CryptoChronicles

LessPM's Quest to a Passwordless Utopian Ecosystem

T-742-CSDA

Håvard Nordlie Mathisen
Reykjavik University
havard22@ru.is

Instructor
Jacky Mallet

Department of Computer Science
Reykjavik University
Match, 2023

Executive Summary

Test[**lol**]

Abstract

Introduction

Polybius' *The Histories* [5] contains the first documented use of passwords, describing how the Romans employed “*watchwords*” to verify identities within the military. This provided a transparent, simple way to allow or deny entry to restricted areas of authorized personnel only. The story of secret writing (in this context referenced as cryptography) goes back the past 3000 years [2], where the need to protect and preserve privacy between two or more individuals blossomed.

Fernando J. Corbató is widely credited as the all-father of the first computer password when he was responsible for the Compatible Time-Sharing System (CTSS) in 1961 at MIT [4]. The system had a `"LOGIN"` command, which, when the user followed it by typing `"PASSWORD"`, had its printing mechanism turned off to offer the applicant privacy while typing the password [1]. Given the long history of passwords and their importance, one could argue that it was a natural and judicious step in the evolution of computer systems.

In today's digital landscape, utilizing various identifiers (such as usernames, email addresses, or phone numbers) combined with passwords has become a prevalent method for verifying an individual's identity and ensuring their authorization to access restricted materials.

In 2004, a study titled *"The Memorability and Security of Passwords"* [7] was conducted into advising users on the entropy of passwords and ways someone can use to remember a or multiple passwords. A typical standard for larger organizations with a form of password creation system is to emphasize the diversity of smaller characters, capitalized characters, length, and not be commonly referred to in a dictionary [7]. The study analyzed the effectiveness of different password-creation strategies, suggesting that acronym-based passwords offer a delicate balance between memorability and security[7].

However, as technology has advanced, the limitations of password-based authentication have become increasingly apparent, leading to the development of more sophisticated methods like Universal Authentication Framework (UAF) [3] and WebAuthn[6] through the Fast IDentity Online Alliance (FIDO) and The World Wide Web Consortium (W3C).

WebAuthn, short for Web Authentication, is an open standard for web-based authentication that enables users to securely access online web services without relying on a traditional password. Through a collaborative effort between FIDO and W3C, WebAuthn is developed to leverage asymmetric cryptography[1][2] and biometric or hardware-based authenticators to provide a more secure and robust authentication experience.

This report delves into the implementation of LessPM, a password manager that leverages WebAuthn to provide a secure authentication experience, free from the constraints of traditional passwords, while placing a strong emphasis on security. By examining recent advancements in authentication mechanisms and the related innovative potential of WebAuthn, we hope to illuminate the prospects of a passwordless future in digital security.

Methodology

Conclusion

Future work

Through implementing LessPM, we aimed to create a barebone implementation that could serve as a reliable Minimal Viable Product (MVP). However, we recognize that more work is needed to further enhance and compliment the product. The related topics to further improve LessPM are listed below and briefly discussed as a way to highlight potential drawbacks of the current version.

- **Authorization Headers**
- **AES Key Encryption for Password**
- **Hardcoded AES for JWT**
- **Encrypted Passkey**

[1] Asymmetric cryptography uses a key pair consisting of a public key and a private key. The public key encrypts data, while the private key decrypts it. The keys are mathematically related, but deriving one from the other is infeasible, ensuring secure communication and data exchange.

[2] **Note:** According to the library used to implement jsonwebtokens in Rust it is the private key that encrypts and the public key is responsible for decrypting. *Last Accessed: 2023-03-25*.

References

[1] *CTSS Programmers Guide*. 2nd ed. MIT Press, 1965.

[2] John F. Dooly. *History of Cryptography and Cryptanalysis: Codes, Ciphers, and Their Algorithms*. Springer, 2018, p. 5. ISBN: 978-3-319-90442-9.

[3] FIDO Alliance. *FIDO UAF Overview*. https://fidoalliance.org/specs/fido-uaf-v1.1-id-20170202/fido-uaf-overview-v1.1-id-20170202.html. Accessed: 2023-03-25. Feb. 2017.

[4] Steven Levy. *Hackers: Heroes of the Computer Revolution*. Sebastopol, CA: O'Reilly Media, 1984, pp. 85–102. ISBN: 978-1449388393.

[5] Polybius. *The Histories*. Accessed on: March 23, 2023. 2023. URL: http://www.perseus.tufts.edu/hopper/text?doc=Perseus%3Atext%3A1999.01.0234%3Abook%3D6%3Achapter%3D34.

[6] World Wide Web Consortium. *Web Authentication: An API for accessing Public Key Credentials Level 2*. https://www.w3.org/TR/webauthn-2/. Accessed: 2023-03-25. Apr. 2021.

[7] Jianxin Yan et al. *The Memorability and Security of Passwords: Some Empirical Results*. Tech. rep. UCAM-CL-TR-500. University of Cambridge, Computer Laboratory, Sept. 2000. URL: https://www.cl.cam.ac.uk/techreports/UCAM-CL-TR-500.pdf.