iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 **Issue:** XI **Month of publication:** November 2023

DOI: https://doi.org/10.22214/ijraset.2023.56988

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue XI Nov 2023- Available at www.ijraset.com

Effective Password Strengthening Techniques

Hemadri Rawat[1], Yamini Singh[2], Prof. Ashwini Kumar (Guide)[3]
Dep. of Electronics and Communication Engineering, Indira Gandhi Delhi Technical University For Women

Abstract: Protecting passwords from hackers is a formidable challenge in the current technological landscape, given the myriad of tools available in the hacking domain. This paper delves into an exploration of various password types and lengths, highlighting those that are ill-advised for use. Additionally, it discusses password types that offer comparatively greater security and outlines prevalent cracking techniques employed by hackers. The document further outlines effective measures to enhance password security, incorporating algorithms designed for this purpose. Comprehensive coverage is provided on methods to safeguard passwords from potential attacks. Notably, the paper extends its scope to include insights into the various types of Wi-Fi passwords.

I. INTRODUCTION

In the realm of user authentication, a password serves as a crucial set of characters, verifying identity or granting access to confidential resources. However, storing passwords in plain text is a security risk due to cyber threats. To safeguard against such threats, various methods and concepts, including algorithms, are employed. Hashing, a specific algorithm, transforms data of any size into fixed-length data. Modern hashing algorithms, such as MD-5, SHA-1, SHA-2, and SHA-3, play a key role in securing passwords. Additionally, three recommended password hashing algorithms—PBKDF2, b-crypt, and s-crypt—add an extra layer of protection. Different methods of hacking, such as dictionary attacks, brute-force attacks, rainbow attacks, DOS attack, man in the middle attack, and birthday attack, pose a substantial threat by targeting passwords. This paper explores preventive measures against these threats, emphasizing the importance of creating secure passwords. In the present-day landscape, cyber security concerns have escalated, with various unauthorized access methods posing significant threats. The compromise of passwords without consent is especially perilous, exposing personal data to unauthorized individuals. Diverse threads of attacks include dictionary attack, brute force attack, rainbow attack, DOS attack, man in the middle attack and birthday attacks.

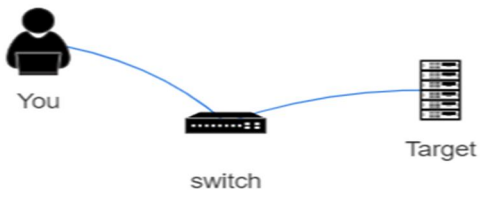
This paper delves into strategies for safeguarding passwords against the aforementioned threats. Recognizing the paramount importance of passwords in our daily lives, it emphasizes the necessity of fortifying them to ensure robust security.

II. PASSWORD CRACKING TECHNIQUES

Methods for cracking passwords encompass dictionary attacks, brute-force attacks, and rainbow attacks.Rainbow attacks leverage precomputed hashes for quick password cracking. It's crucial to implement preventive measures like salting passwords and using strong hashing algorithms to counter these techniques.

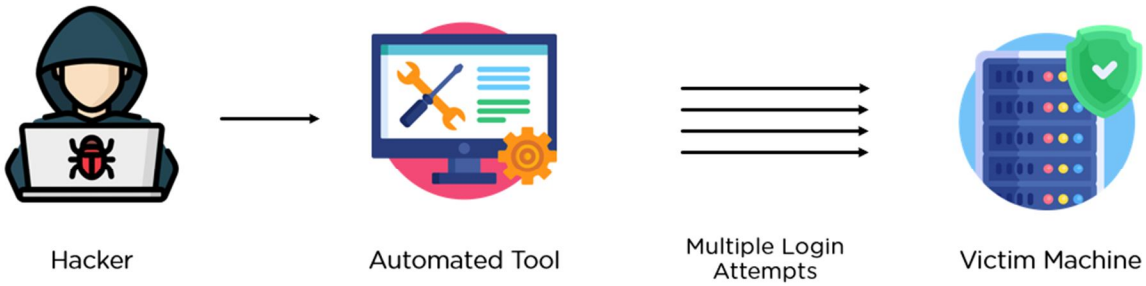
A. Dictionary Attack

Within the realm of cyber security, the dictionary attack emerges as a technique for unauthorized access to secured systems or accounts. This method systematically cross-references a predefined set of words, often drawn from dictionaries and encompassing commonly used passwords, with a password database. This database, presented as a text file, contains an array of dictionary words arranged in alphabetical order. The central objective of the attack is to efficiently identify matches between these dictionary words and the stored passwords. By leveraging users' tendencies to adopt easily predictable or commonly employed phrases as passwords, the dictionary attack exploits vulnerabilities associated with weak password choices. Consequently, this method poses a significant threat to overall security and is frequently employed by malicious entities seeking to compromise systems.



