KJMS 2022 VOL.4 (2): 113-125

KDU Journal of Multidisciplinary Studies (KJMS)
Volume 4. Issue 2 Nov 2022
DOI: http://doi.org/10.4038/kjms.v4i2.56

STRENGTHENING PASSWORDS AGAINST PEEPING ATTACKS

Induwara Jayalath[1], Thilini Delpachithra[1], Hansika Muthunayake[1], Tharindu Wijethilake[1] and Chamath Keppetiyagama[1]

University of Colombo School of Computing, Sri Lanka[1]

ABSTRACT

Despite a multitude of vulnerabilities of textual passwords, they are more likely to remain widespread since no scheme has become able to come close to providing all desired benefits. Among those vulnerabilities, peeping attacks are recognized as a real threat but, yet remain much unexplored. Most often, applications and systems use textual passwords for authentication, without considering the threat of peeping attacks. Our study provides the first numerical evidence of strength reduction to represent the impact of the attack. We introduce a novel authentication scheme that is conceptually different but purely text-based, as an endeavour towards strengthening textual passwords against the impact of peeping attacks. An experimental approach was used to collect data, simulating a peeping attack. Researchers intended were to provide an idea to the community, at the level of which the strength of a password can be reduced. This vulnerability is something crucial, yet haven't focused enough. Having such an understanding is desirable, as it can provide an image on the impact that these attacks can have on strength of textual passwords.

KEYWORDS: *Password security, Entropy, Peeping attacks, Authentication*

Corresponding Author: Tharindu Wijethilake, Email: *tnb@ucsc.cmb.ac.lk*

1. INTRODUCTION

Digitization acts as one of the core pillars of humans in the modern world as it helps people in their day-to-day lives. People use applications and information systems that contain much sensitive information daily (Zaman et al., 2017). These applications use authentication methods to secure personal data from unauthorized access. Textual passwords are a popular authentication method (Shah et al., 2015). Even though textual passwords are vulnerable to many threats and have been discussed widely, there is no trend to eliminate text-based passwords any sooner (Boˇsnjak and Brumen, 2019). Password strength meters (PSMs) help users to create stronger passwords.

The biggest issue, though, is whether password security can be guaranteed by password strength checks as strength measurement methods now in use have numerous shortcomings.

Most commonly used meters do not provide any publicly available explanation of their threat models or the logic behind their strength assignment techniques (de Carn´e de Carnavalet and Mannan, 2014). Most interesting fact is these strength meters gives different values for the same password (de Carn´e de Carnavalet and Mannan, 2014).

In this paper, we will highlight that the most effective password given by the strength checker can also be vulnerable. If the user enters the password in a crowded or public environment, such as a train or a bus, an observer can grasp some information about the password. Based on this information, the observer can infer the password. This is the peeping attack. Anecdotal evidence reveals these attacks occur more often than we might think (Boˇsnjak and Brumen, 2019). There are some already proposed solutions for this security threat. However as mentioned by Bosnjak et al. these solutions still have problems in usability and deployability. Even though textual passwords have many security issues, they are still in use as the major authentication method of most applications since 1960, which makes it challenging to replace new technology (Yang et al., 2020).

The focus of this study is to propose a solution to the strength reduction of passwords due to peeping attacks. So, we will propose a novel approach to reduce the threat caused by the peeping attack on textual passwords. We will use Shannon entropy-based calculations to measure the strength of the passwords. With a series of experiments, we will calculate the strength reduction of the passwords due to the peeping attack. Then, we will introduce our solution to the text-based authentication mechanism and redo the experiments to measure the strength reduction of the passwords. By comparing the entropy-based values, we will evaluate the effectiveness of our solution. Finally, we will also check the usability aspects of the solution.

2. LITERATURE REVIEW

According to NIST Digital Identity Guidelines, digital identity is the unique representation of digital services and can be claimed with digital authentication (Fenton et al., 2017). Passwords are the simplest and most widely used authentication method and their reliability of it depends on the strength of the password (Ma et al., 2010).

Password strength is a measurement of the effectiveness of a password against guessing or other attacks such as brute force, dictionary attacks, etc. (Panda et al., 2020). The current threat models of existing strength meters only have measures to mitigate or decrease the risk associated with brute-forcing and dictionary attacks (de Carn´e de Carnavalet and Mannan, 2014).

Attacks on Textual Passwords

The study of Hsien Cheng Chou et al. has shown the attackers can use methods such as social engineering, phishing and shoulder surfing to collect information on passwords and they have proposed a method to create strong passwords against dictionary attacks by considering the keyboard patterns. Wang Yao et al. has introduced a method to enter passwords using eye movements accurately using the front camera of mobile phones (Wang et al., 2018).

But this is difficult to use on laptops or desktops. A study by Shukia et al. introduces a new type of side-channel attack on the smartphone Personal Identification Number (PIN) entry process, that relies on the Spatio-temporal hand dynamics (Shukia et al., 2014). They have collected a corpus of 200 PIN entry videos that capture a part of the user's hand while entering the PIN and the backside of the smartphone. They have used a Tracking Learning Detection framework to track the positions. Based on their observations, they have concluded that the attack can decode up to 94% of the PINs. They believe the attack can be extended to passwords as well.

