IJIRSET
International Journal of Innovative Research in
SCIENCE | ENGINEERING | TECHNOLOGY

e-ISSN: 2319-8753 | p-ISSN: 2347-6710

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

Volume 13, Issue 11, November 2024

ISSN INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 8.524

9940 572 462 | 6381 907 438 | ijirset@gmail.com | www.ijirset.com

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.524

|www.ijirset.com |A Monthly, Peer Reviewed & Referred Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 13, Issue 11, November 2024

|DOI: 10.15680/IJIRSET.2024.1311138|

Password Manager

Gubba Abhiram[1*], Nooneti Manohar[2*], Somisetty Badrinath[3], E.Radha Krishnaiah[4]

Student, Department of Computer Science and Engineering, Anurag University, Hyderabad, India[1,2,3]

Department of Computer Science and Engineering, Anurag University, Hyderabad, India[4]

ABSTRACT: In today's day and age of our rising digital bandwidth, password management has become an absolute must not just for preventing hackers but also for securing your personal information. The Password Manager project provides a strong web based solution to the challenges above; providing it as secure, automated and user friendly way to manage the various passwords. It is built on HTML, CSS, JavaScript for the front side, Java and Java Server Pages (JSP) for the back end and MySQL for storage of secure credential. The main goals of this project are to improve security of a password, improve or simplify the management of a password, and decrease the risk of using a weak or used password.

Strong password generation, automated 24 hour password updates, encrypted credential storage – all features meant to reduce human error and minimize potential sources for password related security vulnerabilities. The module responsible for generating system's password creates complex passwords consisting of alphanumeric characters with symbols and meets to modern security standards. Having encrypted all credentials before storing, Advanced Encryption Standard (AES) key is used to protect the user data from unauthorized access. The Password Manager makes updating your passwords a manual feature and enables you to modify passwords in a safe manner whenever you need it. To enhance security even further, this project uses SSL/TLS for secured transport of communication from client to server.

Functionality, usability, performance, and overall robustness under different conditions was extensively tested. The assessment of the Password Manager results shows the increase in its scope of functional completeness towards the main requirements such as storing and retrieving passwords securely, and the ability to automate their updates, and to manage users. There are future work by the integration of multi factor authentication, mobile application development and cloud based syncing to expand device access. As a complete solution for the management of secure and reasonable passwords, this project resolves common problems like password tiresome, rehash, and password behavior, part of the objective at securing users in general digital security.

An application written primarly in Java, JSP, and MySQL for a web-based software, using .

KEYWORDS: Password Manager, AES encryption, automated password update, secure password generation.

I. INTRODUCTION

The era of digitalization has caused many people to have many online accounts that require different and secured passwords to secure their data. Nevertheless, issue of creating, storing, and updating these credentials frequently has resulted in common security risks, such as weak or reused passwords. Many users however prefer to simply memorise or use an easy password, which can result in accounts being exposed to the cyber threats like brute force attack and phishing. There has been existing password management solutions, like LastPass, Dashlane, and 1Password, which provides basic features ranging from secure storage and password generation but unfortunately they don't provide flexibility in automated updates and may not cover the full scope of user needs in terms of security and ease of use.

This Password Manager project overcomes these limitations by providing a complete and user friendly software that generates strong passwords, securely stores them and automates password update without user interaction. This system is built as web applications using HTML, CSS, and Javascript for the frontend, and Java and Java Server Pages (JSP) for its strong backend. Encrypted password storage dictates that the database used to secure the information will also be resilient to unauthorized access, so MySQL is the database of choice here. Their solution is based on AES for storing passwords and SSL/TLS for protecting data 'in transit'. This paper discusses the motivation of the project, its main objectives and key features, and technical implementation of an efficient password manager that enhances user security practice.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.524

|www.ijirset.com |A Monthly, Peer Reviewed & Referred Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 13, Issue 11, November 2024

|DOI: 10.15680/IJIRSET.2024.1311138|

II. RESEARCH METHODOLOGY

This Password Manager research methodology deals with the security and automation of the approach of credential management. Finally, the system is developed based on Model-View-Controller (MVC) architecture that implements the concern in the data storage (backend layer), user interface (view layer) and logic layer (model). The frontend, HTML, CSS and Javascript, while the backend is done with Java, JSP, and MySQL is used for securing the data in storage by using AES encryption. Automated password updates and encryption are validated by functional and performance testing. Using this approach guarantees that the system is reliable, obtainable for everybody, and scalable across several use situations.