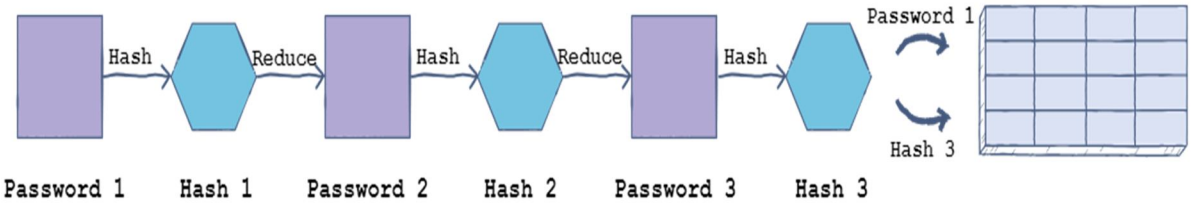
B. Brute Force Attack

Within the domain of cyber security, brute-force attacks employ a systematic approach to unveil passwords by exhaustively testing every possible combination of numbers, letters, and special characters. In the arsenal of password-cracking utilities, users can specify testing criteria, including character sets, preferred password length, and known characters, enabling the execution of a "mask" attack. This method poses a significant security risk as it relentlessly explores all potential password permutations until successfully infiltrating the targeted system.



C. Rainbow Attack

A rainbow password attack leverages rainbow cracking techniques to swiftly decipher password hashes associated with LM, NTLM, Cisco PIX, and MD5, boasting remarkably high success rates, often approaching 100 percent. This method expedites the password-cracking process by utilizing precomputed hashes, eliminating the need for on-the-fly generation, as seen in dictionary and brute-force cracking approaches. However, it is essential to acknowledge a length constraint inherent in rainbow attacks due to the substantial time required for generating these precomputed tables. Given sufficient time, an ample number of tables can be produced. Nevertheless, it's worth noting that over time, computers and applications may adopt diverse authentication mechanisms and hashing standards, potentially introducing new vulnerabilities that must be addressed.

