Human Interaction and Emerging Technologies (IHIET 2024), Vol. 157, 2024, 105–115
https://doi.org/10.54941/ahfe1005469

Exploring the Risks of Password Reuse Across Websites of Different Importance

Anurag Mathews[1] and S M Taiabul Haque[2]

[1]Computer Science, Bergen County Technical High School, Teterboro, NJ 07608, USA
[2]Department of CSE, BRAC University, Merul Badda, Dhaka 1212, Bangladesh

ABSTRACT

This study attempts to simulate the different ways through which a malicious hacker may attempt to gain unauthorized access to user accounts by leveraging the similarities between multiple linked passwords of the same user. The issue of managing multiple password-protected accounts exemplifies the usability/security trade-off in cybersecurity. Users often reuse the same password, with little or no modifications, across websites of different importance, compromising the security of the high-value accounts. By combining syntactic similarity, dictionary attack, service-related keywords, and semantic similarity on a set of 62,213 linked passwords available from the leaked databases on the internet, 82.3% of the high-value passwords were cracked with an average of 1.82 seconds spent on each attempted password. Similarly, the syntactic method alone achieved an accuracy of 73.6% at 0.82 seconds spent per password attempted. We further connect our findings to the broader issues in cybersecurity and offer a few suggestions to protect the high-value accounts of the users.

Keywords: Usability, Security, Authentication, Password, Hierarchy

INTRODUCTION

Despite being the most popular method of user authentication on the internet, textual passwords have several usability issues that become more pronounced due to the existence of multiple password-protected accounts. A recent study reveals that an average user may need around 100 passwords to manage across various accounts (Sicora, 2022). According to Miller's law, however, a typical user can retain about seven items in working memory (Miller, 1956). Consequently, users reuse the same password, with little or no modifications, for multiple accounts (Haque, Wright and Scielzo, 2014).

The existing literature on password security and usability has examined password reuse from different viewpoints, but very few studies have adopted the nuanced approach of formulating a user password hierarchy to investigate this issue. This hierarchy was first proposed and then extended by Haque, Wright and Scielzo (Haque, Wright and Scielzo, 2013; Haque, Wright and Scielzo, 2014), in which they categorized the password-protected accounts of a user into two parts: higher-level and lower-level. While the higher-level passwords are used to protect financial and email/social media accounts, the lower-level passwords are mostly used in the contexts where

the users have little incentives to protect the security of the accounts such as the news/weather websites and lesser known websites (Haque, Wright and Scielzo, 2013; Haque, Wright and Scielzo, 2014). To further examine how an attacker can exploit these lower-level passwords to compromise the higher-level accounts, they conducted a comprehensive user study to investigate the similarities between these two types of passwords and the results demonstrate that the degree of similarity is quite high as users often reuse the lower-level passwords, with little or no modifications, to create their higher-level passwords.

One major limitation of the study by Haque, Wright and Scielzo (2014) is that they relied on a lab study to understand the relationship between the passwords of different categories of a user. The participants were asked to construct new, artificial passwords (different from their actual passwords) for both the higher and lower-level websites in a lab setting and answer some survey questions regarding their password reuse habit. Consequently, the ecological validity (Fahl et al., 2013) was low in their study. Furthermore, when cracking the higher-level passwords by using the lower-level ones, they only considered the syntactic similarities between these two groups.

Our current work is built upon this proposed hierarchy, and we measure the degree of similarity between the lower and higher-level passwords by collecting actual passwords from the leaked databases on the internet. Furthermore, in addition to syntactic similarity, we focus on dictionary attacks, service-related keywords, and semantic similarity. Our results show that by combining syntactic similarity, dictionary attack, service-related keywords, and semantic similarity, 82.3% of the higher-level passwords were cracked. When only a dictionary attack was used on the same set of linked passwords, 30.1% of the higher-level passwords were cracked whereas the syntactic similarity method alone could crack 73.6% of the passwords. The additional percentage of passwords cracked by using service-related keywords and semantic similarity was nominal.