Hardware and Software Configuration:
Specific hardware and software capabilities exist to achieve a secure and efficient operation of the Password Manager system. The specs are that you need an Intel Core i5 processor or better, 8 or more GB of RAM (we recommend 16), and 256 GB of SSD (or 500 GB or more HDD). For Windows 10 or later, macOS Catalina or later or any recent Linux distribution, with resolution of 1366 x 768 or more. The frontend is written in HTML to define how the content is laid out, CSS to style in, and JavaScript for the client side interactivity of the page.

The backend is written using Java and JSP (Java Server Pages) to control how user requests are handled on the server and also to execute server side logic. The web server is Apache Tomcat and hosts the JSP pages and Java Servlets. MySQL is used as the database with credentials AES encrypted and JDBC connectivity to the backend. It is developed in Eclipse IDE for Java and JSP while for frontend coding it is used Visual Studio Code. The advantage of configuring it in this way is that we have a strong, responsive and secure environment for managing user credentials properly.

Software Development
But there is a progressive nature of the development of the Password Manager system respecting to perform both frontend and backend components to be secure, efficient and user friendly. Here we describe the major stages and technologies in software development process from design to implementation and testing.

Frontend Development : To make the Password Manager frontend experience intuitive and responsive it is built using HTML, CSS, and JavaScript. HTML is what makes unsophisticated web pages structured and allows you to organize forms and tables to display and manage credentials. The CSS is used to style the application and provides a visually appealing layout and adapts nicely to different devices having different screen sizes. Since JavaScript is used for client side operations, password generation, validation and user interaction is handled with it. Suppose JavaScript is used to create a strong password containing alphanumeric characters and symbols, whose user customization is allowed. Also with design tailor made for the frontend we prioritize accessibility so that the interface is intuitive for users regardless of their abilities.

Backend Development : For all the server side logic, data processing and user authentication, the Password Manager backend is created using Java and Java Server Pages (JSP). Based on java, it offers secure and robust environment for implementation of must have business logic for such feature as password generation, encryption etc., will allow the dynamic updates. Dynamically generating web pages on server side logic and information interaction with user interface and backend in real time through JSP. Through this setup, we make sure that sensitive operations like password encryption and storage are done securely. Java Servlet also processes HTTP requests and responses, so that the frontend and backend components can follow.

Database and Data Security : Relational database to store user credentials — usernames, websites, and passwords, securely — is MySQL. By using AES (Advance Encryption Standard) to encrypt passwords before they are stored on the database, we not only protect users against Unauthorized access to the sensitive data but also employ a layer of security. Java Database Connectivity (JDBC) presents a way to connect the Java backend to MySQL database for retrieving and storing data as well as update the data. The secure storage combined with encryption means user's credentials are out of reach in the event of a potential breach or unauthorized access.

Development Environment and Tools : The development process is streamlined with the use of two primary integrated development environments: Java and JSP Eclipse IDE for backend development and Visual Code for front end development in HTML, CSS and JavaScript. The application server is Apache Tomcat which hosts JSP pages and

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.524

|www.ijirset.com |A Monthly, Peer Reviewed & Referred Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 13, Issue 11, November 2024

|DOI: 10.15680/IJIRSET.2024.1311138|

Java Servlets so that it can be accessed through a web browser. Taken together, these tools allow the construction of a secure, scalable, and user friendly application while maintaining reliability and preservation of performance across a range of use cases.

