**PASSWORD ANALYZER: ANALYSIS OF USER ATTRIBUTES**

**SEMINAR-II REPORT**

***Submitted by***

# GANESH. D [RA2111030020057]

# SWAMINATHAN.S [RA2111026020056]

# NAVEEN RAMANA [RA2111030020014]

### Under the guidance of

**Dr. R. Nagalakshmi M.E., (Ph.D).,**

**Mrs. P. Jayalakshmi M.E., Ph.D.**

**(Assistant Professors, Department of Computer Science and Engineering)**

***In partial fulfilment for the award of the degree***

***of***

**BACHELOR OF TECHNOLOGY**

***in***

## COMPUTER SCIENCE AND ENGINEERING

## with specialization in Cyber Security

***of***

## FACULTY OF ENGINEERING AND TECHNOLOGY

## 

**SRM INSTITUTE OF SCIENCE AND TECHNOLOGY**

**RAMAPURAM, CHENNAI-600089.**

**MAY 2024**

**SRM INSTITUTE OF SCIENCE AND TECHNOLOGY**

(Deemed to be University Under Section 3 of UGC Act, 1956)

**BONAFIDE CERTIFICATE**

Certified that the Seminar-II report titled “**PASSSWORD ANALYSER: ANALYSIS OF USER ATTRIBUTES**” is the bonafide work of “**GANESH. D [RA2111030020057], SWAMINATHAN.S [RA2111030020056], NAVEEN RAMANA [RA2111030020014]**” submitted for the course 18CSP106L Seminar – II. This report is a record of successful completion of the specified course evaluated based on literature reviews and the supervisor. No part of the Seminar Report has been submitted for any degree, diploma, title, or recognition before.

#### 

**SUPERVISOR SIGNATURE HOD SIGNATURE**

P JAYALAKSHMI, M.E., (Ph.D)., Dr. K. Raja, M.E., Ph.D.

Assistant Professor Professor and Head

Dept. of Computer Science & Engineering Dept. of Computer Science & Engineering

SRM Institute of Science and Technology SRM Institute of Science and Technology

Ramapuram, Chennai. Ramapuram, Chennai.

Submitted for the Seminar-II Viva Voce Examination held on \_\_\_\_\_\_\_\_\_\_\_\_\_\_ at SRM Institute

Of Science and Technology, Ramapuram, Chennai-600089.

**EXAMINER 1 EXAMINER 2**

**TABLE OF CONTENTS**

**C. No. PAGE NO**

**LIST OF FIGURES**

**ABSTRACT**

1. **INTRODUCTION 1**

1.1 Objective

1.2 Project Statement 3

1.3 Project Domain 4

1.4 Project Scope 4

1. **PROJECT DESCRIPTION 7**

2.1 Existing System 7

2.2 Project Description 7

2.3 Related Studies 8

2.4 Literature Review 9

1. **SYSTEM DESIGN 11**

3.1 Proposed System 11

3.2 Architecture Diagram 11

3.3 Design Phase 14

3.4 Sequence diagram 15

3.5 Data Preprocessing Module 16

1. **METHODOLOGY 18**
2. **CONCLUSION AND FUTURE ENHANCEMENT 20**

5.1 Conclusion 20

5.2 Future Enhancement 21

**REFERENCES 22**

**LIST IF FIGRURES**

**S.NO FIGURE NAME PAGE. NO**

**3.2 ARCHITECTURE DIAGRAM 11**

**3.3 DESIGN PHASE 14**

**3.4 SEQUENCE DIAGRAM 15**

**ABSTRACT**

A comprehensive password analyser is a critical cybersecurity tool designed to evaluate and improve password security. It assesses key factors such as password length, complexity, and diversity of characters (uppercase, lowercase, numbers, and special symbols), guiding users to create strong, robust passwords that are resistant to common hacking techniques. The tool enforces security checks to ensure compliance with organizational password policies, which typically require a certain level of complexity, prohibit specific patterns, and mandate password rotation. This helps maintain a consistent security standard across an organization, reducing vulnerabilities. A unique feature of a comprehensive password analyzer is its ability to detect compromised passwords by cross-referencing against databases of known data breaches. This feature alerts users to avoid using passwords that might have been compromised, lowering the risk of unauthorized access. The analyzer also includes history tracking, which prevents the reuse of old or weak passwords. This capability ensures users continuously update their passwords with secure combinations, promoting a more dynamic and secure environment. The primary goal of the comprehensive password analyzer is to protect against common password-based attacks, such as brute force and dictionary attacks. By promoting best practices in password creation and management, it fosters a culture of strong password hygiene. Overall, this tool is vital for organizations and individuals seeking to strengthen cybersecurity and defend against evolving threats.

