGs Model," in 2021 IEEE 6th International Conference on Signal and Image Processing (ICSIP), pp. 990-994, IEEE, 2021.

[24] X. Zhang et al., "A New Targeted Online Password Guessing Algorithm Based on Old Password," in 2023 26th International Conference on Computer Supported Cooperative Work in Design (CSCWD), pp. 1-6, IEEE, 2023.

[25] Y. Yin et al., "PassImg: A Secure Password Generation and Management Scheme without Storing," in Proceedings of the 2022 IEEE 25th International Conference on Computer Supported Cooperative Work in Design, 2022, pp. 341-346.

[26] J. B. Billa et al., "PassMan: A New Approach of Password Generation and Management without Storing," in 2019 7th International Conference on Smart Computing & Communications (ICSCC), IEEE, 2019, pp. 1–5.

[27] O. Tremblay-Savard et al., "Strong Password Generation Based On User Inputs," in 2019 IEEE 13th International Conference on Anti-counterfeiting, Security, and Identification (ASID), pp. 416-423, IEEE, 2019.

[28] M. Sameer et al., "Unforgettable Password Generation using LoRA Fine-tuned Large Language Model," in 2024 International Conference on Advances in Data Engineering and Intelligent Computing Systems (ADICS), IEEE, 2024, pp. 1-5.

[29] B. A. Smadi, A. A. S. AlQahtani and H. Alamleh, "Secure and Fraud Proof Online Payment System for Credit Cards," 2021 IEEE 12th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON), New York, NY, USA, 2021, pp. 0264-0268, doi: 10.1109/UEMCON53757.2021.9666549.

[30] B. A. Smadi, W. B. Glisson, M. Tahat, H. Alamleh and A. A. S. AlQahtani, "Credit Card Transactions Fraud Detection for Multiple Consumer Behaviors," 2024 International Conference on Computing, Networking and Communications (ICNC), Big Island, HI, USA, 2024, pp. 26-32, doi: 10.1109/ICNC59896.2024.10556040.