Testing and Validation: The Password Manager undergoes well defined testing for verification of functionality, performance and security. We perform functional testing to make sure all features—such as password generation and storage and retrieval—function as expected. The purpose of the performance testing is to assess how system responds, and how it scales under various load condition to check if the application can efficiently deal with concurrent users. Validation of encryption, secure session management and input validation protects user data using the security testing technique. However, these stages are extremely crucial in order to deliver such a high quality product, that meets user's expectation for security and usability, to the market.

The software development practices follow a structured approach and then its tested robustly and data handled securely in order to see the Password Manager as the system that helps in managing passwords in a secure and efficient way.

III. RESULTS AND ANALYSIS

We have done many tests to ensure that it is working and Secure. They test whether the system is securely able to create, hold, update and use passwords without degrading responsiveness and usability for end users. With automated password updates, secure encryption, and a user friendly interface, the results show that the Password Manager achieves the desired functionality in efficient credential management.

Functional Testing

A functional test was performed to ensure that each feature does what is expected and falls in line with design requirements. Password generation, credential storage, password updates and secure login are all key test cases. The generation of passwords feature generated strong, randomly generated passwords containing mixed characters, numbers, and symbols. Credentials were successfully stored, with data encrypted with AES encryption inside the database. Password update function was automated and trigged every 24 hours without user interference. Invalid credentials caused strong error handling, preventing the session and login to be authentic on a user. The core features of the system were validated through all the functional tests; they went all green!

Performance Testing

The responsiveness, scalability, and the efficiency of the resources of the system were performance tested. Even when dealing with high complexity passwords or large number of records, password generation and storage operations completed in 200-400 ms. Login processes had similarly responsive response time under 400millisecondeven peak load condition. The system proved scalable, able to support up to 500 concurrent users without significant degradation in response time, providing a reliable performance for many users. CPU and memory usage were stable during high loads, approaching 30 to 50 percent, while the system proved that it could handle increased demand without sacrificing performance.

Security Testing

That sensitive data meant the security testing really mattered. With the Password Manager, we made sure that passwords are securely encrypted, and there's no plaintext passwords in the database. Data in transmission was secured using security features: encrypted communication using SSL/TLS and secure sessions. SQL injection and cross-site scripting (XSS) attack input validation mechanisms existed to avoid attacks on the application and prevent common security threats. We ran security alerts for multiple failed login attempts and they found users, hence proactive threat detection.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.524

|www.ijirset.com |A Monthly, Peer Reviewed & Referred Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 13, Issue 11, November 2024

|DOI: 10.15680/IJIRSET.2024.1311138|

Summary of Results

Functional, performance and security tests show high performance regardless of password complexity, and security is high too. Table I summarizes the core test results:

Test	Objective	Outcome
Password Generation	Generate strong, random passwords	Pass
Credential Storage	Securely store and encrypt credentials	Pass
Automated Password Update	Update passwords automatically every 24 hours	Pass
Login and Authentication	Validate secure login and session management	Pass
Scalability and Performance	Handle 500 concurrent users	Pass
Security Measures	Prevent unauthorized access and encrypt data	Pass

Analysis and Insights

The Password Manager succeeds at reaching these design goals of offering a secure, effective, and user friendly credential manager. With heightened security features to ensure that data remains protected and a robust performance that scales the more you use it, the systems ability to perform at scale is reliable. The system was robust and though performance was good, further caching and load balancing optimizations would improve performance greatly under very high loads. Future improvements can include better security via more advanced encryption algorithms, and features such as multi factor authentication (MFA) to prevent unauthorized access.

Test results show the Password Manager can be relied upon to cope within real world use cases and provide users a useful means to securely manage their passwords. This result thus confirms that the system design and its implementation are effective and constitutes a solid footing for continuing development and future scalability in enterprise contexts.

IV. CONCLUSION

Its ability to maintain secure and efficient credential management is the dedication of today's digital world which has made the Password Manager a success in addressing the critical need. The system reduces the widespread vulnerabilities of passwords that are reused and that are based on poor password choices, by automating password generation, storage, and periodic update. The Password Manager secures user data through robust AES encryption and secure session handling so that it’s not accessible to any unauthorized users. The reliability and scalability are confirmed with functional and performance testing, which showed that the system can handle multiple users and concurrent operations well. Furthermore, the interface is user friendly, even for persons who are not technical can move and manage their credentials.

