How Does Your Password Measure Up?
The Effect of Strength Meters on Password Creation

Blase Ur, Patrick Gage Kelley, Saranga Komanduri, Joel Lee, Michael Maass,
Michelle L. Mazurek, Timothy Passaro, Richard Shay, Timothy Vidas,
Lujo Bauer, Nicolas Christin, Lorrie Faith Cranor
Carnegie Mellon University
{*bur, pgage, sarangak, jlee, mmaass, mmazurek, tpassaro,
rshay, tvidas, lbauer, nicolasc, lorrie*}*@cmu.edu*

Abstract

To help users create stronger text-based passwords, many web sites have deployed password meters that provide visual feedback on password strength. Although these meters are in wide use, their effects on the security and usability of passwords have not been well studied.

We present a 2,931-subject study of password creation in the presence of 14 password meters. We found that meters with a variety of visual appearances led users to create longer passwords. However, significant increases in resistance to a password-cracking algorithm were only achieved using meters that scored passwords stringently. These stringent meters also led participants to include more digits, symbols, and uppercase letters.

Password meters also affected the act of password creation. Participants who saw stringent meters spent longer creating their password and were more likely to change their password while entering it, yet they were also more likely to find the password meter annoying. However, the most stringent meter and those without visual bars caused participants to place less importance on satisfying the meter. Participants who saw more lenient meters tried to fill the meter and were averse to choosing passwords a meter deemed "bad" or "poor." Our findings can serve as guidelines for administrators seeking to nudge users towards stronger passwords.

1 Introduction

While the premature obituary of passwords has been written time and again [22, 25], text passwords remain ubiquitous [15]. Unfortunately, users often create passwords that are memorable but easy to guess [2, 25, 26]. To combat this behavior, system administrators employ a number of measures, including system-assigned passwords and stringent password-composition policies. System-assigned passwords can easily be made difficult to guess, but users often struggle to remember them [13] or write them down [28]. Password-composition policies, sets of requirements that every password on a system must meet, can also make passwords more difficult to guess [6, 38]. However, strict policies can lead to user frustration [29], and users may fulfill requirements in ways that are simple and predictable [6].

Another measure for encouraging users to create stronger passwords is the use of password meters. A password meter is a visual representation of password strength, often presented as a colored bar on screen. Password meters employ suggestions to assist users in creating stronger passwords. Many popular websites, from Google to Twitter, employ password meters.

Despite their widespread use, password meters have not been well studied. This paper contributes what we believe to be the first large-scale study of what effect, if any, password meters with different scoring algorithms and visual components, such as color and size, have on the security and usability of passwords users create.

We begin by surveying password meters in use on popular websites. Drawing from our observations, we create a control condition without a meter and 14 conditions with meters varying in visual features or scoring algorithm. The only policy enforced is that passwords contain at least eight characters. However, the meter nudges the user toward more complex or longer passwords.

We found that using any of the tested password meters led users to create passwords that were statistically significantly longer than those created without a meter. Meters that scored passwords more stringently led to even longer passwords than a baseline password meter. These stringent meters also led participants to include a greater number of digits, symbols, and uppercase letters.

We also simulated a state-of-the-art password-cracking algorithm [38] and compared the percentage of passwords cracked in each condition by adversaries making 500 million, 50 billion, and 5 trillion guesses. Passwords created without a meter were cracked at a higher rate than passwords in any of the 14 conditions with meters, although most differences were not statistically significant. Only passwords created in the presence of the two stringent meters with visual bars were cracked at a significantly lower rate than those created without a meter. None of the conditions approximating meters we observed in the wild significantly increased cracking resistance, suggesting that currently deployed meters are not sufficiently aggressive. However, we also found that users have expectations about good passwords and can only be pushed so far before aggressive meters seem to annoy users rather than improve security.

We next review related work and provide background in Section 2. We then survey popular websites' password meters in Section 3 and present our methodology in Section 4. Section 5 contains results related to password composition, cracking, and creation, while Section 6 summarizes participants' attitudes. We discuss these findings in Section 7 and conclude in Section 8.