**CHAPTER – 1**

**INTRODUCTION**

In today’s automated and highly interconnected networks, security has become a critical priority for both individuals and organizations. As automation and technological sophistication grow, so do the threats from malicious actors, potentially causing significant damage. To combat these threats, robust security measures are necessary, with passwords serving as the first line of Defence. A password, typically a combination of characters and numbers, acts as a digital key that authorizes individuals to access restricted information or systems. Given its central role in security, there is a pressing need for tools that can ensure passwords are strong and well-managed. Passwords are widely used for securing access to various systems, user accounts, and sensitive data. However, maintaining strong passwords is a challenge, especially since a single user may have multiple accounts, each requiring a unique password. A strong password generally includes uppercase and lowercase letters, numbers, and special characters, while avoiding easily guessed information like names, birthdates, or common words. Research indicates that users often reuse passwords across multiple accounts with slight variations, making it easier for attackers to compromise multiple systems once a single password is breached. Password strength is critical because a weak password can lead to unauthorized access and severe security breaches. With the increasing number of accounts and applications that require passwords, from postal machines and ATMs to user authentication systems, users often struggle to remember unique passwords for each. This challenge can result in poor password practices that compromise security. Furthermore, online attackers can leverage network access to crack weak passwords, increasing the risk to individual and organizational security.

To address these issues, we propose a comprehensive password analyzer designed to evaluate the security of user passwords in real-time. This tool analyses the structure and complexity of a password, identifying potential vulnerabilities and offering recommendations for improvement. It goes beyond traditional password assessment by incorporating checks against known compromised passwords and ensuring compliance with organizational security policies. This analyzer supports organizations in establishing robust guidelines for password creation and management, reducing the risk of password-related security breaches. Password authentication remains the most common form of user authentication, even though more secure alternatives like biometrics and smart cards exist. The password analyzer helps organizations and individuals strengthen their password practices, ensuring a higher level of security against evolving cyber threats. By fostering good password hygiene and providing actionable insights, the password analyzer is a crucial tool in promoting strong password policies and enhancing overall cybersecurity.

**1.1 Objective of the Project**

Passwords serve as a system’s universal means of authentication and serve as a user’s identity. People are always reminded that in order to safeguard personal information, they need to use a strong password. It has become a prevalent method for users to secure themselves and gain access to restricted services. A password analyzer that analyses the passwords of the user and determine password strength based on existing methods and using the check of user related attributes for the password. The objective of a password analyzer is to help users create strong, secure passwords that are resilient against various types of cyber threats, thereby enhancing the overall security of their accounts and data.

The tool should analyse the password in order to determine its overall strength, taking into consideration factors like length, complexity, and unpredictability.

Identifying weaknesses ought to point out any weaknesses and flaws in the password, like the usage of words from dictionaries, recurring themes, or dependable sequences.

**1.2 Project Statement**

Developing a password analyzer that protects user accounts and the data they store is the fundamental problem. The problem is to develop a system that analyses the strength of a given password. A strong password is crucial for cybersecurity as it minimizes the risk of unauthorized access to accounts and data. The password analyzer achieves the objective through evaluating the strength of passwords, which is the first line of defence against unauthorized access. By performing a thorough assessment of password difficulty based on several parameters, the system is able to differentiate passwords that are vulnerable to attacks using brute force and ones that would prove difficult for the most determined attacker to guess. In the end, the system provides users direct input on how strong their passwords are (inadequate, moderate, strong). This input is crucial because it ensures the users to make strong passwords and take part in in safeguarding their digital and physical data in their system or a cloud-based service that they use.