A paper by Panda et al. shows using of acoustic signals to recover 4–6-digit PIN from the emanations generated from the keystrokes with a chance of 60% (Panda et al., 2020). Their attack model is specifically made for modern PINs such as Automated Teller Machine keypads. Based on the results they provide a defence mechanism as well.

The study of Chou et al. has identified special keyboard patterns as AP-pattern (Adjacent and Parallel key). They have proved it has increased the effectiveness of dictionary brute force attack when incorporating these identified patterns (Chou et al., 2012).

The GazeRevealer is a side-channel attack technique (mobile app) which records the victim's eye patterns when tapping the keys on the screen using front camera. By analyzing those videos, GazeRevealer infers the keystrokes. The study has showed their approach achieved 77.43% accuracy for a single key number and 83.33% accuracy for the entire 6-digit password.

Peeping Attacks

Among techniques for cracking or acquiring passwords, peeping attacks remain a real threat. As a solution for peeping attacks, most of the studies introduce graphical/ picture-based authentication methods (Renaud and De Angeli, 2009), (Takada, 2008), (Zaman Nizamani et al., 2017), (Hameed et al., 2017), (Boˇsnjak and Brumen, 2019). The research conducted by FoongHo et al., has proposed a method of concealing password information to prevent shoulder surfing attacks using graphical passwords (Ho et al., 2014).

Zaman at el. has introduced a new approach with two steps to protect the textual password from peeping attacks and increase ~~the~~ security. The first step is the registration phase which will collect the password, encrypt, and store. The second step is password transformation, which transforms the character to different characters. Each time ~~when~~ the user enters the password it will show a mapping of the characters (decimals numbers). Therefore, the user will type the decimal numbers instead of real characters which will mislead the peepers (Zaman et al., 2017). ~~In~~ The study conducted by Cain et al. has shown that the nonadjacent, diagonal knight moves have reduced a significant level of threat in Over Shoulder Attacks (OSA), especially in swipe passwords (Cain et al., 2016).

The study conducted by Leon Boˇsnjak et al. with 274 participants for on-site shoulder surfing experiments provides empirical evidence that graphical passwords are easier to observe (Boˇsnjak and Brumen, 2019). This was one of the closest studies from our literature that explores the field of Shoulder Surfing attacks. As the final output, they provide vulnerability metrics and most importantly they verify their method on four conceptually different authentication methods. Considering three similarity metrics, password characteristics, distance metrics, guessing order and thirteen factors under them the vulnerability metrics have been developed. Their experiment of simulating the observation attack consists of two types of attackers (active and passive) and four types of authentication methods.

3. METHODOLOGY

We adopted the experimental research design and in-person interviews to collect the data for the research. A laboratory setup was created to simulate the password-entering process. An experimental approach was considered since examining a real peeping attack is not possible. Researchers followed

design science approach with empirical analysis for the research.

Preliminaries of experiment design

1) *Entropy Calculation:* Many studies have used entropy as a measurement of password strength (Yang et al., 2020), (de Carn´e de Carnavalet and Mannan, 2014), (Golla and D¨urmuth, 2018). It shows that passwords with lower entropies are weak and higher entropies are strong (Ma et al., 2010). We use Shannon entropy to calculate entropy (Shannon, 1948) before the experiment. This is the equation for a single-character password and it needs to be multiplied by the number of characters in the password according to the password.

$$E = -\sum_{i=1}^{n} P(x=1).\log_2 P(x=1)$$

E = Entropy of the password

n = Number of characters in the character pool

P(X) = Probability of the character

2) Proxemic Interactions: People naturally correlate physical distance to social distance. (Laga et al., 2016), (Hans et al., 2015). Social space ~~was~~ turned out to be the most suitable area to perform the experiment without sensing that some stranger is watching us. Therefore, we consider social space for our experiment.

3) *Peripheral Vision:* This vision covers an area of more than approximately 200 degrees in diameter horizontally (Valero et al., 2018). For our experiment, we considered the area out of this far peripheral vision of the user to avoid the user from getting any sense of the attacker.

4) *Lighting Condition:* In our experiment setup, we simulated the same lighting condition equals to 500 lux as same as working environments which was measured by a mobile application called Light Meter in apple app store (SUMMERS, 1989).

5) Data Source: We chose "RockYou", a corpus of 14 million unique passwords which was available on Kaggle (Mutalik et al., 2021). This corpus contains passwords, leaked from multiple popular applications like 'Facebook', 'myspace' and 'Friendster' etc (Weir et al., 2010).

6) Participants: The experiment has two roles for the participants; attackers and victims. External participants were considered as attackers and one of the experimenters were considered a victim. We employed 30 participants voluntarily, within the age from 18 to 25, which is considered as the highest computer literate group in Sri Lanka (Department of Census and Statistics - Sri Lanka, 2020). These participants are familiar with using passwords and authentication in general. As the sampling method, we use the judgemental sampling method, in the non-probabilistic sampling category (Taherdoost, 2018).