2 Related Work

Prior work related to password meters has focused on password scoring rather than how meters affect the security and usability of passwords users create. We summarize this prior work on password scoring, and we then discuss more general work on the visual display of indicators. In addition, we review work analyzing security and usability tradeoffs in password-composition policies. Finally, we discuss the "guessability" metric we use to evaluate password strength.

2.1 Password Meters

Algorithms for estimating password strength have been the focus of prior work. Sotirakopoulos et al. investigated a password meter that compares the strength of a user's password with those of other users [31]. Castelluccia et al. argued that traditional rule-based password meters lack sufficient complexity to guide users to diverse passwords, and proposed an adaptive Markov algorithm that considers n-gram probabilities in training data [7]. In contrast, we use simple rule-based algorithms to estimate strength, focusing on how meters affect the usability and security of the passwords users create. To our knowledge, there has been no formal large-scale study of interface design for password meters.

Many password meters guide users toward, but do not strictly require, complex passwords. This approach reflects the behavioral economics concept of nudging or soft paternalism [24, 34]. By helping users make better decisions through known behavioral patterns and biases, corporations, governments, and other entities have induced a range of behavioral changes from investing more toward retirement to eating more fruit.

2.2 Visual Display of Indicators

While the literature on visual design for password meters is sparse, there is a large corpus of work in information design generally. For instance, researchers have studied progress indicators in online questionnaires, finding that indicators can improve user experience if the indicator shows faster progress than a user anticipated. However, progress that lags behind a user's own expectations can cause the user to abandon the task at hand [8].

Much of the past work on small meters has focused on physical and virtual dashboards [11]. Information design has also been studied in consumer-choice situations, such as nutrition labels [19] and over-the-counter drug labels, focusing on whitespace, font size, and format [40].