Our work joins the growing body of literature in cybersecurity that addresses the usability-security trade-off (Di Nocera and Tempestinim, 2022; Di Nocera, Tempestini and Orsini, 2023; Adams and Sasse, 1999; Whitten and Tygar, 1999). We show how a malicious hacker can exploit the usability issues related to managing multiple password-protected accounts to gain access to the more important accounts of a user through publicly available information and moderate computing resources. Our study calls for a deeper understanding of the behavioral approaches of the users to protect their high-value accounts on the e-commerce and social media websites.

Our main contributions include: 1) We leverage an established user password hierarchy to determine the degree of similarity between the higher and lower-level passwords by using actual leaked passwords on the internet. 2) We examine the effectiveness of syntactic similarity, semantic similarity, dictionary attacks, and service-related keywords. 3) We connect our findings to the usability/security trade-off in cybersecurity in regard to managing multiple password-protected accounts.

METHODOLOGY

Similar to the work of Haque, Wright and Scielzo (2014), the major goal of this study is to obtain an understanding of the risks of users reusing similar passwords between lower and higher importance websites. However, the passwords obtained for this study is from the leaked data breaches and not from surveyed participants. We aim to simulate the hypothetical scenario in which a hacker has a lower-level password and attempts to use it to guess the higher-level ones. In this regard, the attacker would use dictionary attack, syntactic similarity, semantic similarity, and service-related keywords. This, in turn, would help us to better understand if there are increased risks for the internet users who use similar passwords for their lower and higher-level websites. In summary, there are two novel characteristics in our developed approach. First, we examine the effectiveness of combining multiple approaches: dictionary attack, syntactic similarity, semantic similarity, and service-related keywords. Second, we test our approach on real-world passwords that are publicly available due to several data breaches, which removes any discrepancies between passwords created during artificial lab studies (Haque, Wright and Scielzo, 2014) and passwords constructed by the general population for their actual accounts.

Dataset

The dataset used for this study must contain linked passwords, which we define as two or more passwords from the same internet user across multiple websites. While password databases are available such as the RockYou dataset or the AuthInfo dataset, such datasets either have no userIDs or contain userIDs that are sufficiently anonymized (Güven, Boyaci and Aydin, 2022). While this was done to protect internet users and allow researchers to expand on password practice research, these datasets are not effective for simulating this hypothetical situation as it is not possible to know whether two or more passwords belong to the same internet user. As a result, we downloaded several email+password combination lists from intelx.io. Operating under the ground truth that email addresses are unique to an internet user, we recorded the instances where there were duplicate email addresses across data breaches, suggesting that the associated passwords would belong to the same user.

Data Collection and Procedural Analysis

We downloaded numerous sections of data breaches from intelx.io (as their database splits data leaks into smaller files), performed a check to ensure that the email address provided is in a valid format, and proceeded to search for duplicate email addresses among all the files. In order to search quickly among the growing list of user credentials, we adopted a dictionary approach that involved creating a Python dictionary (unrelated to the dictionary attack). Using the dictionary data structure allowed us to quickly acquire the values for each key in the dictionary. This approach begins by looping through a list of user entries including an email address, password, and the source website. As the list is being looped through, every

email address would be checked to find out if it has been seen in previous iterations. If not, the email address would be marked as seen, while storing the passwords/sources, and the iterations would continue. If the email address has been seen in previous iterations, then the password and the source of the current entry iteration would be stored with the original email address entry. The entries were later organized into financial, identity, and sketchy/content sites based on the purpose of the site (Haque, Wright and Scielzo, 2014). For example, bitcoin and other cryptocurrency accounts are typically used for financial transactions. Websites like FlashFlashRevolution, GamerzPlanet, and Gamigo have the intended purpose of being used for playing games, but may also appear reasonably sketchy to the individuals using the sites. MySpace was the only site used in the identity category as it was the only available social media platform with both plaintext passwords and corresponding email addresses. Table 1 shows the list of the websites along with their categorization. On a set of about 13.1 million email+password entries from various sites, this method was able to find duplicate email addresses in a total of 75.4 seconds. This time consisted of 21.1 seconds for storing the combined 13.1 million entries in the RAM for processing, 47.7 seconds for classifying entries into linked passwords, and 6.6 seconds to write the linked passwords to a file. This strategy demonstrates that it is a relatively simple task for malicious hackers to find the linked passwords from publicly available account data breaches.