**1.3 Project Domain**

The project domain is cybersecurity with a focus on password management. The goal is to develop a comprehensive password analyzer that enhances password security through a combination of password strength assessment and compromised password detection. It assesses password length, complexity, and character diversity, ensuring compliance with organizational security policies. A key feature is its ability to cross-reference against databases of known data breaches, alerting users to avoid compromised passwords. The analyzer also incorporates history tracking to prevent password reuse, promoting a dynamic and secure password environment. By implementing these features, the project aims to minimize common password-based attacks like brute force and dictionary attacks, fostering a culture of strong password hygiene across organizations and individual users alike.

**1.4 Project Scope**

* **Complexity**

Assessing the complexity of passwords based on factors such as length, character variety like uppercase, lowercase, numbers, special characters, and randomness.

* **Common patterns and sequences**

The password analyzer recognizing and flagging any password patterns, repeats, or predictable sequences that can expose them to dictionary or brute force assaults.

* **Dictionary Attack Resistance**

Checking passwords to make sure they don’t contain words from dictionaries, phrases that are often used, or information that are not dictionary words, common phrases, or easily guessable information that could be exploited by attackers using automated tools.

* **Brute Force Attack Resistance**

Analysing passwords to determine their resistance against brute force attacks, where attackers systematically try all possible combinations until the correct password is found. A password analyser should be of brute force resistance.

* **Compliance with Security Policies**

Verifying that passwords adhere to security policies and requirements set by organizations, including minimum length, complexity criteria, and

restrictions on the use of personal information should be done by the password analyser to establish the means of security operations. Providing users with clear feedback on the strength of their passwords and offering suggestions for improvement, such as adding more characters, incorporating special characters, or avoiding common patterns also a method to provide password security.

* **Continuous Improvement**

Incorporating updates and enhancements to adapt to evolving threats and security best practices, ensuring that the password analyzer remains effective and relevant over time.

It reinforces the importance of using strong password policies and promotes better password hygiene across organizations and online platforms.

* **Password age and history**

It is a security measures evaluate the length of time a password has been used in addition to the user’s previous password selections. To improve security, these techniques are frequently used in password management and authentication systems.

Because they maintain account of how long a password has been used and the user’s past password selections, password age and history are important security factors. It reduces the possibility of unauthorized access to accounts by limiting the reuse of outdated or simple-to-guess passwords.

* **Blacklist check**

Blacklist checking in passwords involves verifying if a chosen password appears on a list of known compromised or weak passwords, helping users select stronger, more secure options.

**CHAPTER 2**

**PROJECT DESCRIPTION**

**2.1 Existing System**

A password analyzer is a valuable tool designed to evaluate the strength and security of a given password. It scrutinizes various characteristics to assess how resistant the password is to unauthorized access. An effective analyzer considers factors like length and complexity, checking whether the password has a mix of uppercase and lowercase letters, numbers, and special characters. It also identifies common patterns, such as predictable sequences like "12345" or common words like "password" or "CAT." Advanced password analysers may also calculate randomness, providing a measure of the password’s randomness to gauge how challenging it would be for an attacker to guess it. Additionally, they often cross-check the password against blacklists of commonly used or compromised passwords and consult databases that track whether the password has appeared in known data breaches.

**2.2 Project Description**

Developing a password analyzer that protects user accounts and the data they store is the fundamental problem. The problem is to develop a system that analyses the strength of a given password. A strong password is crucial for cybersecurity as it minimizes the risk of unauthorized access to accounts and data.

The password analyzer achieves the objective through evaluating the strength of passwords, which is the first line of defence against unauthorized access. By performing a thorough assessment of password difficulty based on several parameters, the system is able to differentiate passwords that are vulnerable to attacks using brute force and ones that would prove difficult for the most determined attacker to guess. In the end, the system provides users direct input on how strong their passwords are (inadequate, moderate, strong). This input is crucial because it ensures the users to make strong passwords and take part in in safeguarding their digital and physical data in their system or a cloud-based service that they use.

