PASDJO: Quantifying Password Strength Perceptions with an Online Game

Tobias Seitz
LMU Munich
Germany
tobias.seitz@ifi.lmu.de

Heinrich Hussmann
LMU Munich
Germany
hussmann@ifi.lmu.de

ABSTRACT

Users often fail to create strong passwords. Besides lack of motivation, another possible explanation are misconceptions about the factors that contribute to password strength. Such misconceptions play an important role for the design of feedback systems during password selection. In this paper, we present an online game that helps quantifying the perception of password strength. Players score points by rating the strength of passwords accurately under time pressure. We analyzed the usage logs from the first four months after rollout. We found that users underestimate passphrases by 1.4 points on a 5-point strength scale, while their other ratings are fairly consistent with our estimates. Although we used a different methodology, we were able to corroborate related findings and narrow down the features that users think contribute to password strength. We highlight how the data collected through PASDJO can help designing better password feedback and boost user experience during account creation.

CCS CONCEPTS

• **Security and privacy → Usability in security and privacy;**

KEYWORDS

usable security; authentication; passwords; psychology; game;

ACM Reference format:

T. Seitz, and H. Hussmann. 2017. PASDJO: Quantifying Password Strength Perceptions with an Online Game. In ***Proceedings of the 29th Australian Conference on Human-Computer Interaction, Brisbane, QLD, Australia, November 2017 (OzCHI 2017)***, 9 pages.
https://doi.org/10.1145/3152771.3152784 [1]

1 INTRODUCTION

The most prevalent method to authenticate users for numerous types of systems is still a combination of a username and a password [2]. The process is easy to implement and requires little learning because most users have become accustomed to it. However, in many cases users do not select strong passwords, despite the available tools and countermeasures taken by service providers [12]. Among other usability issues [2,10,12], it is often argued that users sometimes have a suboptimal perception of the factors that add to password strength [35,36,38]. Encountering inconsistent and misleading password creation rules and policies contributes to faulty password strength perceptions [7,13,29,40].

Moreover, many services utilize password strength meters [37]. These visual aids are shown as growing colored bars close to the password input and provide instant feedback on the estimated strength of the selected password. Instead of forcing the users to choose a stronger password, password meters use a softer approach. One of their goals is to persuade users to rethink their choice if the password is rated poorly. However, just like policies, the ratings are largely inconsistent across services: A password like "password$1" can be considered "very strong" by one website, while it may receive a much lower rating by another [5]. In any case, the password would be accepted which leads to a positive reinforcement that it is okay to use the password despite security concerns. This can also confuse users and reduce the overall credibility and understandability of password meters.

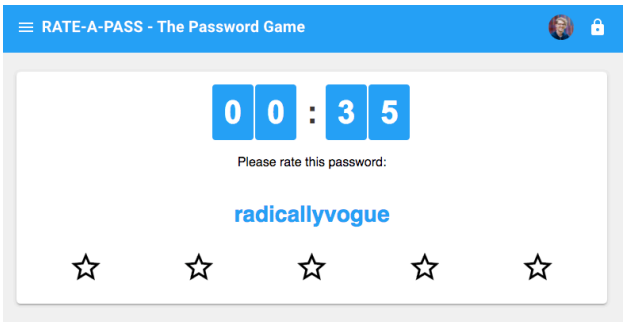

Fig. 1. Screenshot during game-play. The player guesses how strong the password is by rating it with one to five stars. A countdown incentivizes to act quickly and intuitively. "radicallyvogue" is a passphrase and was rated with four stars by the game, while the median of users' perception was two stars for password category.

Misleading policies and password meters can thus lead to suboptimal perceptions of the factors that add to password

OzCHI'17, November 2017, Brisbane, QLD, Australia

ACM 978-1-4503-5379-3/17/11 $15.00 https://doi.org/10.1145/3152771.3152784

strength. This is an important problem because an understanding of password strength is a necessary prerequisite to create appropriate passwords. In order to design more useful and reliable systems to help users with password selection, we need to find out which kind of feedback is necessary, e.g. for password meters and what can be omitted. To achieve this goal, we aim to identify factors which are erroneously considered beneficial for password strength and which are underestimated. As a consequence, we would be able to provide more understandable and trustworthy feedback. Ideally, users are then able to choose passwords of appropriate strength that they have a fair chance to remember and type in easily. Studies to this day focused on the perception of common passwords and alterations of them [36], while the perception of other types of passwords is still less understood.

Our work provides further insights into password strength perceptions. Besides common passwords and character substitutions, we wanted to understand how accurately users can rate random passwords and estimate the strength of passphrases. The latter strategies provide security benefits because such passwords require a high effort to guess them. However, those benefits are potentially unknown to the users [23,32]. To evaluate this quantitatively, we created PASDJO, an online game that awards points for accurate strength estimation. The game displays four different types of passwords and lets the user estimate their strength (see **Fig. 1**). Passwords are picked from four different categories: They are either (a) common passwords or (b) alterations of them, (c) combinations of two random dictionary words, or (d) randomly generated character sequences. The game selects one of the four categories at random with equal probabilities.