Table 1. Website classifications.

Sketchy/Content	Identity	Financial
Aipai.com	MySpace	Bitcoin Accounts
FlashFlashRevolution.com		Generic Cryptocurrency Accounts
FrostLand.pro		
Game-Tuts.com		
GamerzPlanet.net		
Gamevn.com		
Gamigo.com		

Password Cracking Methods

Now we describe each step of our password cracking method.

Tokenizing Given Passwords

At first, we used a Python script to tokenize the lower-level password. The given password was broken down into individual parts such as words, numbers, and special characters for syntactic/semantic similarity exploitation.

Recognizing Words in Passwords and Syntactical Similarity Exploitation

In order to use syntactic similarity, the pre-built WordNinja module was used, which can take a string (the given password) and produce a list

of all of the words within the string. The remaining entities within the password (extra letters, numbers, digits) would also be appended to the end of the list of words. The WordNinja module uses NLP (Natural Language Processing) techniques to extract the words based on what the most likely words are within the string. The words can then be rearranged, removed, and added. Additionally, extra characters such as digits or special characters could be added, the password could try to be reversed, and different letters could be capitalized or replaced with similar characters (for example, password → p@ssw0rd).

Semantic Similarity

In order to leverage semantic similarity, we used a word2vec model trained on GoogleNews-vectors-negative300.bin.gz. A word2vec model uses NLP techniques to recognize the relationships between a large set of words from the training set with regards to how similar they are. We extracted ten of the most similar words for each lower-level password. The Google training set does a fairly good job in returning the most similar words while simultaneously not being encumbered with data and thereby compromising efficiency. By generating the words that are similar to the ones seen in the password, our password cracker can recombine these new words with each other, words from the previous passwords, and even return to strategies in the syntactic similarities as mentioned above.

Service-Related Keywords

Service-related keywords were created semi automatically using prompts on ChatGPT. Such prompts included "please generate a Python list of 50 keywords related to MySpace" and "please ensure that each keyword is a single word". These keywords could then also be combined/fragmented with words in a user's given password. Some keyword examples are: MySpacekeywords = ["MySpace", "myspace", "Myspace", "music"], GamesKeywords = ["games", "minecraft", "dantdm", "creeper"].

Dictionary Attack

For the dictionary attack, the password cracker used about 14 million unique passwords from the famous RockYou leaked dataset from December, 2009. These commonly used passwords were tried as potential guesses to match the users' password. Using a dictionary of leaked passwords differs from the list of words and phrases used in the dictionary attack of Haque, Wright and Scielzo (2014). The dictionary attack, syntactic/semantic similarities, and service-related keywords generate guesses that vary depending on the lower-level password and the website of the higher-level password.

RESULTS

Password Breakdown

A total of 13,174,417 email and passwords combinations were downloaded from publicly available account data breach sources. These sources

include generic video game sites (sketchy/content), MySpace (identity), and bitcoin/generic cryptocurrency sites (financial). We acknowledge that cybercriminals post false password data leaks to harm companies' reputations or make password leak checkers less efficient, so it is important to perform some data cleansing before any testing takes place (Maschler, Niephaus and Risch, 2017). In this study, the downloaded entries with invalid email structures were removed as they could easily be proven as fake accounts. The remaining downloaded entries that are composed of only ASCII characters made up the clean entries. The number of entries downloaded versus the remaining clean entries used can be seen in Table 2. A total of 13,093,960 entries remained after the invalid entries were removed as shown by Table 2. More information regarding the breakdown of password lengths and composition is included in Appendix as this information can allow this study to be more easily compared to other research on password cracking but is not explicitly relevant to the goal of the study.

Table 2. Downloaded vs clean entries.