**2.3 Related Studies**

Proactive Password Strength Analyzer Using Filters and Machine Learning Techniques: Passwords are ubiquitous authentication methods and they represent the identity of an individual for a system. Users are consistently told that a strong password is essential these days to protect private data. Despite the existence of more secure methods of authenticating users, including smart cards and biometrics, password authentication continues to be the most common means in use. Thus, it is important for organizations to recognize the vulnerabilities to which passwords are subjected, and develop strong policies governing the creation and use of passwords to ensure that those vulnerabilities are not exploited. This work proposes a framework to analyse the strength of the password proactively. To analyse the chosen password, filters and support vector machine are employed. This framework can be implemented as a submodule of the access control scheme.

**2.4 Literature review**

1. Garfinkel Simson and Spafford Gene, Practical UNIX Security, Sebastopol, CA:O'Reilly & Associates, Inc., pp. 35, 1991.

2. Stallings William, Network Security Essentials: Applications and Standards, Prentice-Hall, 2000.

3. Aviel D. Rubin, White-Hat Security Arsenal: Tackling the Threats, Addison-Wesley, 2001.

4. Choosing a safe password, Secure access to the TIAA-CREF Web Center, TIAA-CREF, ITS guidelines for choosing passwords Rachna Dhamija and Adrian Perrig.

5. "Déjà Vu: A user study using images for authentication", Usenix Security Symposium, 2000.

6. Anne Adams and Martina Angela Sasse, "Users are not the enemy", Communications of the ACM, vol. 42, pp. 40-46, December 1999.

7. Moshe Zviran and William J. Haga, "User authentication by cognitive passwords: An empirical assessment", Proc. IEEE Jerusalem Conference on Information Technology 1990: Next Decade in Information Technology, pp. 137-144.

8.K. Kato and V. Klyuev, "Strong passwords: Practical issues," 2013 IEEE 7th International Conference on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS), Berlin, Germany, 2013, pp. 608-613.

9 J. Ma, W. Yang, M. Luo and N. Li, "A Study of Probabilistic Password Models," 2014 IEEE Symposium on Security and Privacy, Berkeley, CA, USA, 2014, pp. 689-704.

10. V. Taneski, M. Heričko and B. Brumen, "Password security — No change in 35 years?" 2014 37th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), Opatija, Croatia, 2014, pp. 1360-1365.