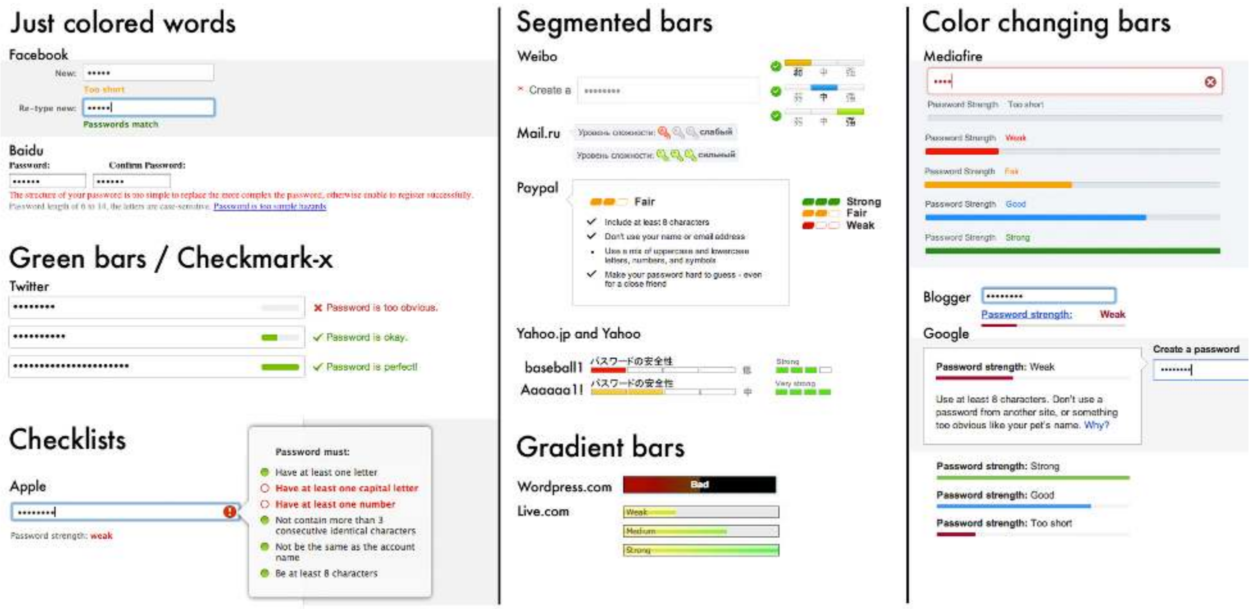
2.3 Password-Composition Policies

In this paper, we examine security and usability tradeoffs related to nudging users with password meters, rather than imposing strict requirements. Significant work has been done evaluating tradeoffs for enforced password-composition policies.

Without intervention, users tend to create simple passwords [12, 23, 33, 41]. Many organizations use password-composition policies that force users to select more complex passwords to increase password strength. However, users are expected to conform to these policies in predictable ways, potentially reducing password strength [6]. Although prior work has shown that password-composition policies requiring more characters or more character classes can improve resistance to automated guessing attacks, many passwords that meet common policies remain vulnerable [18, 26, 37, 38]. Furthermore, strict policies can frustrate users, inhibit their productivity, and lead users to write their passwords down [1, 14, 16, 21, 32].

2.4 Measuring Guessability

In this work, we use "guessability," or resistance to automated password-cracking attacks, to evaluate the strength of passwords. Guessability cannot be measured as a single statistic for a set of passwords; instead, a given algorithm, with a given set of parameters and training, will crack some percentage of the passwords after a given number of guesses. Weir et al. argue that guessability is a more accurate measure of password strength than the more commonly used entropy metric [38]. Dell'Amico et al. [9], Bonneau [3], and Castelluccia et al. [7] have also used guessability as a metric. We measure guessability using a guess-number calculator, which computes how many guesses a given cracking algorithm will require to crack a specific password without running the algorithm itself [18].



The structure of your password is too simple to replace the more complex the password, otherwise unable to register successfully.
Password length of 6 to 14, the letters are case-sensitive. Password is too simple hazards

Password must:
- ● Have at least one letter
- ○ Have at least one capital letter
- ○ Have at least one number
- ● Not contain more than 3 consecutive identical characters
- ● Not be the same as the account name
- ● Be at least 8 characters

- ✓ Include at least 8 characters
- ✓ Don't use your name or email address
- Use a mix of uppercase and lowercase letters, numbers, and symbols
- ✓ Make your password hard to guess - even for a close friend

Use at least 8 characters. Don't use a password from another site, or something too obvious like your pet's name. Why?

Figure 1: A categorized assortment of the 46 unique indicators we found across Alexa's 100 most visited global sites.

3 Password Meters "In the Wild"

To understand how password meters are currently used, we examined Alexa's 100 most visited global sites (collected January 2012). Among these 100 sites, 96 allowed users to register and create a password. Of these 96, 70 sites (73%) gave feedback on a user's password based either on its length or using a set of heuristics. The remaining 26 sites (27%) provided no feedback. In some cases, all sites owned by the same company used the same meter; for example, Google used the same meter on all 27 of its affiliates that we examined. In other cases, the meters varied; for example, ebay.de used a different mechanism than ebay.com. Removing duplicate indicators and sites without feedback, there were 46 unique indicators. Examples of these indicators are shown in Figure 1.

Indicators included bar-like meters that displayed strength (23, 50%); checkmark-or-x systems (19, 41.3%); and text, often in red, indicating invalid characters and too-short passwords (10, 21.2%). Sites with bar-like meters used either a progress-bar metaphor (13, 56.5%) or a segmented-box metaphor (8, 34.8%). Two sites presented a bar that was always completely filled but changed color (from red to green or blue) as password complexity increased. Three other sites used meters colored with a continuous gradient that was revealed as users typed. Sites commonly warned about insecure passwords using the words "weak" and "bad."