Sources	Entries Downloaded	Clean Entries
Sketchy/Content	6,227,357	6,211,512
Identity/Financial	6,947,060	6,882,448

Password Cracking Findings

Out of the 13,093,960 clean entries, there were 62,413 linked passwords. Each instance of linked passwords was derived from a unique user as suggested through a unique email address. Given one lower-level password per higher-level password for a unique user, 51,349 passwords out of the 62,413 higher-level passwords were cracked using the four aforementioned methods (dictionary attack, syntactic/semantic similarities, and service-related keywords), giving a password cracking rate of 82.3%. The average time spent attempting to crack a password was 1.82 seconds using an Intel(R) Core(TM) i5-1035G1 CPU, 16 GB RAM, an integrated graphics card, and Windows 11, 64-bit operating system.

Additionally, we ran each of the four methods separately on the same set of 62,413 hidden passwords to understand their individual impact. Overall, 18,755 passwords were cracked using only the dictionary method, giving a password cracking rate of 30.1% at an average of 0.19 seconds per password. The syntactic method was able to crack 45,938 passwords, giving an accuracy of 73.6% at 0.82 seconds spent per password attempted. The semantic method cracked only 27 passwords, giving an accuracy of 0.04% at 0.31 seconds per password. Finally, the service-related keywords method cracked 3008 passwords, giving an accuracy of 4.8% at 0.63 seconds per password. The differences in accuracy between the methods have been demonstrated in Figure 1.

From the above results, we attempt to re-answer the following question from Haque, Wright and Scielzo (2014) using real-world leaked passwords as opposed to surveyed participants (Haque, Wright and Scielzo, 2014).

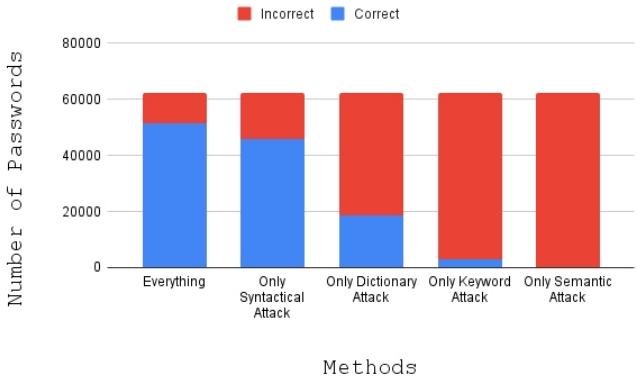

Figure 1: Password cracking comparison.

Research Question: "What percentage of higher-level passwords could be cracked by an attacker by compromising one lower-level password of each participant?" Given a single lower-level password, 82.3% of the higher-level passwords were cracked compared to 26.8% in (Haque, Wright and Scielzo, 2014). It should be noted that there are passwords that could have been cracked by any of the several methods. When visually examining the passwords, it was clear that many linked passwords appeared exactly identical in spelling. When the passwords were not exactly the same, one password often had the first letter capitalized or a single digit was appended to the password. As a result, the password cracker could effectively exploit syntactic similarities between the passwords. However, this study's unique approach of using WordNinja to extract the words in the passwords and incorporating a word2vec model did not prove effective.

DISCUSSION

We highlight the limitations of our work before discussing the potential implications.

Limitations

We followed the standard research procedures, which would simply be ignored by a malicious hacker. The malicious hackers would be able to take advantage of several resources that are restricted due to ethical reasoning such as accessing the dark web. The range and variety of data breaches on the dark web are perhaps significantly greater than those used in this study.

Implications

Despite these limitations, we demonstrate that by using moderate computational resources and publicly available information, it is possible to crack a very high percentage of higher-level passwords of the users. It has been suggested that the hackers only need to be able to crack roughly 1% to

2% of the passwords to get a return on investment when it comes to accessing user accounts (Acin, 2019), which this study surpasses greatly.

The dictionary attack and syntactic similarities methods proved highly effective at cracking a considerable percentage of passwords. Implementing only a simple dictionary attack that took passwords from a list of 14 million passwords from the RockYou data breach, 30.1% accuracy was obtained. Additionally, manipulating passwords syntactically allowed for an impressive 73.6% accuracy. The service-related keywords also proved effective at 4.8% accuracy, which is above the threshold of the 1% to 2% of accounts needed for a return on investment. The semantic similarity approach proved to be ineffective, however, and its inclusion ultimately led to wasted computational time and effort.