Overall the project meets its primary objectives of improving security, reducing user load and increasing usability in password management. The present implementation is adequate for single use, however additional features like multi factor authentication (MFA), cloud synchronization, and integration with mobile apps can enhance its range of application in future. Taken as a whole, the Password Manager is a pragmatic, secure, and convenient digital credential management tool, on which to build the next wave of personal and organizational security.

V. DISCUSSION AND FUTURE SCOPE

The Password Manager successfully addresses some of the issues of secure credential management in today’s electronic world. Automating password generation, storage and updates reduces the most common risks, like weak passwords or password reuse, which are too often exploited during cyber attacks. Since AES encryption is used to store passwords, it strongly protects the data and thus offers minimal vulnerability of the data breach. This makes it harder for passwords to be vulnerable for more time without any sound layer of security – automated password updates. The system also keeps a priority on keeping things simple and secure for a simple user, where any person without technical knowledge can easily manage their credentials.

The Password manager uses HTML, CSS, JavaScript, Java, JSP, and MySQL to make thing versatile and scalable. Both functional testing, that ensures the features are reliable such as generating and storing passwords and performance

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.524

|www.ijirset.com |A Monthly, Peer Reviewed & Referred Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 13, Issue 11, November 2024

|DOI: 10.15680/IJIRSET.2024.1311138|

testing showing that the system can handle concurrent users are successfully conducted. However, we have several limitations. For instance, login credentials have to be manually entered into the system, and can never be synchronised over multiple devices or browsers, thereby limiting the user's flexibility.

FUTURE SCOPE

1. Multi-Factor Authentication (MFA): MFA for logging in would reinforce user security through a secondary check which includes SMS codes or email verification of user identity. This feature will add to the protection of the account from unauthorized access.
2. Cloud-Based Synchronization: Securing cloud storage would solve introduce the concept of securely syncing credentials across multiple devices. It would add to that functionality by allowing user to access stored passwords from any authorized device and without compromising data security.
3. Mobile Application: With the Password Manager you could have a mobile version that would give you the ability to access and keys from the go on your phone. The mobile application can, additionally, be further secured with biometric authentication, for example fingerprint or facial recognition.
4. Browser Extensions: Browser extensions that can be developed for the most popular web browsers (Chrome, Firefox, Safari) can autofill login fields from the Password Manager directly and remain convenient without compromising security.
5. Password Strength Analytics: In a future version an analytics might be made to spot the utilization of weak passwords, or password reuse, and notify the users of such problems with their stored credentials. It would help secure the site by forcing users to make their passwords as weak as possible.
6. Advanced Encryption and Security Audits: To further protect stored credentials, one could implement more advanced encryption algorithms or integrations with hardware security modules (HSMs). The system may also contain periodic security audits to look for breaches, and recommend password changes as necessary.
7. Role-Based Access Control (RBAC) for Enterprises: Sowing the system to establish role based access will empower businesses to decrease away with group lead credentials securely. The Password Manager would have features such as audit logs and access permissions making it suited for organisation use.

REFERENCES

1. LastPass, "Password Management & Security," available at: https://www.lastpass.com
2. Dashlane, "Secure Password Manager & Digital Wallet," available at: https://www.dashlane.com
3. 1Password, "Password Manager for Families, Businesses, Teams," available at: https://1password.com
4. Bitwarden, "Open Source Password Management Solutions," available at: https://bitwarden.com
5. K. Mitnick and W. L. Simon, *The Art of Invisibility: The World's Most Famous Hacker Teaches You How to Be Safe in the Age of Big Brother and Big Data*, New York: Little, Brown and Company, 2017.
6. T. O'Gorman, "Comparative Analysis of AES and RSA Encryption Algorithms," *International Journal of Computer Science and Network Security*, vol. 11, no. 2, pp. 1-7, 2022.

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi cross ref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

9940 572 462 6381 907 438 ijirset@gmail.com

www.ijirset.com

Scan to save the contact details