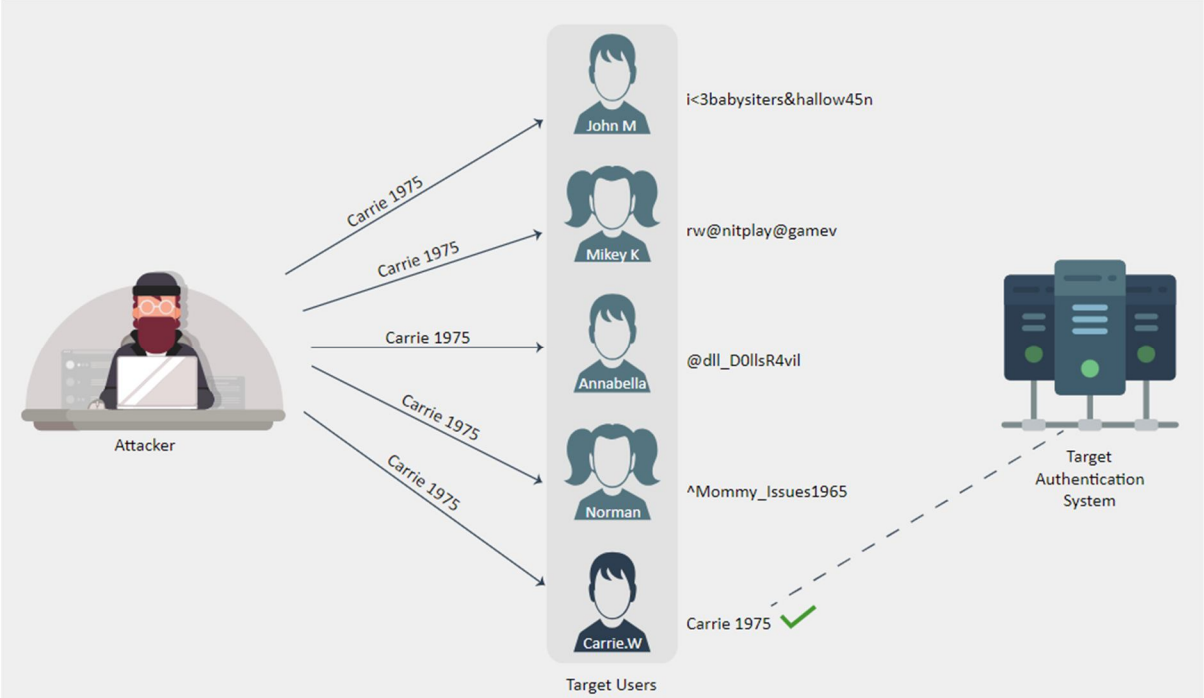


D. Password based Attack

A fundamental element in most operating system and network security strategies involves access control based on passwords. This entails that your permissions to access a computer and its network resources depend on your identity, specifically your user name and password.

In the context of older applications, the safeguarding of identity information during network validation is not always guaranteed. This vulnerability could potentially enable an eavesdropper to exploit the network by masquerading as a legitimate user.

Once an attacker identifies a valid user account, they inherit the same privileges as the authentic user. Consequently, if the compromised user possesses administrator-level rights, the attacker gains the capability to create accounts for future access. Subsequent to infiltrating the network through a valid account, the attacker can execute various actions, including acquiring lists of valid user and computer names along with network details, modifying server and network configurations (inclusive of access controls and routing tables), and manipulating, redirecting, or deleting data.



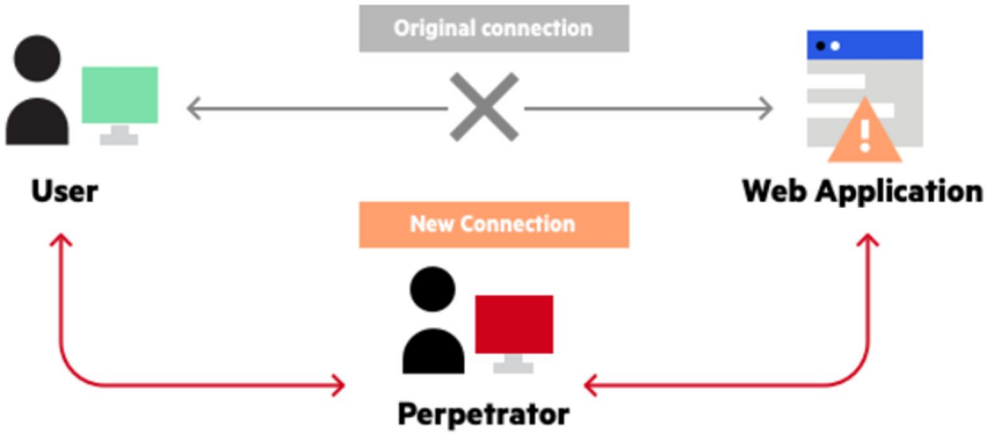
E. Denial-of-service Attack

In contrast to password-based attacks, a denial-of-service attack disrupts the regular use of your computer or network by legitimate users. Once the attacker gains entry to your network, they can execute various actions, such as diverting the attention of your internal Information Systems staff to create a window for additional attacks. Additionally, the attacker may transmit invalid data to applications or network services, inducing abnormal termination or disruptive behaviour. Another tactic involves overwhelming a computer or the entire network with excessive traffic, leading to a shutdown due to overload. Alternatively, the attacker may block traffic, causing authorized users to lose access to network resources.

F. Man-in-the-Middle

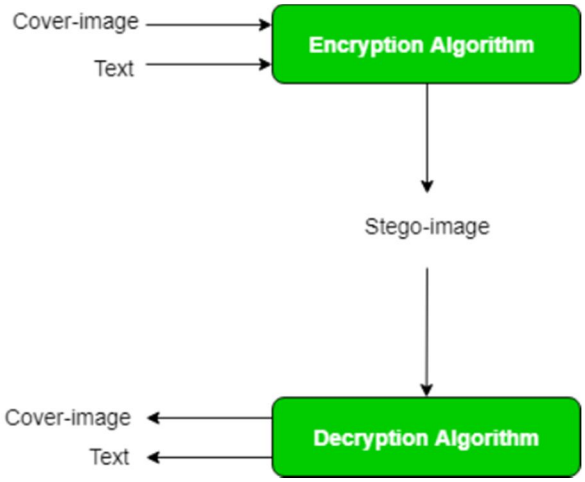
A man in the middle attack takes place whenever an intermediary, positioned between you and your communication counterpart, actively monitors, captures, and controls your communication in a covert manner. This could involve rerouting a data exchange, particularly when computers engage in communication at lower levels of the network layer, making it challenging for them to ascertain the identity of their communication partner.