Pre-processing of data

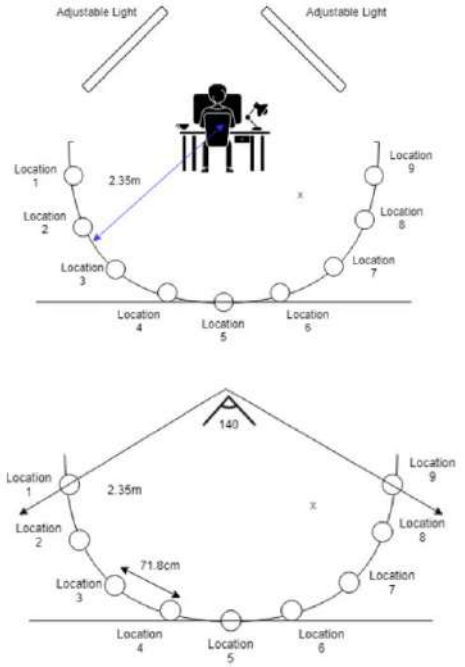
We filtered out set of passwords from the corpus considered as strong. The selection process was conducted in two phases.

1) Calculating the entropy: First, we selected passwords that have a length greater than eight characters which is considered as the minimum length of a strong password (Fenton et al., 2017). Then we calculated the entropy of those selected set of passwords using the aforementioned Shannon's equation assuming all possible keys are equally probable to be used in the passwords. The considered character pool includes 26 lowercase letters, 26 uppercase letters, 10 digits and 32 symbols.

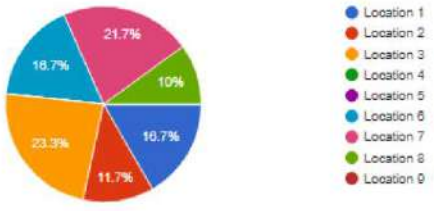
We selected the passwords with entropy greater than 128 since it is considered as the lower limit of the strongest category (Matematik and Winsløw, 2020). Considering the practical usage of passwords in our experiment, the highest entropy was considered as 150.755 and there were 18079 passwords which had this entropy. Then we cleaned the list by removing junk data such as email addresses, some HTML links and passwords which do not contain alphanumeric characters and symbols. Finally, it provided the output with 141 passwords which had the highest entropy.

2) Filtering through password strength meters (PSMs): Since the experimenters are encouraged to

deliberately choose passwords of sufficient length and complexity (Boˇsnjak and Brumen, 2020), we filtered the passwords using PSMs as well. According to the usage and recommendations of previous research, we used three popular third-party PSMs (Yang et al., 2020). We chose *All Things Secured*, *The Password Meter*, and *My1Login* (Julkunen and Molander, 2016). These meters consider different factors like the length of the password, LUDS, dictionaries, etc. to check the strength of the passwords. Based on the readings of all three PSMs separately, 110 passwords were labeled as strong. That set of passwords was again filtered using random sampling and selecting 30 passwords as the final set for the experiment.



(a) Location setup



(b) Location selection

Figure 1: Experiment setup (bird eye view)

Experimental Setup

We adopted an experimental process in order to simulate a peeping attack in an office environment. The laboratory conditions were set up mimicking a real office environment. QWERTY US laptop keyboard was used to input the password, which is considered as standard keyboard mapping. It which is often referred to as the universal keyboard (Buzing, 2003, Kafaee et al., 2022). First, we conducted the interviews using video recordings. Then it was executed as a live experiment as a comparison resource to evaluate the result with respect to the solution and then to check the cognitive impact of the human. A mobile phone camera with a resolution of 1080p was used to record the videos. The camera was placed in a position with a distance of 2.35m to the victim's position. 2.35m is the average distance considering the edges of social distance. Reflecting the average height of a Sri Lankan person, the stand which holds the camera was 5'3" of height (Bostock, 2019). The video recordings were taken from 9 locations around the victim with each having 2.35m of similar distance to the position of the victim. The locations were taken from the area that is not caught in the peripheral vision of the victim (140◦) (Valero et al., 2018). Every two positions have a similar distance in between (Figure 1a). Figure 2 shows the experimental setup which we used to record the videos. We used a laptop as the device for entering the password, with a screen size of 13 inches. 13 inches is considered the standard screen size.

Figure 2: Experiment setup

Experimental Procedure

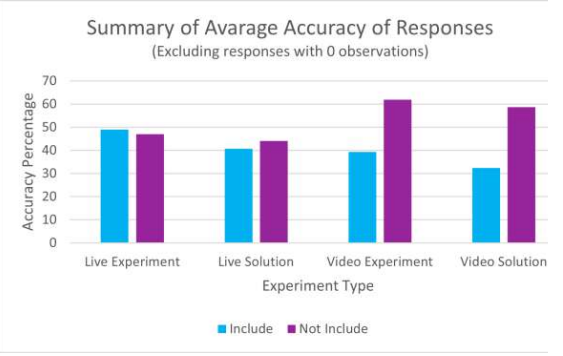
To simulate the attacker, we either can use a live human attacker or use online recruiters and ask them to observe video recordings of the victim entering the password. We performed our experiment in both ways. After preparing the experiment setup, we recorded the videos for the online experiment. One of the experimenters acted as the person who enters passwords (victim). She entered the passwords in a moderate speed imitating an average user. Videos for each password in the selected set were recorded from all nine positions. To avoid the person acting as the victim getting familiarized with the passwords, all passwords for a single location were recorded at one time and then moved to the next location. The videos were labeled according to the location and the password and then stored in Google drive. Once the video recording process was finished, we conducted a pilot study prior to the experiment with two participants. Based on the feedback from the pilot study some of the design decisions were changed.