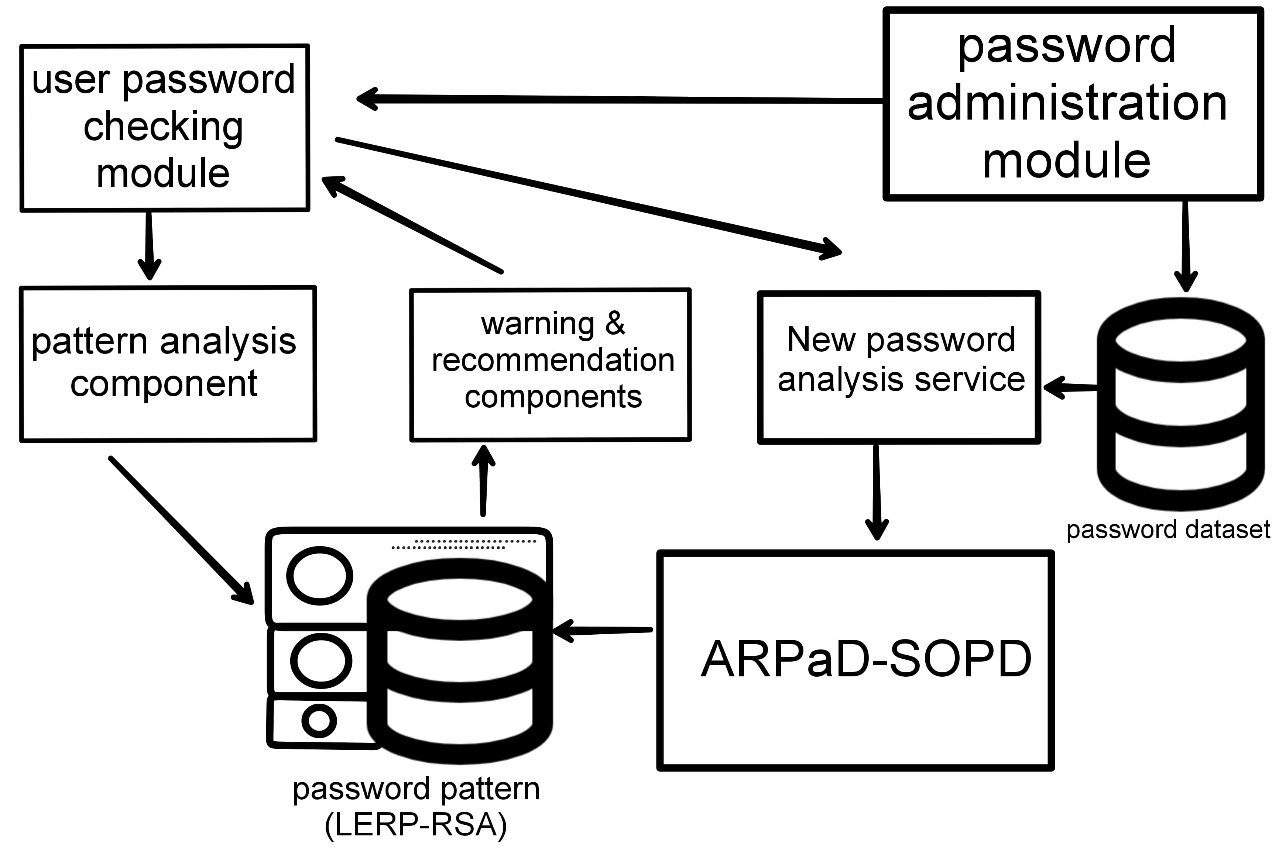
**CHAPTER 3**

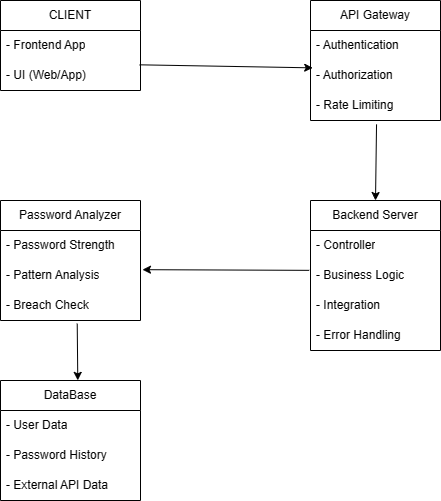
**SYSTEM DESIGN**

**3.1 Proposed System**

Here we propose a password analyser which checks for randomness in the passwords and cross-check the password against blacklists of commonly used or compromised passwords and consult databases that track whether the password has appeared in known data breaches and also analyses the user input and user attributes which are given by the user to enhance the ability of the password analyser.

**3.2 Architecture Diagram**





**Client**:

* The client is where users interact with the system. This could be a web application, a mobile app, or even a command-line tool.
* Users enter their passwords or other related information through the client UI.

**API Gateway:**

* Acts as an intermediary between the client and backend server.
* Handles authentication, authorization, rate limiting, and other common cross-cutting concerns.
* It also helps in routing requests to appropriate backend services.

**Backend Server:**

* Contains business logic to process requests from the client.
* Validates input, invokes the password analyzer, and retrieves or stores data.
* Might also integrate with other internal or external services as needed.