1.1 Contributions and Findings

Our work contributes novel insights into password strength perceptions and a method to acquire them. We show how a research question in the field of usable security can be studied inexpensively and reliably with a game-based approach. Moreover, we present following key insights on password strength estimation, which were drawn from real-world usage of our game:

a. Users are capable of assessing the strength of random passwords and of common passwords nearly equally well. We provide first quantitative evidence that users attribute the highest strength to random passwords, which highlights that users are generally aware of their benefits.
b. With our data, we show that passphrases are perceived as weaker than objective analyses show.
c. We corroborate Ur et al.'s findings [36] that common character substitutions are erroneously perceived to have a large positive effect on password strength.

The game as well as the dataset from the first four months after release are made publicly available on GitHub[2], which paves the way for further investigations.

1.2 Overview

The paper first presents an overview of related work and background information on password strength, policies, and user perceptions. We then explain the game mechanics and present usage analytics and results. The Limitations are discussed before the paper sheds light on future directions, and concludes with a reflection of the contribution.

2 Background and Related Work

We position our work in the field of usable security, particularly the study of passwords. In this section, we give a brief overview about the characteristics of strong passwords and how users go about creating them.

2.1 Password Strength Metrics

Finding an objective and reliable measure for the strength of a given password is difficult. The "NIST-entropy" of a password is a commonly used measure. It reports the degree of randomness of the characters inside a password (see Appendix A in [4]). However, as more advanced threat models emerged, more realistic measures were necessary. In offline-attack scenarios, attackers download the entire database containing the passwords as hashes. Such attacks sporadically occur even with highly frequented services like LinkedIn [11,27,31]. This means attackers can try millions of times to guess passwords and their efforts are only limited by time and computing power, which is difficult to defend against.

Multiple researchers proposed that the number of guesses required to crack a password is often a more accurate metric for strength than entropy [19,31,41]. To obtain the number of guesses, Carnegie Mellon University has established a Password Guessing Service (PGS) that allows uploading a list of passwords and receive success rates from various cracking approaches [39]. To use the service, however, the passwords need to be collected and uploaded in clear-text. This is not always possible in password studies, because sometimes participants re-use passwords from a real account for the study [24]. To avoid collecting disclosed passwords, there are other means to estimate the required number of guessing attempts, without storing the passwords. For example, the zxcvbn algorithm estimates strength similarly and shows high accuracy up to one million guesses, which is a realistic cut-off threshold for online attacks, which are easier to defend against [13,42]. The zxcvbn estimator can be implemented as a lightweight script and is easy to include in pro-active password checks.

[2] https://github.com/mimuc/pasdjo

2.2 Human Factors in Password Strength

Leaked data from real-world accounts has repeatedly shown that user-selected passwords are often predictable [1]. Given the freedom to select any password, many people opt for simple, short, memorable words or numbers that are easy to type. Because such passwords are vulnerable to informed guessing attacks, service providers try to prevent them by introducing a set of requirements to reduce the risk of account hijacking. However, such composition policies are not implemented identically on all web services [29,40]. Often, when users create an account, they re-use a password from elsewhere [7,14], which is prevented if policies differ in requirements. Users then tend to modify the password until the requirements are met [17,21]. The resulting passwords do not necessarily gain strength, if they are only extended by digits or symbols at predictable positions [41]. Thus, balancing the demands in terms of usability and security of a policy is challenging and has been under constant research in the past years [25,31,33,40].

Password policies are important for our work, because they affect how users evaluate password strength. The authoritarian character of policies induces an educational effect [6]. Unfortunately, users are often exposed to requirements that do not necessarily lead to stronger passwords. The length of a password is often more crucial for the strength than character diversity. For instance, when policies require three different character classes with minimum length twelve (3class12), user-selected passwords are often more guessable than those created with a simple length requirement of 16 characters (basic16) [33]. At the same time, users are being told that character variety is necessary to form strong passwords [37]. The long-term consequences are that users sometimes have a suboptimal perception of the factors that add to the objectively measurable strength of a password [36,38]. Some researchers investigated if passphrases, i.e. a combination of dictionary words, can effectively boost both usability and security. The results at this point are mixed, with some arguing in favor (e.g. [18]) and others against the usage of passphrases (e.g. [3,33]).

2.3 What Else Influences Password Choice?

Beside the constraints dictated by composition policies, many users have developed coping strategies for handling authentication tasks [34]. For example, the value of an account is decisive whether users pick a strong or weak password from their portfolio. Stobert and Biddle argue that this process is deliberate and even IT experts are prone to choose weak passwords for accounts that they do not deem worthy to protect [35]. Florêncio et al. argue that this behavior is inevitable if users do not use any digital aid, e.g. a password manager [12]. Still, if users receive security advice by trusted peers, they might reconsider behaviors like password re-use [8].

Apart from deliberate choice, there may be other preconditions that make some users pick stronger passwords than others. In a large field study, Mazurek et al. found that computer science and engineering students created passwords that were less guessable than those from business or politics students [24]. Beyond demographic background, context factors like the emotional state during password selection have also been investigated. Gulenko examined the effect of presenting positive textual messages and icons during password selection and found benefits for the adoption of passphrases [15]. Social pressure as another type of psychological leverage was investigated by Egelman et al. [10]. While they argue that the account value affects the effectiveness of password meters in the first place, others have shown that also the design of a password meter has a measurable impact on the effort users put into creating a password [37]. Moreover, password creation can be subtly influenced by suggesting strong passwords at the opportune moment. In a controlled online study, participants created stronger passwords if a longer password was shown beneath the password input field [30].

