International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320-088X

IMPACT FACTOR: 7.056

IJCSMC, Vol. 10, Issue. 5, May 2021, pg.5 – 14

An Expedient Way to Organize Passwords

M S Sowmya¹; Anvita Ranka²; Abhishek Mohanty³; K Nanda Kishore⁴

Faculty of Engineering and Technology, Jain University

mssowmya.sbmjce@gmail.com¹; kishorekaranam99@gmail.com²; abhishakeit8@gmail.com³; anvita.ranka@gmail.com⁴

DOI: 10.47760/ijcsmc.2021.v10i05.002

Abstract— There are billions of passwords in the world today, with more being created every hour of every day, making them a bane of the modern era. The usage of passwords is becoming more common despite the inadequacies [1]. This is making it more tedious and overwhelming for the users to manage, with the passwords being long and unique. In this paper, we propose a solution to securely manage and organize passwords while requiring the user to keep track of only a single password. As our solution is in the form of a browser extension, there is no need for server-side changes. Unlike other password managers, our extension is a lightweight application and is highly resistant to brute force attacks. We discuss the need for a password manager, the construction of our extension along with the security overview.

Keywords: Chrome Extension, Password Security, Password Manager.

I. INTRODUCTION

With the abundance of personal data stored on social media sites, it has become a tough task to be able to remember all the various passwords that are required while creating accounts on those respective websites. With more of the population now working from home, outside the office intranet, the number of passwords needed might have significantly increased. With the increasing workload, it becomes an additional hassle that most of us are not prepared for. Using

the same or similar password on many different systems increases the risk of compromising the password on all systems if one is accidentally or intentionally shared [2]. Thus, better measures are required.

II. LITERATURE SURVEY

We chose this aspect of technology as our area of research because password memorability is a problem that seems to be getting worse as the numbers grow. Everything in life can now take place online, and thus it is essential to look into potentially the most overlooked issue occurring from using the internet.

A. PASSWORD MANAGERS

In the very likely scenarios where multiple passwords are required, the vulnerabilities that accompany them can be attempted to be mitigated through password managers. Often, the basic working of these software programs is to employ a singular master password to store multiple passwords in encrypted files. These programs also have the ability to generate safe passwords which decreases the risk of password reuse [2]. Even though password managers could solve many of the security issues regarding multiple passwords they seem to be rarely used [3]. The main area of research regarding password managers is the technical design and technical security [3].

B. MANAGING MULTIPLE PASSWORDS

Having multiple passwords can cause even more security issues with regards to security. Using the same or similar password on many different systems increases the risk of compromising the password on all systems if one is accidentally or intentionally shared [4]. Users, time and again, engage in improper password management actions, primarily due to fact that such supervision decreases convenience factor, paired with inability to observe immediate threats and consequences of using weak passwords [5].

C. AUTHENTICATION AND PASSWORDS

Authentication can be divided into four categories; something the user knows, for example, a password; something the user possesses, for example, a smart card; something the user is, for example, static biometrics like fingerprint sensor; something the user does, for example, dynamic biometric like sound recognition. Because the user is involved in creating and managing these passwords, the biggest security issue with this type of authentication is the human factor [6]. In summary, multiple password environments create security issues regarding weak passwords, password reuse, loss of productivity and cost of maintaining a service desk for password related issues, password notes and sharing. Having users remember more than four to five passwords causes them to forget the password and therefore increasing the risk of engaging in poor password management behavior [1].

III. CURRENT LIMITATIONS

Based on our literature survey, we identify one clear problem – being able to direct, control and preserve passwords in this era of digitalization. From the contexts above, we clearly can understand why proper management of multiple passwords is of utmost importance. This supervision and handling can be made efficient if automated through the use of password managers, since authentication and explicit memorization of multitude watchwords by human beings is a tedious, ineffective and an exhaustive task. However, presently functional password managers lack the depth and focus upon the security aspect of storing passwords as they are primarily devised to simply accumulate user passwords.