1) Online experiment using video recordings

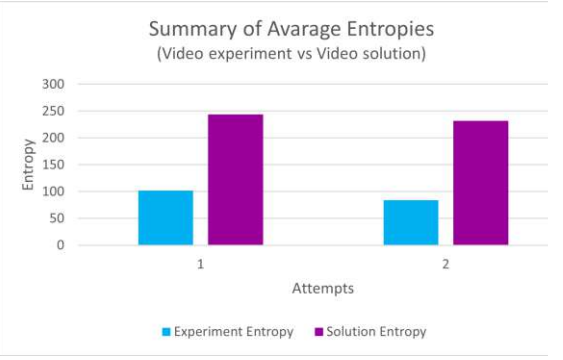
The interviews were conducted on the zoom platform. Only one participant at one time was taken into the virtual meeting room. First, the participant was provided with some instructions of the experiment steps using a PowerPoint presentation. It included an image of the 9 positions, the keyboard layout used to enter the passwords, and a video clip that reflects the 9 positions. Participants were given the freedom to choose a preferred position. They were informed regarding the intention of the experiment and asked to put themselves into the role of an attacker who tries to capture the password. We informed them to note down or memorize all the facts they can obtain about the passwords. After they selected the location, we provided the Google drive link of the video and asked them to share the screen and play the video on their computer. This was for smoother playback and to avoid the quality drop and getting stuck due to weak connection issues. After watching the video, an open-ended interview was conducted with the participant in order to avoid misunderstandings. We marked the keyboard areas according to their observations using an application.

Questions were based on following 4 categories.

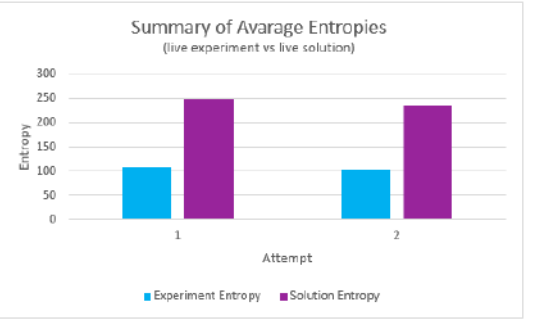
- Exactly included in the password
- Exactly not included in a password
- Approximately included in the password
- Approximately not included in the password



(a) Average guessing accuracy



(b) Average video observation entropy



(c) Average live observation entropy

Figure 3: Observation results

After getting the first feedback we gave them a second chance to watch the video and confirm their observations.

2) Onsite experiments

During the pilot study we noticed a huge difference between the quality of the visuals through the human eye and the camera lens. Therefore, in order to comparatively analyse live and video-based experiment responses, we conducted the same experiment on site.

The attackers were asked to observe the victims key-press events during the login process. Then a live interview was conducted to record what the attacker obtained. Same participants were taken to both online and live experiments, and we have strictly limited the collection and processing of personal data to the best of our abilities.

4 ANALYSIS

The first observation we got from the feedback of attackers was that location 5 was not taken and the majority had taken locations 3 & 7 (Figure 1b). Then we recorded their observations. If attackers that some characters are definitely or approximately not in the password, and if it is actually not in the password we omit them from the character pool, and if the attacker said some characters are definitely or approximately included in the password and actually that password includes these characters, we assume that all characters have the probability of 1. Other characters have an equal probability to reveal.

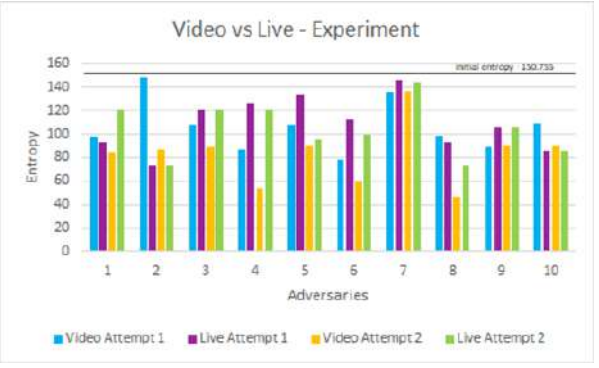


Figure 4: Video vs live entropy comparison summary (experiment)

Figure 3a shows the guessing accuracies of the adversaries. After this analysis, we identified, in the second attempt of video observation shows a progress on guessing the password (Figure 3b) and live

observations do not have considerable difference on that (Figure 3c). In our character pool, there are 94 characters. Each key in the keyboard owns two characters in the pool and if one key is revealed to the attacker, we have to assume that both characters in the pool are revealed. Based on these assumptions we calculate the entropy and compare them with the initial entropy values of the passwords. Figure 4 shows how the entropy values behaved in live and video-based experiments, with respect to the attempt.

Even though a drastic reduction of entropies were expected in live experiment after the login process was exposed to the attacker, it was identified most of the participants provided more accurate responses in the video experiment. Since we recruited 10 participants for the live experiment, the relevant 10 passwords from the video experiment were taken for the purpose of entropy comparison. As initially we assumed, a significant reduction of the entropies in the second attempt than in the first attempt can be observed reflecting they have provided more accurate responses in their second attempt. Compared to their initial entropies of original passwords, the entropies of both attempts have been reduced.