In essence, man-in-the-middle attacks resemble an individual assuming your identity to intercept your messages. The recipient at the other end may be deceived into believing it is a genuine communication with you because the attacker can actively respond as you, thereby sustaining the exchange and extracting more information. The potential damage inflicted by this attack is comparable to that of an application-layer attack, as detailed later in this section.



G. Birthday Attack

This attack capitalizes on the Birthday paradox, a phenomenon that, in essence, suggests that with a substantial collection of user password digests, the likelihood of generating a password whose digest coincides with at least one digest in the set is significantly higher than one might intuitively anticipate. Furthermore, the likelihood of this occurrence significantly rises with the expansion of the set, proportional to the number of users.



III. ALGORITHMS

Hashing constitutes an algorithmic process that transforms data of variable sizes into a fixed-length format. This technique is commonly employed to streamline data retrieval by condensing substantial amounts of information into a more manageable string for comparison. To illustrate, consider a DNA sample, which inherently comprises considerable data (approximately 2.2 – 3.5 MB). When seeking to identify the individual associated with this sample, comparing the full 2.2 MB against an entire database of DNA samples could be time-consuming, particularly when dealing with numerous samples. Hashing provides a practical solution in this scenario. Instead of directly comparing the data, the hash of the information is computed (accounting for various chromosomal locations). This process yields a fixed-length value, such as 128 bits, making it more efficient to query a database compared to the original 2.2 MB dataset.

A crucial distinction between hashing and encryption lies in reversibility. Unlike encryption, a hash is non-reversible. When discussing cryptographic hash functions, certain properties are essential:

1) Efficient computation of the hash value for any given message.
2) Impracticality of generating a message with a specific hash.
3) Impracticality of modifying a message without altering the hash.
4) Impracticality of finding two distinct messages producing the same hash.

Furthermore, a robust hash function should resist:

a) Collisions (occurrence of two different messages generating identical hashes).
b) Preimage resistance, making it challenging to find a message corresponding to a given hash.
c) Resistance to second-preimages, wherein it is infeasible to find a distinct message producing the same hash as a given message (e.g., MD-5(m) = MD-5(m')).

IV. MODERN HASHING ALGORITHM

Some hashing algorithms you may encounter are:

- MD-5
- SHA-1
- SHA-2
- SHA-3

1) *MD-5:* MD-5, a widely utilized hashing algorithm, is acknowledged for its cryptographic vulnerabilities, particularly in terms of susceptibility to collisions. Despite its prevalent use, MD-5 is considered compromised due to its propensity for generating identical hash values for different inputs. Notably, MD-5 is compromised in the context of collisions, but its preimage and second-preimage resistance remain intact. The initial challenges to MD-5 surfaced in 1996, focusing on an attack on MD-5's compression function rather than the algorithm itself. Subsequently, in 2004, a theoretical attack emerged, posing a potential threat to the pre-image resistance property of MD-5. However, in practical terms, the execution of this attack is prohibitively slow, rendering it impractical for malicious use.
2) *SHA:* SHA, or Secure Hashing Algorithm, constitutes a group of cryptographic hash functions introduced by the National Institute of Standards and Technology (NIST) as part of the U.S. Federal Information Processing Standard (FIPS).

The SHA family encompasses three distinct algorithms:

- *SHA-1:* A 160-bit hash function, reminiscent of the earlier MD-5 algorithm, devised by the National Security Agency (NSA) for inclusion in the Digital Signature Algorithm. Cryptographic vulnerabilities in SHA-1 were identified, leading to its disapproval for most cryptographic applications after 2010.
- *SHA-2:* Comprising two similar hash functions, SHA-256 and SHA-512, differing in block sizes with SHA-256 using 32-bit words and SHA-512 using 64-bit words. Truncated versions, SHA-224 and SHA-384, were also standardized, all designed by the NSA.
- *SHA-3:* Currently undefined, NIST is determining the specific parameters for SHA-3, which will be based on Keccak, or a version deemed "close enough." The finalized SHA-3 is expected to deviate slightly from the originally proposed Keccak due to configurable function parameters.