2.4 Summary

The related work suggests that users choose suboptimal passwords either by ***accident*** or by ***deliberate choice***. The former is more crucial, but might be solvable through better feedback. However, it is often difficult to distinguish the two cases. There is also mixed evidence about how users ***estimate*** the strength of their own passwords and what they think is necessary to create an appropriately strong password. To better understand potential misconceptions, we investigate how accurately users can assess password strength in a playful way.

3 Game Mechanics

The goal of PASDJO is to rate as many passwords as accurately as possible within 60 seconds. The game is easy to understand and play: When the game starts, passwords are displayed and their strength needs to be assessed by giving it one to five stars. In the following we explain the scoring, conditions, and design elements in the game.

3.1 Scoring Algorithm

The scores depend on the zxcvbn password strength estimator [42]. Its JavaScript implementation rates passwords on a scale from zero (weak) to four (strong), which is mapped to a scale from one star to five stars in PASDJO. The remainder of the paper uses the scale from one to five.

For an answer that matches the zxcvbn score, the player is awarded 100 points. We call the difference between the user rating and the zxcvbn score the ***deviation*** (D). In the worst case, a player's rating deviates by four stars, e.g. when they rate a password with five stars, while zxcvbn gives it a one-star rating. In this case, the player would not get any points. There is an error penalty of 100 / 4 = 25 points per error. Hence, rating a four-star password with only two stars will give the player a score of 50. The scores of each round are summed up and build

the ***achieved*** score (A). The game also calculates a ***percentage*** (P) of achieved and possible points at the end of the game.

The game thus has the following score calculation, where U is the user's estimation, Z is the zxcvbn score, and N is the number of passwords the user sees during the game:

$$D = U - Z \quad (1)$$

$$A = \sum_{k}^{N} 100 - (|D_k| * 25) \quad (2)$$

$$P = \frac{A}{N*100} \quad (3)$$

3.2 Password Generation Algorithms and Conditions

There are four ways in which PASDJO picks the displayed password. The condition during a round is chosen at random with equal probabilities.

Common passwords are taken without any modification from a list of commonly used passwords. They are shipped with the zxcvbn library and originate from data leaks of password databases at RockYou, Yahoo, and Xato [42]. There are 47023 all-lower-case passwords in this list. These passwords are usually very memorable but the least secure. The top 1000 passwords, e.g. "12345" or "password", score one star, while the rest receives two stars (e.g. "iloveyou2" or "thuglife").

Mangled passwords come from the identical password list, but are randomly altered with common substitutions and uppercasing. We only used substitutions that zxcvbn recognizes as "l33t" substitutions, for example **@** is mapped to the letter a[3]. We modify at most 30% of the characters this way. Moreover, at most 20% of the characters are transformed to uppercase. Mangling passwords is a strategy that users choose to make their passwords seemingly less predictable, which indeed often works [24]. Mangled passwords are rated with two stars most of the times. In very rare cases, they receive higher scores, e.g. "Qaz123wsx456" is rated with two stars, while a simple l33t substitution (s → $) would score 4 points in this particular case.

A **Passphrase** is a random combination of two dictionary words. We use the English Wikipedia 1-grams which are also shipped with zxcvbn. Only words between 4 and 11 characters in length are considered, leaving 27202 remaining words. This makes for $27202^2 \approx 10^9$ possible combinations, or ≈ 29 bits of entropy. We can assume that such passwords well withstand online attacks, where attackers are throttled in the number of guesses they can perform per second [13]. At the same time, they offer a memorability benefit [3,18,33]. Zxcvbn scores the resulting passphrases mostly with three or four stars.

[3] Matching table: https://github.com/dropbox/zxcvbn/blob/master/src/matching.coffee

Random passwords are generated using the German alphabet plus digits, i.e. 39 characters [a-z0-9äöü]. All random passwords are ten characters long and lowercase, which makes for $39^{10} \approx 10^{16}$ different combinations, or ≈ 52 bits of entropy. Such passwords are the most difficult to crack and require brute-force approaches [13,42]. However, from a usability perspective random passwords are more difficult to memorize and to type than, e.g., passphrases, so their benefit in online attack scenarios is questionable [13]. Zxcvbn scores random ten character passwords with four points.

For the remainder of the paper, we refer to these four conditions as "Common", "Mangled", "Passphrase" and "Random".

3.3 Feedback Screen

After the game, the player can review their achievements. The objective analyses and player ratings are displayed alongside to facilitate comparison. Moreover, the accuracy is color-coded (dark red = off by four, red = off by three, orange = off by two, light green = off by one, dark green = accurate). The score is displayed as fraction of achieved and maximum points, as well as a percentage. **Fig. 2** shows a screenshot of the feedback screen. From there, the players can reflect on how they came to their ratings and identify patterns in the objective analysis, which aims to produce a learning effect.

3.4 Game Design Elements

The most important game design elements are points and time pressure [9]. To keep the players motivated, the scoring algorithm and the conditions are designed to produce rather high scores. Since both one-star and five-star passwords are less likely to occur across all conditions, players will at least get 25 points per round most of the time. This aims to leverage the "goal gradient effect" [16], where the player is expected to intensify their efforts, the closer they are to the goal. In our case, the ultimate goal would be to achieve a score of 100%.