5. PROPOSED SOLUTION

Based on the results and observations obtained from the experiment, we designed a novel method of authentication. Most importantly, this form of authentication is not another graphical or totally different method from existing, widely used methods. Since we identified the usability and deployment issues in previously proposed solutions, we did not want to build a totally different method, replacing textual passwords which will require a lot of effort for a user to get used to. Hence, we built a solution based on textual passwords. This is a pure text-based solution and no additional tool or memorability is required. The actual changes for the

authentication process have been applied on the login screen. No changes have to be made in the password creation process; hence we can simply use the same password creation process when we implement this solution. Our primary intention here is to reduce the amount of strength reduction due to peeping attack. If we can reduce the number of correctly guessed characters, we can reduce the strength reduction. The usual elements we commonly see on a login form are username and password fields and submit button. The basic concept behind the proposed solution is the user has to insert the real password characters mingled with some random garbage characters. It will mislead the attacker making him think the garbage characters are also real characters in the password. The user is provided with the instructions by some indications in the input field such that he can see when to enter the real password characters and when to enter garbage characters. The input field borders and the label of the field will appear in green colour when the real password characters need to be entered. When the colour is changed into red and the label appears as Garbage, we need to enter some random characters.

(a) Real password characters (b) Garbage characters

Figure 5: Solution password entry field

The number of garbage characters to be entered at one time is randomly generated by the system. It is denoted as the count, and for the ease of the user, we indicated using a circle under the password field. When real characters are entered, the circle is getting filled, when garbage values are entered, it is getting empty. This number will be changed from one attempt to the next. For example, if the appeared count is 3, after entering 3 letters of the password the user has to enter 3 garbage values. For the first 3 characters, the field. will be in green and then will turn into red, after entering 3 garbage values again the field will turn into green colour.

At the same time, the progress circle will fill its perimeter for real characters by one-third of the circle for one character and will remove filled parts for garbage characters one by one. This will continue until the password ends. The interface of two scenarios when the real characters and garbage characters are entered is presented using the Figures 5a and 5b. We use colors like red and green as indications to make it easier to find relevant actions.

6. EVALUATION

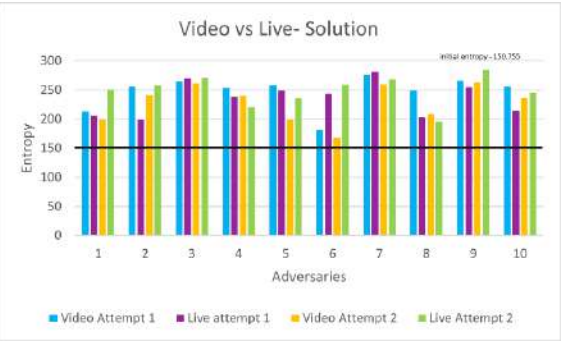


Figure 6: Video vs live entropy comparison summary (solution)

To evaluate the level of resistance to peeping attacks and the effectiveness of the proposed solution, we conducted the same experiment in a similar simulated environment, under the same conditions, as described in *Experimental Setup*. For the evaluation also, both live and video experiments were conducted. As the same participants took part, previous locations were taken as the attacker's position. We shuffled the passwords in order to avoid any attempts of participants trying to respond by recalling their responses from the last interview. If they noted down what they observed during the first experiment, they could use that as well. Then we allowed participants to watch the video and observe the characters in 2 attempts. We were cautious not to provide information about the nature of the new authentication method. They were just asked to observe the password-entering process and provide a response based on what they observed. The interview with the attacker was conducted in the same way as the previous interviews. After the experiment is done with the solution it reveals that the password entropy

is higher than the initial entropy after experiment. Figure 6 illustrates the entropy increase of the password when using our solution in all the attempts of the adversaries in both video and live attempts. Table 1 displays the entropy differences relevant to the attempts.

To evaluate our solution's usability aspect, we conducted a survey with the participation of same users of the experiment. We focused on the login times and the error rate with respect to the number of attempts. The error rate reflects the learnability of the scheme (Ashley A. Cain and Jeremiah D. Still, 2018). The participants were tasked to enter a given password using our proposed method. The password was chosen from our initial corpus. We measure the time starting with the participant clicking on the password field and ending with the participant clicking the login button. Participants were given practice trials as they were new to the scheme, they were asked to enter passwords in five attempts once they are ready.

Table 1: Entropy difference comparison

Description	Overall	Attempt 1	Attempt 2
Initial entropy	**150.755**	**150.755**	**150.755**
After video experiment	92.956	102.090	83.822
Entropy reduction with initial	57.799	48.665	66.933
After live experiment	106.335	103.847	108.822
Entropy reduction with initial	44.420	46.908	41.933
After video solution	237.572	243.593	231.551
Entropy increment with initial	86.817	92.838	80.796
Entropy increment with experiment	144.616	141.503	147.729
After live solution	242.372	235.895	248.849
Entropy increment with initial	91.617	85.140	98.094
Entropy increment with experiment	136.037	132.048	140.026

Usability Evaluation

We used 30 passwords and 30 participants and each one had 5 attempts on solution (Figure 7). The time taken for the normal textual passwords are lower which can be attributed to their familiarity with the scheme. It also showcases when attempts go on, the time taken becomes slightly lower. Figure 8 shows the percentages of wrong password inputs participants have submitted in each of their attempts. 'Normal attempt' of the graph shows the percentage of wrongly typed passwords. Other attempts show the percentage of wrong inputs when using our solution. Inputting garbage values does not take that much of time since user can press some random keys. After a set of garbage characters, user might have to recall the next character of the real password.