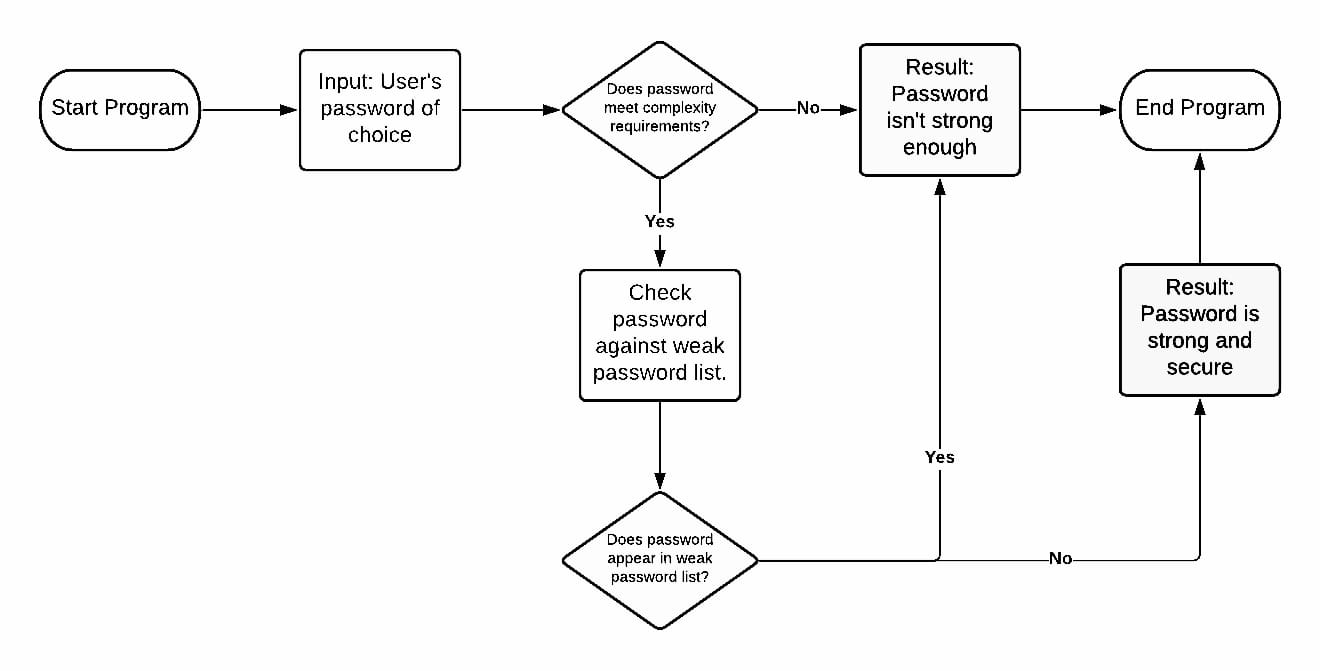
**Password Analyzer:**

* The core component for analyzing passwords.
* Includes different modules for analyzing password strength, identifying patterns (like common words, sequences, etc.), and checking against known breaches.
* This can be a separate service or integrated within the backend server, depending on scalability and separation concerns.

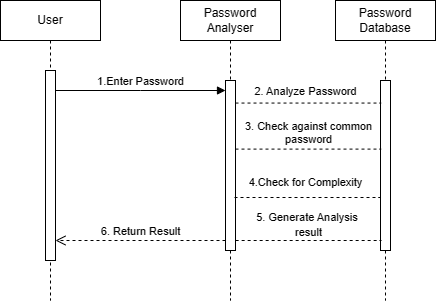
**Databases:**

* Stores user data, password history, and other relevant information.
* May include an external database for checking against known password breaches (like Have I Been Pwned).
* Ensure data is encrypted and secured, especially when storing sensitive information.

**3.3 Design Phase**

****The design phase for a password analyzer involves a structured approach to address both functional and non-functional requirements. First, gather the key features needed, such as password strength analysis and breach checks, along with security, scalability, and compliance considerations. Next, design the client interface, focusing on usability and clear feedback, while setting up the API Gateway for routing, authentication, and authorization. The backend requires robust business logic to process password analyses, and the password analyzer itself should incorporate algorithms to evaluate password strength and check for common patterns and breaches

**3.4 Sequence diagram**

****

* User Inputs Password: The user enters a password for analysis.
* Password Analyzer Analyzes Password: The analyzer receives the input and starts the analysis process.
* Check Against Common Passwords: The analyzer checks if the password is in a database of common passwords.
* Check for Complexity and Length: The analyzer assesses the password's length, complexity (e.g., use of special characters, upper/lowercase), and other common password strength metrics.
* Generate Analysis Result: The analyzer creates a report with the analysis results, including potential security weaknesses and suggestions for improvement.
* Return Analysis Result to User: The analyzer returns the results to the user, possibly via a UI or a simple console output.