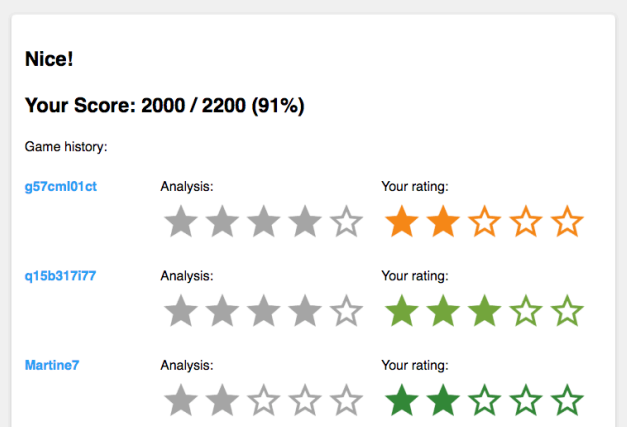

Fig. 2. The feedback screen lets the user review the game. Here, the user played 22 rounds and achieved 91% of the points. The accuracies for individual rounds are color coded to facilitate comparison of objective and subjective rating (deviation). There is no further explanation.

3.5 User Experience

The game was designed with user experience and usability in mind using the Progressive Web-App Design heuristics[4]. We hoped to lower the barrier to try out the game and make onboarding as easy as possible. The game can be started from any modern web browser. The user interface is automatically localized to German or English depending on the user's location and device settings. The welcome screen gives a brief instruction and lets the user try out the rating mechanics before they start the game. This facilitates getting comfortable with the interaction and preparing to act quickly. The rating task is also fairly simple to understand and there is no large time commitment, because one game takes at most 60 seconds. The game caches all necessary data to work off-line using the browser's IndexedDB APIs [20]. If the game is played without an internet connection, the browser synchronizes the game standings as soon as the connection becomes available. Finally, the user interface is responsive to various screen sizes and devices. Although user experience was part of the design and implementation phase, we did not evaluate this aspect because it goes beyond our research question.

3.6 Further Considerations and Disclaimers

When first visiting the site, users need to acknowledge a short disclaimer inside a notification window. The text says "We use cookies to collect anonymous statistics about the usage of this game. If you keep using our site, you agree to our Data Usage Terms.". The notification is displayed until the user actively acknowledges it. Furthermore, the "About" section of the web site makes transparent how the passwords are selected or generated, and that ratings are logged anonymously unless users choose to log in via their Google account. The same section also clearly states that the objective password ratings are just estimates and that passwords may be weaker or stronger in reality. This is to make users cautious not to directly use passwords from the game as their own. However, realistically, not all users will visit this part of the website, which is a small drawback of this study method.

Finally, we try to keep the amount of data to be transferred to a minimum because we expected many people would play the game on their mobile devices. However, we need to transfer the password lists to the devices to make it available offline. Thus, the first page-load consumes 1.9 MB of data.

4 Usage Analysis

We collected usage information in the first four months after deployment (December 2016 to March 2017). The game was advertised personally to peers, by putting up posters at our institution, and with demos during student orientation days.

[4] https://developers.google.com/web/progressive-web-apps/checklist

4.1 Log Data

The log data contains the minimum information required to assess password strength perceptions. We identify users by a unique user ID which is generated when the first game is started. Each game receives a unique game ID. The log data for each round contains the condition and a minified result of the strength estimation, i.e. the password in plain text, the score and the estimated number of guesses required to crack the password for professional attackers. Moreover, the user's rating as well as the deviation from the zxcvbn score is stored to the database. For example, if the password has a zxcvbn score of 3 and the user gave it a 1 star rating, we log the deviation as -2. Finally, to be able to look into the time taken, we also log a timestamp when the rating occurs.

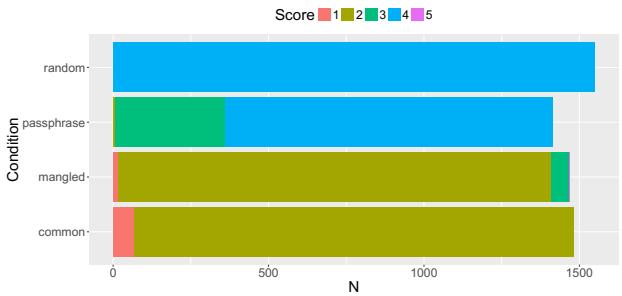

Fig. 3. The zxcvbn scores from 1 (weak) to 5 (strong) for N=5915 passwords in the four conditions are mostly predictable. Random passwords are consistently rated with score 4. Scores for Mangled passwords range from 1 to 5.

Users and Overall Score Distribution

115 users finished at least one game on their own device[5] and played 2.1 full games in average (SD = 2.83, Min = 1, Max = 16). All but two users chose to use the game without logging in. In total, we recorded 242 full games amounting to 5915 individual rounds (24.44 rounds per game, SD = 10.39). The zxcvbn scores were distributed mostly as expected – all random passwords had a score of 4 and all common passwords were rated with either score 1 or 2 (see **Fig. 3**). However, mangled passwords received the full spectrum of scores, although the large majority had score 2. Compared to the other conditions, which received fairly consistent strength ratings, passphrases are the least predictable condition. 24.9% of the passphrases received a score of 3, while 74.4 % resulted in a score of 4. These ratings can originate from the length of the randomly chosen words. Thus, upon correctly identifying the displayed password as a passphrase, the user has a lower chance of guessing correctly than in the other conditions. This aspect adds an element of unpredictability to the game, which is a common game design pattern [22].