In comparison to the prior works on password cracking, we achieved better results in most of the cases. Simulations have shown that an attack can compromise more than 16% of user accounts in less than a thousand guesses given one of the passwords is known to the attacker (Pal, Daniel, Chatterjee and Ristenpart, 2019). Meanwhile, TarGuess-III and IV -- frameworks designed to leverage linked passwords and personally identifiable information (PII) gain success rates over 73% against normal users and over 32% against security-savvy users (Wang, Zhang, Wang, Yan, and Huang, 2016). A similar implementation of TarGuess showed that 78% of the passwords in a list of surveyed passwords could be cracked, with 66% of the passwords cracked in less than 20 guesses (Houshmand and Aggarwal, 2017). Passtrans was able to show that 67.5% of accounts could be guessed in under 1,000 attempts (He, Cheng, Xie, Wang and Liang, 2022). Our password cracking rate of 82.3% surpasses all these values and demonstrates that combining multiple approaches could give attackers a higher success rate.

More importantly, our work brings to the fore the issue of usability/security tradeoff in password management (Di Nocera, Tempestini and Orsini, 2023). Prior works have demonstrated through user surveys and lab experiments that convenience and simplicity trump what users know as secure behaviors when it comes to managing multiple password protected accounts (Haque, Wright and Scielzo, 2013; Haque, Wright and Scielzo, 2014). Our findings further validate this notion by leveraging the actual passwords used by the users on different websites. This calls for a deeper understanding of the behavioral approach to address the tension between usability and security (Di Nocera, Tempestini and Orsini, 2022).

We propose two solutions in this regard. First, as mentioned earlier, a typical user has dozens of password-protected accounts, but they can effectively remember only a few passwords. Hence, assuming that password reuse would be necessary, we recommend that users should reserve strong passwords for their high-value accounts and not reuse those for less important accounts. Furthermore, we recommend reserving a single, easy password for all the less important accounts on the content and sketchy websites. Second, we advocate the implementation of "password tax" (Bonneau and Preibusch, 2010), which would impose a cost on websites for every password-protected account they store. This would prevent sketchy

and content sites from making users register password-protected accounts, and in turn, would reward them instead for using a delegated protocol such as OpenID. As a result, the users would be able to use their limited cognitive capacity to memorize the strong passwords that are only used for their high-value accounts. This reward-based mechanism has also been advocated by the behavioral science researchers to address the usability/security trade-off in cybersecurity (Di Nocera, Tempestini and Orsini, 2022).

CONCLUSION

In this study, we leverage a well-established user password hierarchy to show the similarity between the passwords of high and low importance to the users. As opposed to relying on survey data from the participants, we use the leaked password datasets from the internet to exploit the lower-level passwords of a particular user to crack the higher-level passwords of the same user. Our findings show that password reuse is a major issue that needs to be addressed to mitigate the usability/security trade-off in cybersecurity. We offer a few suggestions that would enable the users to protect their high-value accounts and demonstrate that password reuse is a nuanced issue that should be addressed more carefully in the future research works in this area.

APPENDIX

Average Password Lengths and Compositions

Figure A1 shows the average length of passwords from various sources. While it was expected that financial sites would have the largest length of passwords (8.53 characters) and that identity sites would be smaller in length (7.61 characters), the length of passwords from the collection of games in the sketchy/content category was longer than that of identity sites and closer to the length of financial sites (8.37 characters). Figure A2 shows the percentage of various characters present within the passwords from their respective sources. When comparing the password composition to that of passwords in Haque, Wright and Scielzo (2014), they appear similar apart from the occasional outperformance of the sketchy/content passwords over identity passwords (the use of digits and capital letters).

