Jarrod Saffioti
CIS 129
Professor Griwzow
May 8, 2024

Final Project

Problem identified: Password leaks and possible protection against them

Existing solutions: Password managers seem to be the most common solution to this problem. Although they are not infallible, they use very sophisticated encryption, and create a different password for every website. This makes it very difficult for someone's true password to be identified in the event of a data leak. If used correctly, robust password managers can be very effective. Of course, the password managers themselves could be breached, but that would be quite the task for even the most skilled hackers. "Passkeys" have also arisen as a potential alternative to passwords. They use a public key that is stored by the website, and a private key that is stored on one's device. By storing the information in separate places, it is much more difficult to bypass.

Justification: My mother got her identity stolen in a password breach, so I want to learn more about how to combat that in the future. Pretty much everything these days is digital, so being safe online is of utmost importance. I want to see what can be done to make using the internet safer, starting by having secure passwords.

Proposed software solution: My thought was to use a password that constantly changes. There would be a manager service that keeps track of the password and changes it to something else every hour, let's say. This would require some sort of communication between the website and password manager, but I think it would greatly increase the security of passwords. Randomly generated passwords are already used by popular password managers, but I think having them automatically change would further increase their security. With this system, even if a password became compromised it would no longer have any use or meaning within an hour.

Pseudocode:

```
Declare password variable

Use a function to set the password variable equal to a random string generated by an encryption algorithm

Declare timer variable
```

Set the timer variable equal to the current time

Use a timing function to keep track of when an hour has passed. Once it has, re-call the password function to generate a new password and assign it to the password variable

User interaction: There wouldn’t be a whole lot of user interaction, but basically the user would just have to sign up and then allow the password manager to take care of their passwords. They would be able to see the current passwords that they have stored and when they would update next.

References:

Constantin, L. (2017). LastPass password manager fixes serious password leak vulnerabilities. *PCWorld, 35*(5), 59–60.

Fleishman, G. (2023). Make Your Passwords Indestructible. *Macworld - Digital Edition, 40*(5), 58–62.

Leaks, damned leaks: another nail in the coffin of pathetic passwords. (2012, July 26). *The Independent (London, England)*, 36.

6 tips to protect yourself against data leaks. (2019, January 22). *Philippines Daily Inquirer (Makati City, Philippines)*.

Compromised passwords: Impact and 6 ways to prevent compromise. *Exabeam*. (2023, May 11).