[5] Although many more people played the game, we removed log data from shared computers that we had set up to have people try out the game during public events. However, we cannot rule out that a single user played the game on multiple devices, because anonymous IDs are generated per device.

Moreover, the achieved distribution of the system's password ratings makes underestimating common and mangled passwords less probable than underestimating e.g. passphrases or random passwords: a common or mangled password can usually be underestimated by one point, in case the user gives it a one star rating.

4.2 Statistics of the First Game

A user's first game informs us about their pre-existing perceptions of different passwords. After completing the first game and learning about its rating algorithm, the perception might change and should be evaluated separately in a later step.
On average, players achieved a score of 2010 points in their first game (SD = 653.16, Min = 350, Max = 3675), and they managed to play 27.21 rounds (SD = 8.56). This means it took 2.2 seconds to rate a password on average. Players initially achieved 74.58 percent of the points (SD = 8.34). On average, a game consisted of 6.9 Common (25.3%), 6.8 Mangled (25%), 7.2 Random passwords (26.5%), and 6.3 Passphrases (23.2%). Thus, the conditions were evenly distributed ($F(3) = 2.36$, $p > 0.05$).

4.2.1 Deviation from zxcvbn Score

Next, we evaluate by how much the players' estimation deviated from zxcvbn's score for a given password. In order to run valid pairwise tests for each password condition, we removed data from four users who did not rate a password for every condition in their first game, which leaves us with N=111 users. We use non-parametric tests to account for the lack of fine-grained password ratings.

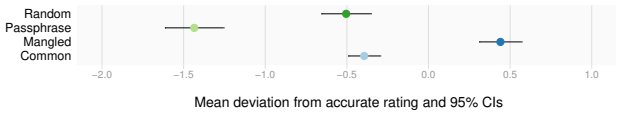

Fig. 4. Mean deviation from zxvbn score. The users' estimations were most inaccurate for passphrases, which they rated 1.4 stars lower than the zxcvbn estimator.

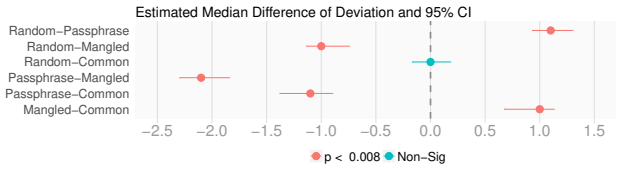

Fig. 5. Pairwise comparison of the estimated medians with confidence intervals. Accuracy for random and common passwords was similar, while other pairs show significant differences.

Mangled passwords were overestimated compared to the score of zxcvbn (estimated median deviation Md = 0.5). While common and random passwords were only slightly underestimated (Md = -0.5), the passphrases were rated worst (Md =-1.6). The means and confidence intervals are visualized in **Fig. 4**. A Friedman rank-sum test showed significant differences regarding the deviation in the four conditions ($F(3) = 187.84$, $p < 0.001$). On a Bonferroni-corrected significance level of $\alpha_{Bonf} = 0.008$, post-hoc Wilcoxon paired sample tests showed significantly different deviations between all conditions, except between common and random passwords deviations (see **Fig. 5**). This means the users' over-/underestimation differs significantly across most conditions.

4.2.2 Multiple Game Score Development

33 users played at least two games. We performed a linear regression with gameIndex as predictor and achieved percentage as dependent variable, weighted by total rounds played. The model shows that, on average, players are able to improve their accuracy significantly when playing more often ($F(1) = 49.37$, $p < 0.001$, $\beta = 0.54$, $R^2_{adj} = 0.23$). **Fig. 6** shows the development of the achieved percentage for users who played multiple times. While playing more often shows a learning effect, the element of randomness (conditions, passwords) might lead to some games being more difficult than others. This could explain why players do not consistently become better with each game played.

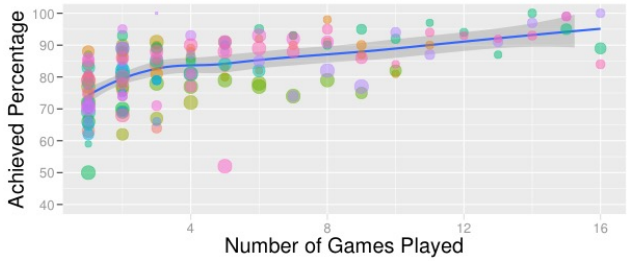

Fig. 6. Series of achieved percentages by individual players with fitted line. Larger dots indicate more rounds per game. The achieved percentages form an upwards trend.

5 Discussion

The data shows how users perceive the strength of different password creation styles. In this section, we point out the implications the effects have on the design of persuasive password feedback.

5.1 Users know what makes for very weak and very strong passwords.

From the results of the players' first games we can infer that users' perceptions of common, mangled, and random passwords are fairly consistent with the scores of zxcvbn. Here, they were only off by 0.5 stars on average. Although we used different methods, we were able to corroborate findings by Ur et al. [36]. We can conclude that if users select weak, common passwords, they are well aware of the consequences. In particular, we hypothesize that deliberately choosing a common password indicates how valuable account is to the user. Therefore, we propose that services react differently to a user's deliberate choice of common passwords. Instead of displaying password meters or enforcing their policies, it might be a good idea to adapt the service itself to the reduced security level. Thus, reacting to common passwords, the service might limit the

amount of sensitive information in the user's profile to prevent financial damage or identity theft. Once users pick stronger passwords, those options can be re-enabled.