Despite SHA-1 being recognized as "cryptographically broken," the fundamental properties sought in a password hashing algorithm remain valid. In practical terms, if a password hashing algorithm built on SHA-1 is implemented securely, there is no immediate necessity to transition to a newer alternative.

V. HASHING PASSWORD ALGORITHMS

There are currently three secure algorithms for password hashing:

A. PBKDF2

PBKDF2 is primarily used for key derivation but is also suitable for password hashing due to its deliberate slowness. The resulting derived key (HMAC) can effectively secure passwords. Despite being slower, the choice of the underlying hashing algorithm, such as SHA-1 or SHA-2, is crucial. PBKDF2, when implemented with SHA-256 or SHA-512 on a 64-bit PC, provides enhanced security. It iterates the SHA-1-HMAC(password+salt) calculation 1024 times, introducing a significant delay. However, distributed systems or GPU-based attacks can still pose a threat.

A notable consideration is that when passwords exceed 64 bytes, PBKDF2 shortens them through a hash, potentially reducing security. Although, this reduction may not significantly impact key brute-force attempts.

B. bcrypt:

bcrypt is the prevailing standard for secure password hashing. Derived from the Blowfish block cipher, it employs memory-intensive lookup tables for hash generation. This memory requirement poses a challenge for GPU-based attacks, offering robust security. Having been extensively vetted over 14 years, bcrypt is considered a reliable choice. However, FPGA processing units could potentially exploit its weaknesses, especially for longer passwords.

C. Scrypt

Scrypt introduces a novel approach to password hashing by emphasizing operations challenging for anything other than a PC, specifically random memory accesses. Similar to bcrypt, it enhances security by increasing calculation time and memory space exponentially with additional rounds. Scrypt, developed in response to evolving attacks on bcrypt, surpasses GPU and FPGA constraints due to its memory-intensive nature. While relatively newer than bcrypt, Scrypt has proven effective in countering emerging threats.

It's important to note that these algorithms play a crucial role in protecting user passwords against various attacks.

VI. PASSWORD STRENGTH

Password strength is vital in thwarting attacks. Length, randomness, and complexity contribute to strong passwords. Preventive measures, including salting and iterative hashing, make brute-force and dictionary attacks computationally intensive, providing better security. A robust hashing algorithm with passwords comprising a minimum of eight randomly chosen characters is always a good practice. Humans, by nature, struggle to remember and generate truly random sequences. Consequently, users are mandated to craft passwords containing a combination of numbers, letters, symbols, and at least one capital letter. This stipulation plays a crucial role in the context of password hashing, where various attack strategies are employed:

1) Dictionary Attacks

- Utilizes word lists containing commonly used passwords, words, names, and years.
- Hashing algorithms are applied to each word, and the generated hash is compared to the entries in the database.
- Successful matches reveal the password associated with the hashed word.

2) Brute-force Attacks

- Entails trying all conceivable combinations of characters.
- With passwords of at least eight characters and utilizing the ASCII character set, there are 128^8 possible combinations.
- Demonstrates the significance of password length, considering the computational capabilities of modern GPUs.
- Moore's law, highlighting the consistent growth in computing power, is a critical factor to consider when evaluating the feasibility of brute-force attacks over time.

3) Rainbow Tables

- Involves generating and pre-storing hash values in a database for rapid lookup during an attack.
- The length limitation of rainbow attacks is determined by the hashing algorithms in use.
- For instance, Microsoft LM hashes have a maximum length of 14 characters, and dictionary-based Windows Vista and 7 hashes support up to 16 characters.

A. Prevention from Attacks

1) *Brute-force Attacks:* Applying an iteration count, e.g., 1,000, to the hash function significantly increases the computational cost for a brute-force attacker generating millions of digests.
2) *Dictionary Attacks:* Introducing a random salt mitigates the weakness associated with common, dictionary-based passwords, as salts make passwords distinct and diminish the likelihood of digests matching a pre-existing set.
3) *Birthday Attacks:* Incorporating a random salt minimizes the success probability of a birthday attack. Each password would require a separate attack due to the uniqueness of salts, transforming the scenario into a more challenging brute-force endeavor.
4) *Rainbow Attacks:* The use of rainbow tables is constrained by the hashing algorithm's limitations, ensuring that longer passwords remain resistant to this method.