Furthermore, on supplementary analysis on password behavior online, we recognize the most popular (i.e., the poor) passwords of 2019, which were ‘12345’, ‘123456’ and ‘123456789’. Other notable mentions included ‘password’ (coming in at 5) ‘qwerty’ (10) and ‘iloveyou’ (7). Paired up with the fact that over 85 percent of people reuse singular passwords among different accounts and sites, it becomes clear that users are still not taking effective measures to protect themselves online.

It is also not advisable for different passwords to be used for various accounts, and it becomes even tougher to remember unique ones, given the constraints imposed upon it, which

include uppercase and lowercase letters, numbers, symbols and special characters. Keeping passwords manually written into a plain file on the system becomes too risky too, since anyone with access to the system can easily acquire the credentials to another user's accounts, and in case of data corruption, the passwords are completely lost. Additionally, in case of the rudimentary password managers offered by web browsers, they lack the dedicated focus on password security, since they have their priorities set elsewhere.

It's important to have a good password for one very simple reason: it prevents unauthorized access to physical devices and online accounts. If a password is easy to crack, a cybercriminal may be able to gain access to banks, social media, email and other private accounts of the users, which could have a devastating effect on their life.

IV. OBJECTIVES

- We try to create a tool that can provide secure passwords for all the websites for which accounts exist for the user.
- The tool does not store any of the generated passwords anywhere, i.e., online or offline, which aims to bolster the aspect of security since data cannot be accessed if it doesn't appear to exist.
- The automation absolves the users from being stuck thinking of the passwords, since all of this work would be handled by the system of this project.
- The tool would also autofill the respective login credentials automatically as a website is opened, so the user may only have to click on the "login" button, and let the tool do the hard work.

V. METHODOLOGIES

- To build a password manager, initially we built a front end of the tool with all it's required fields utilizing HTML, CSS and JavaScript.
- We then contrive it as an external extension that must be loaded in unpacked fashion onto the browser through a json file.

- A python script is included within the entire system. This script fetches the current URL being visited, and through the sorcery that is one-way-hashing techniques, the script uses the obtained web address to generate a digital fingerprint, which serves as the password for the user.
- As a secondary procedure, we also sustain a deep learning model, that is trained on different patterns occurring in passwords to learn and generate human readable passkeys.

VI. PROPOSED BLOCK DIAGRAM

We highlight a simple logical diagram, shown below, as a means to objectively define the workflow of the system through an example.

Explanation: The mail URL is pulled in, where it is used, in collaboration with the “master” password and mentioned hash function to give rise to a message digest. This digest is then passed through filter like settings that transform it into a bona fide, acceptable and strong password.

FLOWCHART

VII. IMPLEMENTATION

A. BASIC STRUCTURE

As the rise of Web APIs has become unparalleled, web apps have become far more powerful than platform-specific applications [3]. Based on this for the sake of development being easy and the end result being more effective, we decided to build the password manager in the form of a browser extension, primarily focusing on Chrome. The reason for going with such a decision is because of the vast number of tools available with chrome and also because of the huge number of users using chrome [4]. However, the limitations of Chrome have led to users shift massively to other browser options, including Firefox, Opera, Brave, Safari, Edge, Tor and so on, and thus we have taken steps to ensure that our tool works relatively well on all popular browsers. Steps to give rise to the structure of the tool started with creating a manifest file is essentially the base and foundation of our system.

B. SECURITY OVERVIEW OF THE EXTENSION

Initially, our course of action was to set up a vault to store all generated passwords there. This feature would utilize IndexedDB to save and query passwords through it's unique keys. However, the fallacy still would stand, since stockpiling sensitive data anywhere, be it online or offline, does pose a threat. Both areas of storage have their pros and cons, but are unfortunately not fool proof. Thus, we decided against storing passwords anywhere at all. The script is coded in such a way that no external force can determine any information about the input used to generate the digital signatures, and even if someone comes to obtain any of the generated passwords by the system, it is computationally infeasible for your master password to be derived through the backward calculation of other generated passwords. Therefore, we empower the security aspect of our project but getting rid of password storage entirely.