Moreover, the random passwords in our game were all lower-case letters plus digits. The lack of uppercase letters may have led to a slight underestimation, as the paper by Ur et al. suggests [36]. However, in their study, participants attributed the highest self-reported security benefit to random passwords, which we can now confirm quantitatively with our data.

5.2 Strength estimation is learnable.

We observed that players found out how to beat the game the longer they played. Consequently, playing the game had an educational effect regarding password strength. On a larger scale, we believe that user behavior regarding password usage can be influenced with better positive reinforcement. For those users who actually played more than one round (29%), achieving a higher score than before was intrinsically motivated, because there was no obligation to play the game at all. If feedback mechanisms can induce the same level of intrinsic motivation, i.e. users ***wanting*** to make their own passwords stronger, it is likely an improved feedback actually achieves this goal.

For instance, password managers already try to leverage this effect by calculating general "security scores" that are supposed to persuade users to increase their scores, e.g. if users update certain passwords. However, how do we apply this tactic for users who do not use password management software? Here again, service-providers could have more responsibility and show a security score to each user on their profile pages. This might help them learn what causes a score-boost and ultimately motivate them to change their behavior - at least for services that become more valuable to them over time. A social network, for example, might implement the ***social proof*** persuasion pattern [6] and inform users how many of their peers have activated the security score feature.

5.3 Generated passphrases are fine, but user-selected passphrases probably are not.

The theoretical password space of passphrases is high, given the number of entries is in the dictionary is large enough. In our case, almost 30,000 words were considered and zxcvbn scored the majority of two-word passphrases were with four out of five points. However, it is very unlikely that users draw from a vocabulary this large when they select their passphrase[6]. Bonneau and Shutova also point out that two-word passphrases are much more predictable if users select the words [3]. Thus, if system designers aim to persuade users to choose a ***strong*** passphrase, they have to suggest a random passphrase to them. However, this may not be a satisfying solution due to the low acceptance of randomly generated passphrases – users usually want passwords that have a meaning for them. However, Seitz et al. showed how passphrase-suggestions can still have a positive effect on ***self***-selected passwords [30]. Moreover, Shay et al. studied system-assigned passphrases and concluded that participants found them too unattractive [32]. Our participants were doubtful about the strength of passphrases, too. In conclusion, if we take the users' perspective and account for the reduced vocabulary, the users' rating might in fact reflect the strength of real-world passphrases better than the zxcvbn score.

5.4 Limitations

As PASDJO was publicly deployed, there are a number of limitations in the data that arise from this methodology. First, all players can choose to remain anonymous. They are only identified by a cookie containing a random user ID, which can be deleted from the browser. Thus, unless users sign in and associate the game play history with their Google account, we lack demographic information. The context makes young adults the most likely user group: During the first four months after releasing PASDJO, there were no other announcements than word-of-mouth and posters around campus. The posters challenged the skills of passers-by. While this may be a side-effect on the results of the game, the distribution of IT-security knowledge across campus is not completely different than in other populations. Yet, we must be careful not expect the same password estimations from a larger audience. Despite these limitations caused by anonymity, the benefits of collecting measurements through real-world usage instead of artificial lab scenarios boost the ecological validity of the data.

Moreover, to sign in, we specifically did not offer password-based authentication, as this might scare off users: Creating passwords at a game-site dedicated towards password strength could make users think their password is scored and saved to the game, too. The high number of people who chose to remain anonymous speaks in favor of this decision.

The password strength ratings in PASDJO are somewhat opinionated due to the use of the strength rating approach as implemented by zxcvbn. Still, zxcvbn is a scientifically evaluated, state-of-the art tool, which was shown to deliver reliable estimations when compared to more sophisticated strength metrics. This gives us confidence that scores can be considered accurate regarding on-line guessing attacks. Moreover, it simplifies the task of matching user ratings and password scores, because other metrics do not implement a [1;N] scoring algorithm. However, it would be interesting to add other strength rating mechanisms (e.g. neural network guessing [26]) to evaluate their consistency with user ratings and vice versa.

[6] We could not find consistent, convincing estimates of a native speaker's active vocabulary, but there is some evidence that 1000 different words make up ≈ 85% of spoken language on TV (https://hackernoon.com/learning-languages-very-quickly-with-the-help-of-some-very-basic-data-science-cdbf95288333)

6 Future Work

At the moment, PASDJO's features and game mechanics were sufficient to assess the feasibility of studying password perception in the wild. The data can serve as a baseline for future studies. For example, the next step is to move from password perceptions to actions. Currently, we do not know if playing the game has a measureable effect on choosing passwords. Thus, a lab experiment which studies password selection after participants had played the game for a while appears worthwhile.

We are also confident that other modules can be plugged into the game to conduct a larger variety of studies on passwords. Therefore, we released PASDJO as open source software on GitHub as well as the anonymous data that we collected so far. For example, this also allows it to be used as an educational tool at schools. Alternatively, if demographic data is required for a certain study, researchers can easily plug-in new modules to collect this kind of information.