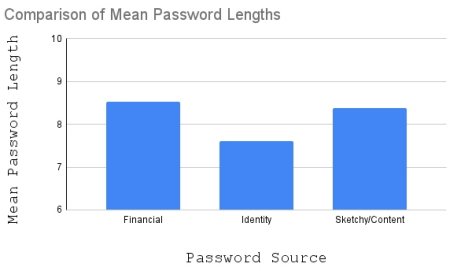

Figure A1: Average password lengths.

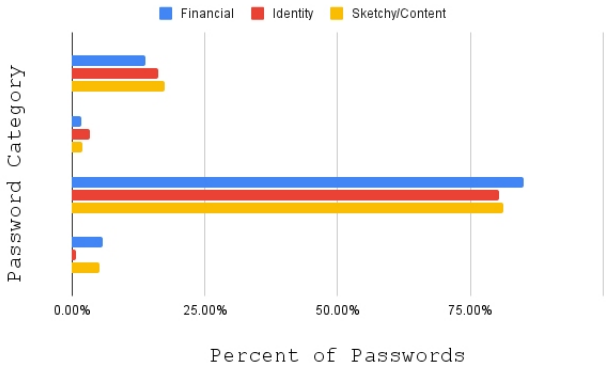

Figure A2: Percentage of characters.

REFERENCES

Acin, V. (2019) Making sense of the dark web. Computer Fraud & Security 2019(7), pp. 17–19.

Adams, A., Sasse, M. A. (1999) Users are not the enemy. Communications of the ACM 42(12), pp. 40–46.

Bonneau, J., Preibusch, S. (2010) The password thicket: Technical and market failures in human authentication on the web. In: WEIS.

Di Nocera, F., Tempestini, G. (2022) Getting rid of the usability/security trade-off: A behavioral approach. Journal of Cybersecurity and Privacy 2(2), pp. 245–256.

Di Nocera, F., Tempestini, G., Orsini, M. (2023) Usable security: A systematic literature review. Information 14(12), 641.

Fahl, S. et al. (2013) On the ecological validity of a password study. In: Proceedings of the Ninth Symposium on Usable Privacy and Security, pp. 1–13.

Güven, E. Y., Boyaci, A., Aydin, M. A. (2022) A novel password policy focusing on altering user password selection habits: A statistical analysis on breached data. Computers & Security 113, 102560.

Haque, S. T., Wright, M., Scielzo, S. (2013) A study of user password strategy for multiple accounts. In: Proceedings of the Third ACM Conference on Data and Application Security and Privacy, pp. 173–176.

Haque, S. T., Wright, M., Scielzo, S. (2014) Hierarchy of users' web passwords: Perceptions, practices and susceptibilities. International Journal of Human-Computer Studies 72(12), pp. 860–874.

He, X., Cheng, H., Xie, J., Wang, P., Yan, J., Huang, X. (2022) Passtrans: An improved password reuse model based on transformer, pp. 3044–3048.

Houshmand, S., Aggarwal, S. (2017) Using personal information in targeted grammar-based probabilistic password attacks. In: Advances in Digital Forensics XIII, pp. 285–303.

Maschler, F., Niephaus, F., Risch, J. (2017) Real or fake? large-scale validation of identity leaks. INFORMATIK 2017.

Miller, G. A. (1956) The magical number seven, plus or minus two: Some limits on our capacity for processing information. Psychological review 63(2), 81.

Pal, B., Daniel, T., Chatterjee, R., Ristenpart, T. (2019) Beyond credential stuffing: Password similarity models using neural networks. In: IEEE Symposium on Security and Privacy, pp. 417–434.

Sicora, M. (2022) How to find saved passwords on Mac. NordPass Website: https://nordpass.com/blog/how-to-find-passwords-on-mac/.

Wang, D., Zhang, Z., Wang, P., Yan, J., Huang, X. (2016) Targeted online password guessing: An underestimated threat. In: Proceedings of the ACM SIGSAC Conference on Computer and Communications Security, pp. 1242–1254.

Whitten, A., Tygar, J. D. (1999) Why johnny can't encrypt: A usability evaluation of pgp 5.0. In: USENIX Security Symposium, vol. 348, pp. 169–184.