C. UNIQUE PASSWORD GENERATION FEATURE

An additional feature included is the option of generating a fresh password through a deep learning model, which stands as a secondary “failsafe” option to the primary method. Passwords, at their core, are nothing but a sequence of alphanumeric characters and symbols, where they are more complex and deemed secure if the sequence is haphazard and if it matches

the conditions imposed upon the composition of the password structure as per each site accordingly. With the powers of modern Natural Language Processing techniques combined with state-of-the-art neural network architectures, we implement a system that outputs such sequences of characters that serve as passwords. We do this by allowing a neural network to "learn" or generalize the characteristics of strong passwords, which in turn can then replicate those patterns and give out new passwords. In order to achieve this, we make use of the Long Short-Term Memory (LSTM) network, a variant of the Recurrent Neural Network structure. This is because such a network architecture specializes in finding patterns in text sequences and figuring out the features within it, which makes it a very suitable algorithm for us to employ in order to extract information as to what exactly makes a password strong and secure. In hindsight, Deep Learning Web Applications in browsers have already gained traction in industries and research, giving rise to various JavaScript-based DL frameworks and libraries which allow several deep learning tasks to be formed directly in the browser without requiring the resources that would be taken up in a standalone system or engine [8]. Through our implementation of a password generator, we look only to further add to the functionality our Manager provides, as well as serve as a basis for it to be scaled up to larger sized applications.

VIII.RESULTS

We have successfully created our system. The password manager, named "EZ Creds", works as intended – there exists a settings handle which the user can tweak to their preference, the password generation feature operates without any issues for a ton of different websites. The tool allows different profiles to be set up with varying master passwords. The additional deep learning model is also functional as an add on. Lastly, the autofill feature too is performing as intended.

IX. CONCLUSION AND FUTURE SCOPE

Despite being one of the most underrated pieces of software, password managers are vital these days as more than ever people are in great need of them with privacy being very little. When it comes to the future enhancements of this extension, we can try to bolster parameters of the tool further. As one of the main functions of the browser extension is the Auto-Fill function and the password generation, we can add that feature to make things better and more efficient. Additionally, we could also add in functionalities to add in important notes, documents, bank-related files. In conjunction with this, we could also add a timer function that alerts the user if any password should be altered or changed based on how long back the password was created so that cycling through passwords on a regular basis prevents attacks even more.

REFERENCES

- [1]. Sonia Chiasson, P.C. Van Oorschot, and Robert Riddle, A usability study and critique of two password managers, USENIX-SS'06, July, 2006.
- [2]. Paolo GastiKasper, B. Rasmussen, Security of Password Manager Database Formats.
- [3]. Furnell.S, Jushoh.A, Katsabas.D, The challenges of understanding and using security: A survey of end-users.
- [4]. Naomi Woods. Frequently Using Passwords Increases Their Memorability - A False Assumption or Reality.
- [5]. Tam, L., Glassman, M., Vandenwauver, M. The psychology of password management: A tradeoff between security and convenience
- [6]. Rui Wang, Shuo Chen, XiaoFeng Wang. Signing Me onto Your Accounts through Facebook and Google: A Traffic-Guided Security study of Commercially Deployed Single-Sign-On Web Services.
- [7]. AlessandroArmando, RobertoCarbone, LucaCompagna. From Multiple Credentials to Browser-Based Single Sign-On: Are We More Secure?

ACKNOWLEDGEMENT

This research is supported by Jain University, Faculty of engineering and technology. We are thankful to our Professors of department who provided expertise that greatly assisted this research, although they may not agree with all of the interpretations provided in this paper.

We express our heartfelt appreciation towards our Guide, Ms. M S Sowmya, Assistant Professor, Dept of ISE, FET-JU, for sharing her pearls of wisdom with us during the course of this research. We are also immensely grateful for her comments on earlier versions of the manuscript, although any errors are our own and should not tarnish the reputations of the esteemed professionals.