We plan to use new versions of PASDJO in fundamental research on authentication mechanisms. We are particularly curious about the influence of personality traits on the perception and selection of passwords. We aim to answer the question which user groups are more accurate in their strength assessments and why. To do this, we need to implement privacy-sensitive ways to collect additional information about the players.

8 Conclusion

In this paper, we presented a new game about passwords that was used to collect and analyze strength perceptions of different password categories. The game allows assessing the problem that some users lack an accurate understanding of what contributes to password strength. This kind of perception is important when users actively seek to create strong passwords. During the first four months of public usage, we found that many users are fairly accurate in their assessment of common, mangled, and random passwords. On the other hand, they perceived two-word passphrases as significantly weaker than objective measures. This suggests that randomly generated passphrases require more explanatory text as to their strength in order to convince users of their benefits or when to use them. For instance, when users create their master password in password managers, they may be offered a generated, memorable passphrase, which should then be accompanied by a reasonable explanation why this password is sufficiently strong and appropriate in this context.

Our results can more generally inform the design of feedback systems during password selection, as well as security policies. For example, we argue that while common passwords can be considered a dangerous choice, users are probably aware of this risk. Feedback like password meters should respect users' choices carefully to favor usability over security. At the same time, services can decide to offer different levels of functionality depending on how securely individual users use the services. Personalization as a persuasive measure could be an effective motivator of secure actions [28]. This may improve the user experience of password-based authentication, and reduce the burden of passwords until a viable solution to replace them is found [2].

ACKNOWLEDGMENTS

We would like to thank our proof readers, Mohamed Khamis and Ceenu George, as well as all the reviewers who provided valuable and insightful feedback on how to improve our manuscript.

REFERENCES

[1] Joseph Bonneau. 2012. The Science of Guessing: Analyzing an Anonymized Corpus of 70 Million Passwords. ***Proceedings - IEEE Symposium on Security and Privacy***, IEEE Comput. Soc, 538–552.

[2] Joseph Bonneau, Cormac Herley, Paul C. Van Oorschot, and Frank Stajano. 2015. Passwords and the Evolution of Imperfect Authentication. ***Communications of the ACM*** 58, 7: 78–87.

[3] Joseph Bonneau and Ekaterina Shutova. 2012. Linguistic properties of multi-word passphrases. ***Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*** 7398 LNCS: 1–12.

[4] William E. Burr, Donna F. Dodson, Elaine M. Newton, Ray A. Perlner, and W. Timothy Polk. 2011. ***SP 800-63-1. Electronic Authentication Guideline***. Gaithersburg, MD, United States.

[5] Xavier De Carné De Carnavalet and Mohammad Mannan. 2014. From Very Weak to Very Strong : Analyzing Password-Strength Meters. ***Ndss 2014*** February: 23–26.

[6] Robert B Cialdini. 2007. ***Influence: The Psychology of Persuasion***. Harper-Collins.

[7] Anupam Das, Joseph Bonneau, Matthew Caesar, Nikita Borisov, and Xf Wang. 2014. The Tangled Web of Password Reuse. ***Proceedings of Network and Distributed System Security Symposium (NDSS 14)***, Internet Society, 23–26.

[8] Sauvik Das, THJ Kim, LA Dabbish, and JI Hong. 2014. The Effect of Social Influence on Security Sensitivity. ***Proceedings of the 10th Symposium On Usable Privacy and Security (SOUPS'14)***, 143–157.

[9] Sebastian Deterding, Dan Dixon, Rilla Khaled, and Lennart Nacke. 2011. From game design elements to gamefulness: defining gamification. ***Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments***, 9–15.

[10] Serge Egelman, Andreas Sotirakopoulos, Ildar Muslukhov, Konstantin Beznosov, and Cormac Herley. 2013. Does My Password Go Up to Eleven?: The Impact of Password Meters on Password Selection. ***Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)***, 2379–2388.

[11] Dinei Florêncio, Cormac Herley, and Baris Coskun. 2007. Do strong web passwords accomplish anything? ***Security***: 10.

[12] Dinei Florêncio, Cormac Herley, and Paul C. Van Oorschot. 2014. Password Portfolios and the Finite-Effort User: Sustainably Managing Large Numbers of Accounts. ***Proceedings of USENIX Security Symposium***, USENIX Association, 575–590.

[13] Dinei Florêncio, Cormac Herley, and Paul C Van Oorschot. 2014. An Administrator's Guide to Internet Password Research. ***Proceedings of the 28th Large Installation System Administration Conference (LISA14)***, USENIX Association, 35–52.

[14] Shirley Gaw and Edward W. Felten. 2006. Password management strategies for online accounts. ***Proceedings of the second symposium on Usable privacy and security (SOUPS '06)***, ACM, 44–55.

[15] Iwan Gulenko. 2014. Improving passwords: influence of emotions on security behaviour. ***Information Management & Computer Security*** 22, 2: 167–178.

[16] Juho Hamari. 2011. Framework for Designing and Evaluating Game Achievements. ***Proceedings of DiGRA 2011 Conference: Think Design Play***, 122–134.

[17] Philip Inglesant and Martina Angela Sasse. 2010. The True Cost of Unusable Password Policies: Password Use in the Wild. ***Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '10)***, 383–392.

[18] Mark Keith, Benjamin Shao, and Paul Steinbart. 2009. A Behavioral Analysis of Passphrase Design and Effectiveness. ***Journal of the Association for Information Systems*** 10, 2: 63–89.