It was observed that the input speed becomes slightly faster and the percentage of incorrect inputs in each attempt moves towards 0 when repeating, showing that the wrong attempts can be reduced when users have sufficient experience. We made available a small browser extension for chromium - based

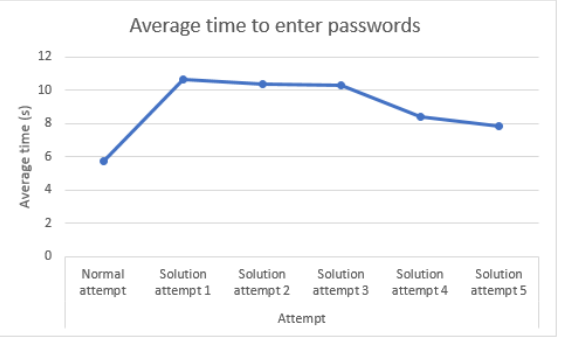


Figure 7: Time spent to enter passwords using the solution

browsers (Figure 9). When the user is in any login page, he or she could use the plugin to type the password. Once the password is typed, user can copy and paste it in the password field of the respective login form. Our solution, "PeepingOne" is designed addressing some limitations of previously proposed authentication with respect to the deployability, usability and security.

Hence, users can use it once it is added to the browser without experiencing compatibility issues, storage issues and privacy issues. This will just be added to the browser and does not require any device storage. The size of the extension itself is 109KB. We do not store any of the passwords typed by the user in PeepingOne. It does not have access to the browser cache, therefore any sensitive information the user might have stored on the browser will not be exposed to it. It can be introduced as a surface running application while it does not require any cookie access or local storage access on the browser. Unlike most other applications, PeepingOne does not ask permission to access any privacy related features such as microphone, video camera, location etc. Another issue which was highlighted is the processing speed of the scheme. Most of the

alternative schemes were using graphical approaches which requires a lot of processing power and time (Zhao and Li, 2007). Those methods also consume a lot of storage space since they use thousands of graphical elements (Shah et al., 2015). From the developer's perspective, the implementation of the software does not require much effort.

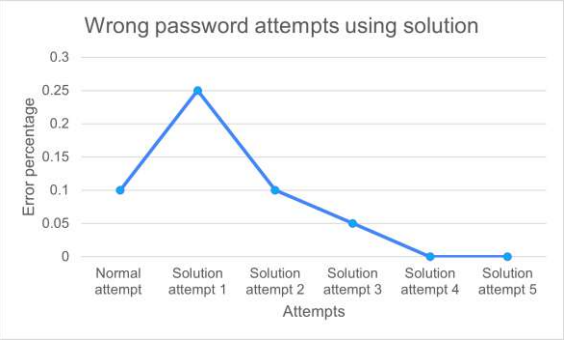


Figure 8: Percentage of wrong password input in each attempt

7. DISCUSSION

Throughout the study, our focus was on the strength reduction of textual passwords by shoulder surfing attacks. It was identified a significant amount of strength reduction in both live and video experiments. There were no significant differences observed between the results of the two types of experiment. Also, participants were more tended towards selecting location 3 and 7 (Figure 1b). This questions most of the design decisions of previous studies because; most of them chose a place directly behind the attacker. In our case, no one chose that particular place.

Another contribution is introducing a novel form of authentication which solves many drawbacks of previously proposed solutions. They addressed this threat of shoulder surfing by introducing a whole new method replacing textual passwords. Considering the easiness of learning, if it requires less time for users to get trained, it is a good sign that it provides a positive user experience. Graphical schemes usually need more complex interactions than simply entering a textual password. For example, the user might have to search for an image from a very large number of images multiple times. (Still and Cain, 2020). However, studies can be found which have recognized graphical schemes are more vulnerable to peeping attacks (Boˇsnjak and Brumen, 2019).

Having a complex registration phase is another issue. (Zhao and Li, 2007), (Renaud and De Angeli, 2009). Users have to create their passwords using the new scheme. Our solution does not require any specific

Figure 9: Browser extension

prior work before the login phase as the usual process of textual registration can be followed. The implementation and deployability were other major issues they had. As mentioned in their researches, significant limitations of those solutions are related to implementation efforts. Also, they need extra memory capacity to store their additional materials. Our solution is pure textual password solution that is easy to implement. Our solution is recommended for people who memorize their passwords since they can recall the real characters after a set of garbage characters. People most often tend to keep their passwords written down somewhere on their computers or physically in notepads, sticky notes etc. For those who copy and paste their passwords from somewhere, our solution would not be an ideal solution. Similarly, we do not recommend this for people who use auto filling password field since the garbage values need to be entered at some points in the middle of the actual password. However, unless you are using a trusted application from a trusted source to store passwords, copying, and pasting passwords you saved on the computers is not recommended. It has a different set of threats associated with it. When using this extension, it is

better not to use the same character repetitively in the password as garbage values as there is a potential security threat to identifying such characters by the attackers successfully.

8. CONCLUSION

It is clear that the password strength meters measure the strength of the password with different matrixes but do not consider the peeping attack. But it is a real threat to textual passwords. Based on the results of the study we state the answers to the following questions.

1) *To which extent password strength can be reduced due to peeping attacks?*

Our experimental setup was created unflavored for the attacker. But still the attackers have obtained some information on the password entered by the victim. Under these conditions, the result of the first analysis shows that there is a potential entropy reduction of 29.46% and 38.33% in both live and video attacks respectively.

2) *How to reduce the strength reduction of passwords after a peeping attack?*

We have introduced a novel form of authentication which does not replace textual passwords. Based on our evolution we have observed that it has an increase of 60.77% of entropy of the passwords in live attacks and 57.58% increment of entropy in video attacks. Most of the proposed solutions have problems with deployment and usability. But we have shown that our solution can easily deployed and user adaptation is very fast.

9. REFERENCES

Ashley A. Cain and Jeremiah D. Still, 2018 Ashley A. Cain and Jeremiah D. Still (2018). Usability comparison of over-the-shoulder attack resistant authentication schemes. *J of Usability Studies*, 13(4): pp. 196–219.

Balzarotti et al., Balzarotti, D., Cova, M., and Vigna, G. ClearShot: Eavesdropping on Keyboard Input from Video. Technical report.

Boˇsnjak and Brumen, 2019 Boˇsnjak, L. and Brumen, B. (2019). Shoulder surfing: From an experimental study to a comparative framework. *Int J of Human Computer Studies*, 130: pp. 1–20.

Boˇsnjak and Brumen, 2020 Boˇsnjak, L. and Brumen, B. (2020). Shoulder surfing experiments: A systematic literature review. *Computers and Security*, p. 99.

Bostock, 2019 Bostock, B. (2019). 25 Countries With the Shortest People in the World.

Buzing, 2003 Buzing, P. (2003). Comparing different keyboard layouts: aspects of qwerty, dvorak and alphabetical keyboards. *Delft University of Technology Articles*, (Jan 2003): pp. 1–11.

Cain et al., 2016 Cain, A. A., Chiu, L., Santiago, F., and Still, J. D. (2016). Swipe authentication: Exploring over-the-shoulder attack performance. *Advances in Intelligent Systems and Computing*, 501: pp. 327–336.

Chou et al., 2012 Chou, H. C., Lee, H. C., Hsueh, C. W., and Lai, F. P. (2012). Password cracking based on special keyboard patterns. Int J of Innovative Computing, Information and Control, 8(1 A): pp. 387–402.

de Carn´e de Carnavalet and Mannan, 2014 de Carn´e de Carnavalet, X. and Mannan, M. (2014). From Very Weak to Very Strong: Analyzing Password-Strength Meters.

Department of Census and Statistics - Sri Lanka, 2020 Department of Census and Statistics - Sri Lanka (2020). Computer Literacy Statistics – 2020 (First six months). Computer Literacy Statistics, 2017:pp. 1–4.

Fenton, J. L., Newton, E. M., Perlner, R. A., Regenscheid, A. R., Burr, W. E., Richer, J. P.,

Lefkovitz, N. B., Danker, J. M., Greene, K. K., Theofanos, M. F., Newton, E. M., and Burr, W. E. (2017). Digital Identity Guidelines: Authentication

and Lifecycle Management. Special Publication (NIST SP) - 800-63B.

Golla and D¨urmuth, 2018 Golla, M. and D¨urmuth, M. (2018). On the accuracy of password strength meters. *Proc of the ACM Conference on Computer and Communications Security*, pp. 1567–1582.

Hameed, S., Qaizar, L., and Khatri, S. (2017). Efficacy of Object-Based Passwords for User Authentication SDN Based IoT Security View

project Securing SDN View project Efficacy of Object-Based Passwords for User Authentication. Technical report.

Hans, A. and Hans, E. (2015). Kinesics, Haptics and Proxemics: Aspects of Non -Verbal Communication. IOSR *J Of Humanities And Social Science*. Ver. IV, 20(2): pp. 47–52.

Ho, P. F., Kam, Y. H. S., Wee, M. C., Chong, Y. N., and Por, L. Y. (2014). Preventing shoulder-surfing attack with the concept of concealing the password objects' information. *Scientific World J*, 2014.

Jebriel and Poet, 2011 Jebriel, S. M. and Poet, R. (2011). Preventing shoulder-surfing when selecting pass-images in challenge set. 2011 *Int Conf on Innovations in Information Technology*, IIT 2011, pp. 437–442.

Julkunen and Molander, 2016 Julkunen, H. and Molander, J. C. (2016). Password Strength and Memorability.

Kelley et al., 2012 Kelley, P. G., Komanduri, S., Mazurek, M. L., Shay, R., Vidas, T., Bauer, L., Christin, N., Cranor, L. F., and L´opez, J. (2012). Guess again (and again and again): Measuring password strength by simulating password-cracking algorithms. *Proc IEEE Symposium on Security and Privacy*, pages 523–537.

Laga, Laga, H. Personal space-based modeling of relationships between people for new human-computer interaction. (Sept 2016).