B. Securing Account

1) *Craft a Sentence Password:* Constructing a robust password involves creating a sentence with a minimum length of 12 characters. Opt for positive sentences or phrases that resonate with you and are easy to recall (e.g., "I love country music."). Some platforms even permit the inclusion of spaces.
2) *Account-Specific Passwords:* Enhance your security posture by maintaining distinct passwords for each account. It is advisable, at the very least, to segregate passwords for work and personal accounts. Critical accounts should be safeguarded with the most resilient passwords.
3) *Record and Secure Your Passwords:* Acknowledging the potential to forget passwords, maintain a securely stored list away from your computer. Alternatively, consider leveraging a password manager service to efficiently organize and safeguard your passwords.
4) *Strengthen Your Login:* Reinforce your online accounts by activating the most robust authentication tools available, such as biometrics, security keys, or unique one-time codes via a mobile app. Relying solely on usernames and passwords is insufficient for safeguarding vital accounts like email, banking, and social media.
5) *Avoid the Following as Passwords*

- Using 'standard' words like boss, master, doall, passwd.
- Employing a dictionary word or the business name.
- Utilizing repetitive letters or numerals (e.g., AAAAAA, 111111).
- Incorporating parental names.
- Including the names of best friends.
- Employing common names.

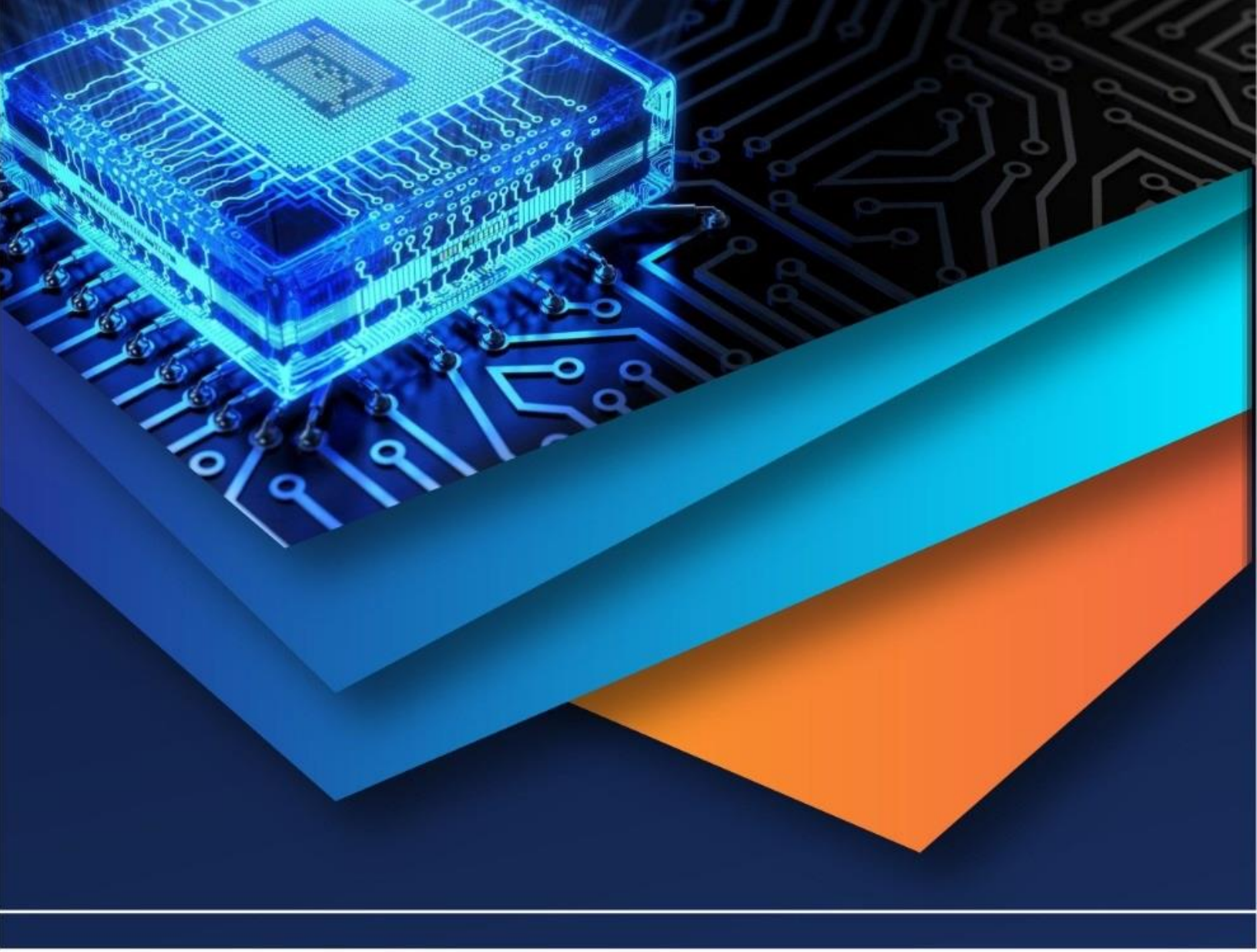
Ensure that the passwords you create do not fall into these categories to fortify your digital security effectively.