**3.5 Data Preprocessing Module**

CHECK

if len(password) >= 10:

score += 1

Check for uppercase letters

if re.search(r'[A-Z]', password)

Check for lowercase letters

if re.search(r'[a-z]', password):

score += 1

Check for numbers

if re.search(r'\d', password):

score += 1

Check for special characters

if re.search(r'[\W\_]', password):

score += 1

Evaluate common patterns

common\_patterns = ['123', 'password', 'qwerty', 'admin']

if any(pattern in password.lower() for pattern in common\_patterns):

score -= 1 # Penalize for common patterns

Provide feedback on strength

if score <= 1:

strength = "Very Weak"

elif score == 2:

strength = "Weak"

elif score == 3:

strength = "Moderate"

elif score == 4:

strength = "Strong"

else:

strength = "Very Strong"

return password strength: {strength} (Score: {score})"

**CHAPTER 4**

**METHODOLOGY**

The uneven distribution of the predictability of user-created passwords, however, makes the entropy of an entire password set a less useful metric. More recently, researchers have argued against using entropy, preferring to use metrics likes “marginal guesswork”. This metric examines the number of guesses required to guess a given password, for a given way of generating guesses. Furthermore, the entropy of a password set does not sufficiently convey how difficult it is for an attacker to guess passwords from that set. Researchers have presented statistical techniques for modelling how much of a password set with stands attacks of different strengths, demonstrating these techniques on a very large scale gathering passwords. These statistical algorithms work effectively for large password databases, but they are not applicable to small datasets and do not accurately represent attackers in the real world. Therefore, we show password strength as a function of the number of guesses needed to crack a password using specific training data and certain password-cracking techniques. Model of guess. Given a set of unknown passwords randomly selected from W, X: {x1,..., xM}, with M ≤ N, we assume the attacker has knowledge of the entire password dataset, W: {w1,..., wN }. We wish to evaluate the data efficiency of the attacker’s effort to identify the set X uniquely.   
The guesswork, or guessing entropy, G(W), defined as follows, is one of the first measures to assess the guessing robustness of a password  
G(W)=X

N

i=1

pi

·i

The guessing estimates the predicted number of guesses X = {x1} needed to discover an unknown password by trying every password in an ideal sequence, that is, from the most likely w1 to the least probable wN. guesswork measures the expected number of guesses before succeeding, taking into account all the instances of the unique passwords. Group D1 is the most robust to guessing attacks given the high number of unique passwords Since G(W) involves an exhaustive search over all the passwords, i.e., even the most unlikely one such as wN, it cannot be considered an efficient way to guess an unknown password. Generally speaking, the above method requests an average of M ·G(W) guesses when the number of passwords unknown to the attacker are M, i.e., X : {x1,...,xM}. Nevertheless, a smarter attacker might do much better with the optimal strategy of first guessing the most likely password w1 for all unknown passwords and then move to the second one w2, and so on. By considering this model there are several guessing measures. The first metric we evaluate is the β-success-rate, yielding: β λβ(Wg) = i=1 pi λβ(Wg) ∈ [0,1] measures the expected success for an attacker limited to β ∈ [1, . . . , N] password guesses. The number of guessed passwords, for each of the considered group. We observe that for low values of guessed passwords, i.e., β << 100, the four groups behave significantly different, i.e., λβ(WD1) > λβ(WD2) ≈ λβ(WD4) > λβ(WD3). For large values of guessed passwords, i.e., β >> 100, the different cardinalities of the datasets affect the results and group D1 turns out to be the most robust.

**CHAPTER 5**

**CONCLUSION AND FUTURE ENHANCEMENT**

**5.1 Conclusion**

In conclusion, because they provide vital information regarding the security and strength of passwords, password analyzers are indispensable instruments for enhancing cybersecurity. These tools help users create stronger passwords that are more challenging for hackers to crack by evaluating variables like length, complexity, patterns, entropy, and known breaches. They also offer practical advice on how to improve password security, such as avoiding themes that are too common and using a range of character combinations. Password analyzers can significantly improve security, but it's important to choose reputable programs that respect user privacy and don't retain or abuse private information. The use of password analyzers can be reinforced even more by combining them with other advised procedures, enhance both individual and company cybersecurity.

**5.2 Future Enhancement**

Combining AI and ML to Detect Threats in Real Time: Use machine learning algorithms to detect new password-related threats, such as unique brute-force techniques or new patterns that hackers are employing. This might provide users the most recent password security advice.

Behavioral Analysis: Employ AI to analyze user behavior and spot anomalous activity that might point to a compromised password. Then, provide early alerts and recommend changing passwords.