Ma et al., 2010 Ma, W., Campbell, J., Tran, D., and Kleeman, D. (2010). Password entropy and password quality. Proceedings - 2010 4th International Conference on Network and System Security, NSS 2010, pp. 583–587.

Matematik and Winsløw, 2020 Matematik, K. and Winsløw, V. C. (2020). Study and Research Paths in Discrete Mathematics Two Designs for Upper Secondary School. (92).

Mutalik et al., 2021 Mutalik, R., Chheda, D., Shaikh, Z., and Toradmalle, D. (2021). Rockyou.

Panda, S., Liu, Y., and Hancke, G. P. (2020). Behavioral Acoustic Emanations: Attack and Verification of PIN Entry Using Keypress Sounds. pp. 1–25.

Renaud and De Angeli, 2009 Renaud, K. and De Angeli, A. (2009). Visual passwords: Cure-all or snake-oil? *Communications of the ACM*, 52(12): pp. 135–140.

Seneviratne, P., Perera, D., Samarasekara, H., Keppitiyagama, C., Thilakarathna, K., De Soyza, K., and Wijesekara, P. (2020). Impact of Video Surveillance Systems on ATM PIN Security. *20th Int Conf on Advances in ICT for Emerging Regions*, ICTer 2020 - Proc, (Feb): pp. 59–64.

Shah et al., 2015 Shah, A., Ved, P., Deora, A., Jaiswal, A., and D'Silva, M. (2015). Shoulder-surfing resistant graphical password system. Procedia Computer Science, 45(C): pp. 477–484.

Shannon, 1948 Shannon, C. E. (1948). A Mathematical Theory of Communication. *Bell System Technical J*, 27(4):623–656.

Shukia et al., 2014 Shukia, D., Kumar, R., Phoha, V. V., and Serwadda, A. (2014). Beware, your hands reveal your secrets! *In Proc of the ACM Conf on*

Computer and Communications Security, pp. 904–917. Association for Computing Machinery.

Still and Cain, 2020 Still, J. D. and Cain, A. A. (2020). Over-the-shoulder attack resistant graphical authentication schemes impact on working memory. Technical report.

SUMMERS, 1989 SUMMERS, A. J. (1989). Lighting and the Office Environment: A Review. *Australian J of Physiotherapy*, 35(1):15–24.

Taherdoost, 2018 Taherdoost, H. (2018). Sampling Methods in Research Methodology; How to Choose a Sampling Technique for Research. *SSRN Electronic Journal*, 5(2): pp. 18–27.

Takada, 2008 Takada, T. (2008). FakePointer: An authentication scheme for improving security against peeping attacks using video cameras. *In Proc The 2nd Int Conf on Mobile Ubiquitous Computing, Systems, Services and Technologies*, UBICOMM 2008, pp. 395–400.

Ur et al., 2012 Ur, B., Kelley, P. G., Komanduri, S., Lee, J., Maass, M., Mazurek, M. L., Passaro, T., Shay, R., Vidas, T., Bauer, L., Christin, N., and Cranor, L. F. (2012). How does your password measure up? The effect of strength meters on password creation. Proceedings of the 21st USENIX Security Symposium, pp. 65–80.

Valero, F., Mittal, A., Tollmar, K., Lungaro, P., Sj, R., and Jos, A. (2018). Gaze-Aware Streaming Solutions for the Next Generation of Mobile VR Gaze-aware streaming solutions for the next generation of mobile VR experiences. Sept.

Wang, Y., Cai, W., Gu, T., Shao, W., Khalil, I., and Xu, X. (2018). GazeRevealer: Inferring password using smartphone front camera. ACM International Conference Proceeding Series, (September): pp. 254–263.

Weir et al., 2010 Weir, M., Aggarwal, S., Collins, M., and Stern, H. (2010). Testing metrics for password creation policies by attacking large sets of revealed passwords. Proceedings of the ACM Conference on Computer and Communications Security, pp. 162–175.

Yang, Y., Yeo, K. C., Azam, S., Karim, A., Ahammad, R., and Mahmud, R. (2020). Empirical study of password strength meter design. *Proc of the 5th International Conference on Communication and Electronics Systems*, ICCES 2020, pp. 436–442.

Zaman, S., Raheel, S., Jamil, T., and Zalisham, M. (2017). A Text based Authentication Scheme for Improving Security of Textual Passwords. *Int J of Advanced Computer Science and Applications*, 8(7).

Zaman Nizamani et al., 2017 Zaman Nizamani, S., Jamil Khanzada, T., Raheel Hassan, S., and Zalisham Jali, M. (2017). A Text based Authentication Scheme for Improving Security of Textual Passwords. Technical Report 7.

Zhao and Li, 2007 Zhao, H. and Li, X. (2007). S3PAS:A Scalable shoulder-surfing resistant textual-graphical password authentication scheme. *In Proc 21st Int Conf on Advanced Information Networking and Applications Workshops/Symposia*, AINAW'07, vol 1, pp. 467–472

Kafaee, M., Daviran, E. and Taqavi, M. (2022) "The QWERTY keyboard from the perspective of the collingridge dilemma: Lessons for co-construction of Human-Technology," *AI & SOCIETY* [Preprint]. Available at: https://doi.org/10.1007/s00146-022-01573-1.