[19] Patrick Gage Kelley, Saranga Komanduri, Michelle L. Mazurek, et al. 2012.

Guess again (and again and again): Measuring password strength by simulating password-cracking algorithms. ***Proceedings - IEEE Symposium on Security and Privacy***, 523–537.
[20] Stefan Kimak and Jeremy Ellman. 2016. The role of HTML5 IndexedDB, the past, present and future. ***2015 10th International Conference for Internet Technology and Secured Transactions, ICITST 2015***, 379–383.
[21] Saranga Komanduri, Richard Shay, Patrick Gage Kelley, et al. 2011. Of Passwords and People. ***Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11)***, 2595–2604.
[22] Raph Koster. 2005. Theory of Fun for Game Design. ***A Theory of Fun for Game Design***: 256.
[23] Michael D. Leonhard and V. N. Venkatakrishnan. 2007. A Comparative Study of Three Random Password Generators. ***2007 IEEE International Conference on Electro/Information Technology, EIT 2007***, IEEE, 227–232.
[24] Michelle L. Mazurek, Saranga Komanduri, Timothy Vidas, et al. 2013. Measuring password guessability for an entire university. ***Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security - CCS '13***, 173–186.
[25] William Melicher, Darya Kurilova, Sean M Segreti, et al. 2016. Usability and Security of Text Passwords on Mobile Devices. ***Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems***, 527–539.
[26] William Melicher, Blase Ur, Sean M Segreti, et al. 2016. Fast, Lean, and Accurate: Modeling Password Guessability Using Neural Networks. ***Usenix Security***.
[27] Cory Scott. 2016. Protecting Our Members, LinkedIn Official Blog. Retrieved September 1, 2016 from https://blog.linkedin.com/2016/05/18/protecting-our-members.
[28] Tobias Seitz. 2017. Personalizing Password Policies and Strength Feedback. ***Personalizing Persuasive Technologies Workshop (Persuasive '17 adjunct)***, Springer Berlin Heidelberg.
[29] Tobias Seitz, Manuel Hartmann, Jakob Pfab, and Samuel Souque. 2017. Do Differences in Password Policies Prevent Password Reuse? ***CHI '17 Extended Abstracts on Human Factors in Computing Systems***, ACM.
[30] Tobias Seitz, Emanuel von Zezschwitz, Stefanie Meitner, and Heinrich Hussmann. 2016. Influencing Self-Selected Passwords Through Suggestions and the Decoy Effect. ***Proceedings of the 1st European Workshop on Usable Security***, Internet Society, 2:1-2:7.
[31] Richard Shay, Adam L Durity, Sean M Segreti, Blase Ur, Lujo Bauer, and Nicolas Christin. 2016. Designing Password Policies for Strength and Usability. ***ACM Transactions on Information and System Security*** 18, 4: 13:1-13:34.
[32] Richard Shay, Patrick Gage Kelley, Saranga Komanduri, et al. 2012. Correct Horse Battery Staple. ***Proceedings of the Eighth Symposium on Usable Privacy and Security (SOUPS '12)***, ACM, 1–20.
[33] Richard Shay, Saranga Komanduri, Adam L Durity, et al. 2014. Can Long Passwords Be Secure and Usable? ***Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)***.
[34] Elizabeth Stobert and Robert Biddle. 2014. The Password Life Cycle: User Behaviour in Managing Passwords. ***Proceedings of the 10th Symposium On Usable Privacy and Security (SOUPS '14)***, ACM, 243–255.
[35] Elizabeth Stobert and Robert Biddle. 2015. Expert Password Management. ***Proceedings of Passwords 2015***, Springer International Publishing, 3–20.
[36] Blase Ur, Jonathan Bees, Sean M Segreti, Lujo Bauer, Nicolas Christin, and Lorrie Faith Cranor. 2016. Do Users' Perceptions of Password Security Match Reality? ***Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)***, ACM, 3748–3760.
[37] Blase Ur, Patrick Gage Kelley, Saranga Komanduri, et al. 2012. How Does Your Password Measure Up? The Effect of Strength Meters on Password Creation. ***Security'12 Proceedings of the 21st USENIX conference on Security symposium***, 5–16.
[38] Blase Ur, Fumiko Noma, Jonathan Bees, et al. 2015. " I Added '!' at the End to Make It Secure ": Observing Password Creation in the Lab. 123–140.
[39] Blase Ur, Sean M Segreti, Lujo Bauer, et al. 2015. Measuring Real-World Accuracies and Biases in Modeling Password Guessability. ***24th USENIX Security Symposium (USENIX Security 15)***, USENIX Association, 463--481.
[40] Ding Wang and Ping Wang. 2015. The Emperor's New Password Creation Policies. ***Proceedings of the 20th European Symposium on research in Computer Security - ESORICS'15***, Springer, 456–477.
[41] Matt Weir, Sudhir Aggarwal, Michael Collins, and Henry Stern. 2010. Testing metrics for password creation policies by attacking large sets of revealed passwords. ***Proceedings of the 17th ACM conference on Computer and communications security - CCS '10***: 162.
[42] Daniel Lowe Wheeler. 2016. zxcvbn: Low-Budget Password Strength Estimation. ***25th USENIX Security Symposium (USENIX Security 16)***, USENIX Association, 157–173.

All web links were last followed on June 11 2017