VII. CONCLUSION

While password security remains a challenge, implementing best practices, using robust hashing algorithms, and embracing alternative authentication methods contribute to a more secure digital environment. User awareness, adherence to secure practices, and continuous advancements in cybersecurity measures are crucial in mitigating evolving threats.

REFERENCES

[1] https://www.researchgate.net/publication/236898951_A_Survey_of_Password_Attacks_and_Comparative_Analysis_on_Methods_for_Secure_Authentication
[2] R. Morris and K. Thompson. “Password security: a case history” Communications. ACM, 22(11):594–597, 1979.
[3] M. Weir, Using Probabilistic Techniques to aid in Password Cracking Attacks, Dissertation, Florida State University, 2010
[4] http://www.openwall.com/john/ [Online document]
[5] T. Booth and R. Thompson, “Applying Probability Measures to Abstract Languages,” IEEE Transactions on Computers, Vol. C-22, No. 5, May 1973
[6] Yan, J.J., Blackwell, A., Anderson, R. and Grant, A., “The Memorability and Security of Passwords -- Some Empirical Results”, Technical Report No. 500 (September 2000) Computer Laboratory, University of Cambridge.
[7] Notoatmodjo, Gilbert and Thomborson, Clark. Passwords and perceptions.
[8] Proceedings of the Seventh Australasian Conference on Information Security – Volume 98 (AISC 2009, pages 71–78. Australian Computer Society, Jan 2009.
[9] I. Ghafir, V. Prenosil, M. Hammoudeh, F. J. Aparicio-Navarro, K. Rabie and A. Jabban, “Disguised Executable Files in Spear-Phishing Emails: Detecting the Point of Entry in Advanced Persistent Threat.” International Conference on Future Networks and Distributed Systems. Amman, Jordan, 2018.
[10] Miller, G.A. The magical number seven, plus or minus two: some limits on our capacity for processing information. Psychol. Rev. 63(2), 81 (1956)
[11] Malaysia, pp. 1-6, 2014.
[12] I. Ghafir, M. Husak and V. Prenosil, “A Survey on Intrusion Detection and Prevention Systems,” IEEE/UREL conference, Zvule, Czech Republic, pp. 10-14, 2014.
[13] Carstens, D.S., Malone, L.C., McCauley-Bell, P. Applying chunking theory in organizational password guidelines. 1, 97–113 (2006)
[14] I. Ghafir, J. Svoboda and V. Prenosil, “Tor-based malware and Tor connection detection,” International Conference on Frontiers of Communications, Networks and Applications. Kuala Lumpur,
[15] [13] Shirley Gaw, Edward W. Felten. Password Management Strategies for Online Accounts – Quantifying Password reuse, 2006
[16] Matt Bishop. COMPCON Spring '91 Digest of Papers - Password management, 1 March 1991
[17] Bander AlFayyadh, Per Thorsheim, Audun Jøsang and Henning Klevjer, Improving Usability of Password Management with Standardized Password Policies 03 June 2014
[18] I. Ghafir and V. Prenosil, “Advanced Persistent Threat and Spear Phishing Emails.” International Conference Distance Learning, Simulation and Communication. Brno, Czech Republic, pp. 34-41, 2015.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY