Impact Factor (QJIF): 8.47 E-ISSN: 2583-1615, P-ISSN: 3049-3498

International Journal of Research in Academic World

Received: 30/January/2026 **IJRAW:** 2026; 5(3):190-194 **Accepted:** 06/March/2026

Secure Login System with Password Strength Analyzer

[1]Bhuvaneshwari NA and [*2]R Nithya

[1]Student of II Year M.Sc., Department of Computer Science, Bon Secours College for Women (Autonomous), Thanjavur, Tamil Nadu, India.

[*2]Assistant Professor, Department of Computer Science, Bon Secours College for Women (Autonomous), Thanjavur, Tamil Nadu, India.

Abstract

With the rapid growth of digital platforms and online services, ensuring secure authentication has become a critical requirement for protecting sensitive user information. Many web applications rely on password-based authentication systems, but weak passwords and poor security practices often lead to data breaches and unauthorized access. This research proposes a Secure Login System with Password Strength Analyzer designed to improve authentication security by enforcing strong password policies and secure credential storage mechanisms. The proposed system evaluates password strength in real time based on complexity factors such as length, character diversity, and entropy measurement. Additionally, cryptographic hashing techniques such as bcrypt are used to securely store passwords, preventing plaintext exposure in case of database compromise. The system architecture follows a client-server model where user credentials are validated through secure authentication processes. Experimental evaluation demonstrates that the proposed framework significantly improves resistance against brute-force attacks and weak password vulnerabilities compared to traditional login systems. The system also promotes cybersecurity awareness among users by providing real-time password strength feedback and encouraging secure password practices.

Keywords: Cyber Security, Password Strength Analyzer, Authentication, Hashing, Secure Login System, Access Control.

1. Introduction

The rapid growth of internet-based applications has transformed the way individuals interact with digital services. Online platforms are widely used for communication, banking, education, and e-commerce, and these systems rely heavily on authentication mechanisms to verify user identity. Among various authentication techniques, password-based authentication remains the most widely used due to its simplicity and ease of implementation [1].

Studies on user behavior have revealed that many users reuse passwords across multiple platforms and often select passwords that are easy to remember but insecure [2]. This behavior significantly increases the risk of unauthorized access and identity theft. Attackers frequently exploit weak password policies using automated password-cracking tools that perform dictionary attacks or brute-force attempts to guess credentials [10].

To mitigate these risks, modern authentication systems implement password strength policies and cryptographic hashing techniques. Hashing algorithms convert plaintext passwords into irreversible values that cannot easily be decoded even if the database is compromised [3]. Advanced hashing functions such as Argon2 have been designed to resist GPU-based attacks and improve password protection [4].

This research focuses on designing a secure login system that integrates password strength analysis and secure password storage techniques.

2. Literature Review

Bonneau *et al*. conducted a comprehensive evaluation of web authentication schemes and proposed a framework for analyzing authentication mechanisms based on security, usability, and deployability factors. Their research demonstrated that although several alternatives to passwords exist, password-based authentication remains dominant due to its simplicity and cost efficiency. However, the study also highlighted the need for stronger password policies to prevent authentication vulnerabilities [1].

Florêncio and Herley analyzed millions of passwords collected from real-world systems to understand password usage behavior among users. Their findings revealed that many users reuse passwords across multiple websites and frequently choose weak password combinations. This behavior increases the risk of credential compromise, particularly when attackers exploit stolen password databases to perform credential stuffing attacks [2].

Provos and Mazières introduced the bcrypt hashing algorithm as a future-adaptable password protection mechanism. Their work demonstrated that adaptive hashing techniques can significantly reduce the success rate of brute-force attacks by increasing computational complexity. Bcrypt has since become one of the most widely adopted password hashing

***Corresponding Author:** R Nithya

algorithms in secure authentication systems [3].

Biryukov, Dinu, and Khovratovich proposed the Argon2 hashing algorithm, which won the Password Hashing Competition and was designed to resist modern password cracking techniques. Argon2 uses memory-hard computations to make large-scale brute-force attacks more difficult, thereby improving the security of stored passwords [4].

The National Institute of Standards and Technology (NIST) developed comprehensive guidelines for electronic authentication systems. These guidelines emphasize the importance of enforcing strong password policies, protecting against online guessing attacks, and securely storing credentials using salted hashing techniques. NIST recommendations have become an industry standard for designing secure authentication frameworks [5].

The OWASP Foundation provides security guidelines for web application authentication systems through its Authentication Cheat Sheet. OWASP recommends implementing password complexity rules, secure hashing algorithms, login attempt limitations, and user awareness mechanisms to improve authentication security [6].

Weir *et al*. analyzed password composition policies by evaluating how attackers exploit weak password patterns using real-world password datasets. Their research showed that traditional password policies often fail to significantly improve security unless combined with advanced password strength evaluation mechanisms [7].

Blocki *et al*. studied the optimization of password composition policies to balance usability and security. Their research demonstrated that overly complex password requirements may lead to user frustration and password reuse. Therefore, authentication systems must encourage strong passwords while maintaining usability [8].

Verma and Raj proposed a password strength analysis system that uses machine learning techniques to evaluate password complexity and identify weak passwords. Their research demonstrated that intelligent password strength analyzers can significantly improve authentication security by guiding users toward stronger password choices [9].

Narayanan and Shmatikov studied large-scale password cracking attacks and demonstrated how attackers can exploit weak password policies using time-memory tradeoff techniques. Their research highlighted the importance of secure password storage and strong password policies in preventing large-scale password compromises [10].

Reeder, Felt, and Wagner analyzed the security of password managers and authentication tools used in web browsers. Their study showed that password managers can improve security by generating and storing strong passwords, but improper implementation may still introduce vulnerabilities [11].

Kelley *et al*. evaluated password strength measurement techniques by simulating password cracking algorithms. Their study demonstrated that password strength analyzers must consider multiple factors such as entropy, character diversity, and length in order to accurately evaluate password security [12].

3. Existing System

Most traditional login systems rely on simple username and password authentication mechanisms without enforcing strong password security policies. In many systems, users are allowed to register with weak passwords such as short numeric combinations or commonly used words. These passwords can be easily guessed using automated attack techniques, making the system vulnerable to unauthorized access [2]. Another major limitation of traditional authentication systems is insecure password storage practices. Some systems store passwords in plaintext format, which exposes user credentials if the database is compromised. Even systems that use hashing algorithms may still be vulnerable if salting techniques are not implemented correctly [3].

Additionally, traditional systems rarely provide real-time feedback to users regarding password strength. Without such guidance, users often create passwords that are easy to remember but insecure. Attackers exploit these weaknesses using brute-force attacks and dictionary attacks that systematically attempt multiple password combinations [10]. These limitations highlight the need for improved authentication mechanisms that enforce strong password policies and secure password storage techniques.

Limitations of Existing Systems

- Weak password acceptance
- Plaintext password storage
- Vulnerability to brute-force attacks
- Lack of password complexity enforcement
- Limited user awareness regarding password security

These limitations highlight the need for a more secure authentication framework that enforces strong password policies and protects stored credentials.

4. Proposed System

The proposed Secure Login System with Password Strength Analyzer aims to address the security weaknesses present in traditional authentication systems. The system integrates password strength evaluation and secure hashing techniques to improve authentication security. The proposed system follows a client–server architecture in which user credentials are processed through a secure backend server before being stored in the database. During user registration, the password strength analyzer evaluates password complexity based on factors such as password length, uppercase and lowercase characters, numeric values, and special symbols. The system provides real-time feedback to users by categorizing passwords as weak, moderate, or strong. This approach encourages users to create stronger passwords and improves overall security.

Once a password meets the required security criteria, it is processed using the bcrypt hashing algorithm before being stored in the database. Hashing ensures that the original password cannot be retrieved even if the database is compromised [3]. During the login process, the system compares the hashed password entered by the user with the stored hash value to verify authentication. Additional security measures such as login attempt restrictions, session management, and input validation further enhance the security of the system. The proposed system provides a secure and scalable authentication framework suitable for modern web applications and educational platforms.

A. Architecture of Proposed System

The system architecture consists of the following components:

1. User Interface (Frontend)
2. Password Strength Analyzer
3. Authentication Server
4. Secure Database

The system follows a client-server model where user credentials are processed securely by the backend server before being stored in the database.

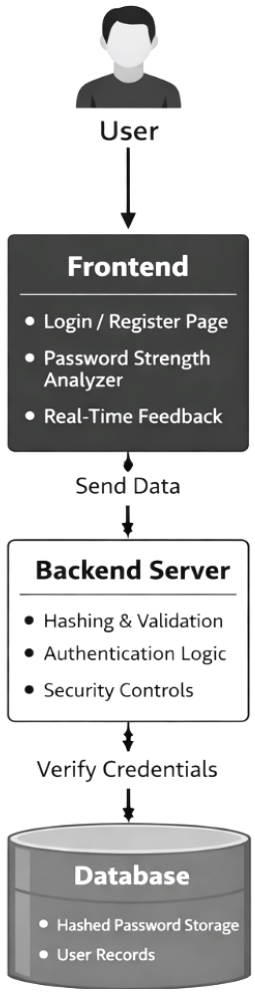

Fig 1: Architecture of Secure Login System with Password Strength Analyzer

B. Key Features of the Proposed System

i) Password Strength Evaluation

The system evaluates password strength based on:

- Password length
- Uppercase and lowercase letters
- Numeric characters
- Special symbols

Passwords are categorized as Weak, Medium, or Strong based on these parameters.

ii) Secure Password Storage

Passwords are stored using bcrypt hashing, which converts plaintext passwords into irreversible hash values.

iii) Authentication Verification

During login, user credentials are validated by comparing hashed passwords with stored hash values.

iv) Security Enhancements

Additional security features include:

- Input validation
- Login attempt restrictions
- Session management

5. Methodology

The methodology of the proposed system focuses on implementing password strength evaluation and secure password storage mechanisms. The password strength analyzer evaluates password complexity using entropy-based analysis and character diversity metrics. Passwords with higher entropy values are considered stronger and more resistant to brute-force attacks [12]. The system also incorporates hashing algorithms such as bcrypt to securely store user credentials. Hashing transforms plaintext passwords into irreversible hash values, ensuring that passwords cannot be recovered even if the database is compromised [3]. The authentication process involves verifying user credentials by comparing hashed values during login. This process ensures that passwords are never transmitted or stored in plaintext form.

A. Password Strength Evaluation Model

Password strength is evaluated using complexity rules.
Example criteria:

- Minimum length ≥ 8 characters
- Combination of uppercase and lowercase characters
- Inclusion of numeric values
- Inclusion of special symbols

Strength Score Calculation

Strength Score = Length + Character Diversity + Entropy Score
Higher scores indicate stronger passwords.

B. Authentication Process

The authentication process consists of:

1. User registration
2. Password strength validation
3. Password hashing
4. Secure storage in database
5. Login verification

Passwords are hashed using bcrypt before storage.

C. System Workflow

User → Login/Register Interface → Password Strength Analyzer → Authentication Server → Database
This workflow ensures that sensitive credentials are processed securely.

6. Experimental Research

The proposed system was implemented using web technologies and tested under different password scenarios to evaluate its effectiveness. The experimental evaluation focused on measuring password strength classification accuracy, authentication verification reliability, and resistance to brute-force attacks. The password strength analyzer successfully identified weak passwords and prevented them from being used during registration. Strong passwords containing a combination of letters, numbers, and special characters were classified correctly and securely stored in the database.
During simulated brute-force testing, the hashed password storage prevented attackers from retrieving the original credentials. Performance analysis showed that the bcrypt hashing algorithm provided strong security while maintaining acceptable authentication response time [3]. These results demonstrate that the proposed system significantly improves authentication security compared to traditional login systems.

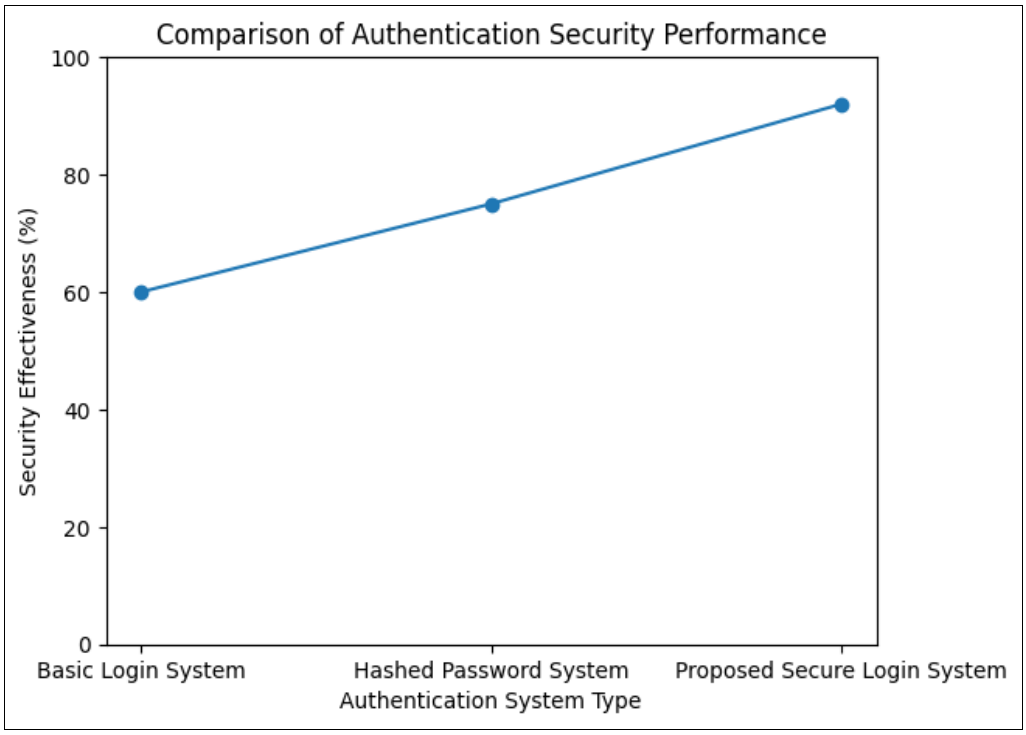

Fig 2: Authentication Security Graph

A. Experimental Setup

The system was implemented using:

- HTML, CSS, JavaScript (Frontend)
- Python/Node.js (Backend)
- bcrypt hashing library
- Secure database storage

B. Evaluation Metrics

The system was evaluated using:

- Password strength classification accuracy
- Brute-force attack resistance
- Authentication verification accuracy
- System response time

C. Results

Experimental results show that the password strength analyzer effectively identifies weak passwords and prevents insecure registrations.

Passwords classified as strong were successfully hashed and securely stored. During simulated brute-force testing, the hashed passwords prevented retrieval of original credentials.

The system demonstrated improved security compared to traditional login systems without password strength evaluation.

7. Summary

Method	Security Level
Basic Login System	Low
Hashed Password System	Moderate
Proposed Secure Login System	High

The proposed system improves authentication security by integrating password strength evaluation and secure hashing techniques.

The experimental evaluation confirmed that the proposed secure login system provides improved protection against weak password vulnerabilities and brute-force attacks. By integrating password strength evaluation and secure hashing techniques, the system enhances authentication security while maintaining usability for users.

8. Challenges

Despite improvements in authentication security, several challenges remain in password-based systems. Users often reuse passwords across multiple platforms, which increases the risk of credential compromise if one system is breached. Additionally, phishing attacks can trick users into revealing their credentials even if the password itself is strong. Future authentication systems may need to incorporate multi-factor authentication and biometric verification methods to further strengthen security.

Despite improvements in authentication security, several challenges remain:

- User resistance to complex passwords
- Password reuse across platforms
- Phishing attacks targeting login credentials
- Need for stronger authentication methods

Future systems may integrate multi-factor authentication to address these challenges.

Conclusion

This research presented a Secure Login System with Password Strength Analyzer designed to enhance authentication security in web applications. By integrating real-time password strength evaluation and cryptographic hashing mechanisms, the proposed system addresses several vulnerabilities present in traditional authentication frameworks. Experimental results demonstrate improved resistance against brute-force attacks and weak password vulnerabilities. The system also promotes cybersecurity awareness by encouraging users to create stronger passwords. Future research may focus on integrating multi-factor authentication, biometric verification, and artificial intelligence–based intrusion detection systems to further strengthen authentication security.

References

1. Bonneau J, Herley C, Van Oorschot PC, Stajano F. The Quest to Replace Passwords: A Framework for Comparative Evaluation of Web Authentication Schemes. In: *Proceedings of the IEEE Symposium on Security and Privacy (SP)*; 2012. p. 553-567.
2. Florêncio D, Herley C. A Large-Scale Study of Web Password Habits. In: *Proceedings of the International World Wide Web Conference (WWW)*; 2007. p. 657-666.
3. Provos N, Mazières D. A Future-Adaptable Password Scheme. In: *Proceedings of the USENIX Annual Technical Conference*; 1999. p. 81-91.
4. Biryukov A, Dinu D, Khovratovich D. Argon2: The Memory-Hard Function for Password Hashing and Other Applications. In: *Proceedings of the IEEE European Symposium on Security and Privacy (EuroS&P)*; 2016. p. 292-302.
5. Burr WE, Dodson DF, Polk WT. *Electronic Authentication Guideline*. National Institute of Standards and Technology (NIST) Special Publication 800-63; 2013. p. 1-67.
6. OWASP Foundation. *Authentication Cheat Sheet*. OWASP Documentation Project; 2023. p. 1-25.
7. Weir M, Aggarwal S, Collins M, Stern H. Testing Metrics for Password Creation Policies by Attacking Large Sets of Revealed Passwords. In: *Proceedings of the ACM Conference on Computer and Communications Security (CCS)*; 2010. p. 162-175.
8. Blocki J, Komanduri S, Procaccia A, Sheffet O. Optimizing Password Composition Policies. In: *Proceedings of the ACM Conference on Economics and Computation (EC)*; 2014. p. 105-122.
9. Verma BGGT, Raj SRR. Password Strength Analysis and Enhancement Using Machine Learning Techniques. *International Journal of Computer Applications*. 2018; 179(32):20-25.
10. Narayanan A, Shmatikov V. Fast Dictionary Attacks on Passwords Using Time-Memory Tradeoff. In: *Proceedings of the ACM Conference on Computer and Communications Security (CCS)*; 2005. p. 364-372.
11. Reeder AGGP, Felt A, Wagner D. Password Managers: Attacks and Defenses. In: *Proceedings of the USENIX Security Symposium*; 2013. p. 1-16.
12. Kelley PG, Komanduri S, Mazurek ML, Shay R, Vidas T, Bauer L, *et al*. Guess Again (and Again and Again): Measuring Password Strength by Simulating Password-Cracking Algorithms. In: *Proceedings of the IEEE Symposium on Security and Privacy*; 2012. p. 523-537.