Integration of Password Generation and Management: To automatically generate complex, secure passwords, integrate secure password generation capabilities into the analyzer. Reduce password reuse and streamline password management by integrating with password managers.

Cloud Integration and Cross-platform Compatibility: Create password analyzers that work with a variety of platforms and gadgets, including cloud services. This would guarantee encrypted communication and storage and allow users to verify the security of their passwords from any device.

Integration with Other Security Tools: Make password analyzers compatible with Single Sign-On (SSO), two-factor authentication (2FA), and Security Information and Event Management (SIEM) systems, among other cybersecurity tools. This might offer a comprehensive perspective on user security.

Continuous Monitoring and Alerts: Provide users with immediate action steps to reduce risks by implementing real-time monitoring of password-related events and sending out alerts for possible security breaches.

Enable users to securely collaborate and share strong passwords with family members or authorized team members while maintaining end-to-end encryption to protect user privacy.

**REFERENCE**

1. Proactive Password Strength Analyzer Using Filters and Machine Learning Techniques

Suganya, G., Karpgavalli, S., & Christina, V. (2010). Proactive password strength analyzer using filters and machine learning techniques. *International Journal of Computer Applications*, *7*(14), 1-5.

2. The trends in the offline password-guessing field: Offline guessing attack on Swedish real-life passwords

Zarzour, Y., & Alchtiwi, M. (2023). The trends in the offline password-guessing field: Offline guessing attack on Swedish real-life passwords.

3. A Longitudinal Study on Web-Sites Password Management (in)Security: Evidence and Remedies

Raponi, S., & Di Pietro, R. (2020). A longitudinal study on web-sites password management (in) security: Evidence and remedies. *IEEE Access*, *8*, 52075-52090.

4. A Password-Based Authentication System Based on the CAPTCHA AI Problem

Alajmi, M., Elashry, I., El-Sayed, H. S., & Faragallah, O. S. (2020). A password-based authentication system based on the CAPTCHA AI problem. *IEEE Access*, *8*, 153914-153928.

5. Kanta, A., Coisel, I., & Scanlon, M. (2022). A novel dictionary generation methodology for contextual-based password cracking. *IEEE Access*, *10*, 59178-59188.

6. That Was Then, This Is Now: A Security Evaluation of Password Generation, Storage, and Autofill in Browser-Based Password Managers

Oesch, S., & Ruoti, S. (2020, August). That was then, this is now: A security evaluation of password generation, storage, and autofill in browser-based password managers. In *Proceedings of the 29th USENIX Conference on Security Symposium* (pp. 2165-2182).

7. Your Culture is in Your Password: An Analysis of a Demographically-diverse Password Dataset

AlSabah, M., Oligeri, G., & Riley, R. (2018). Your culture is in your password: An analysis of a demographically-diverse password dataset. *Computers & security*, *77*, 427-441.

8. Kumar, B. P., & Reddy, E. S. (2019). Identification of password strength and time analysis for hacking the generated key: a survey. *Int J Anal Exp Modal Anal XI (VII)*, 1307-1315.

9. Password Security: An Analysis of Password Strengths and Vulnerabilities Chanda, K. (2016). Password security: an analysis of password strengths and vulnerabilities. *International Journal of Computer Network and Information Security*, *8*(7), 23.

10. Designing Password Policies for Strength and Usability

Shay, R., Komanduri, S., Durity, A. L., Huh, P., Mazurek, M. L., Segreti, S. M., ... & Cranor, L. F. (2016). Designing password policies for strength and usability. *ACM Transactions on Information and System Security (TISSEC)*, *18*(4), 1-34.

11. A Simple and Secure Reformation-Based Password Scheme

Ali, M., Baloch, A., Waheed, A., Zareei, M., Manzoor, R., Sajid, H., & Alanazi, F. (2021). A simple and secure reformation-based password scheme. *IEEE Access*, *9*, 11655-11674.

12. Password cracking based on learned patterns from disclosed passwords

Chou, H. C., Lee, H. C., Yu, H. J., Lai, F. P., Huang, K. H., & Hsueh, C. W. (2013). Password cracking based on learned patterns from disclosed passwords. *IJICIC*, *9*(2), 821-839.