We examined scoring mechanisms both by reading the Javascript source of the page, when available, and by testing sample passwords in each meter. Across all meters, general scoring categories included password length, the use of numbers, uppercase letters, and special characters, and the use of blacklisted words. Most meters updated dynamically as characters were typed.

Some meters had unique visual characteristics. Twitter's bar was always green, while the warning text changed from red to green. Twitter offered phrases such as "Password could be more secure" and "Password is Perfect." The site mail.ru had a three-segment bar with key-shaped segments, while rakuten.co.jp had a meter with a spring-like animation.

We found some inconsistencies across domains. Both yahoo.com and yahoo.co.jp used a meter with four segments; however, the scoring algorithm differed, as shown in Figure 1. Google used the same meter across all affiliated sites, yet its meter on blogger.com scored passwords more stringently.

4 Methodology

We conducted a two-part online study of password-strength meters, recruiting participants through Amazon's Mechanical Turk crowdsourcing service (MTurk). Participants, who were paid 55 cents, needed to indicate that they were at least 18 years old and use a web browser with JavaScript enabled. Participants were assigned round-robin to one of 15 conditions, detailed in Section 4.2. We asked each participant to imagine that his or her main email provider had changed its password requirements, and that he or she needed to create a new password. We then asked the participant to create a password using the interface shown in Figure 2.

Passwords needed to contain at least eight characters, but there were no other requirements. The participant was told he or she would be asked to return in a few days to log in with the password. He or she then completed a survey about the password-creation experience and was asked to reenter his or her password at the end.

Two days later, participants received an email through MTurk inviting them to return for a bonus payment of 70 cents. Participants were asked to log in again with their password and to take another survey about how they handled their password.

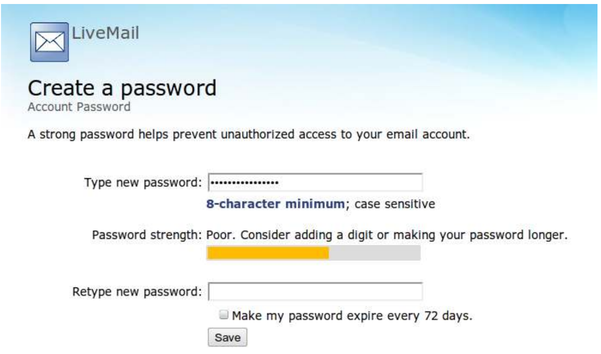
4.1 Password-Scoring Algorithms

Password-strength meters utilize a scoring function to judge the strength of a password, displaying this score through visual elements. We assigned passwords a score using heuristics including the password's length and the character classes it contained. While alternative approaches to scoring have been proposed, as discussed in Section 2, judging a password only on heuristics obviates the need for a large, existing dataset of passwords and can be implemented quickly in Javascript. These heuristics were based on those we observed in the wild.

In our scoring system, a score of 0 points represented a blank password field, while a score of 100 points filled the meter and displayed the text "excellent." We announced our only password-composition policy in bold text to the participant as an "8-character minimum" requirement. However, we designed our scoring algorithm to assign passwords containing eight lowercase letters a score of 32, displaying "bad." To receive a score of 100 in most conditions, participants needed to meet one of two policies identified as stronger in the literature [6,21], which we term *Basic16* and *Comprehensive8*. Unless otherwise specified by the condition, passwords were assigned the larger of their Basic16 and Comprehensive8 scores. Thus, a password meeting either policy would fill the meter. Each keystroke resulted in a recalculation of the score and update of the meter.

The *Basic16* policy specifies that a password contain at least 16 characters, with no further restrictions. In our scoring system, the first 8 characters entered each received 4 points, while all subsequent characters received 8 points. Thus, passwords such as *aaaaaaaaaaaaaaaa*, *WdH5$87T5c#hgfd&*, and *passwordpassword* would all fill the meter with scores of exactly 100 points.

The second policy, *Comprehensive8*, specifies that a password contain at least eight characters, including an uppercase letter, a lowercase letter, a digit, and a symbol. Furthermore, this password must not be in the OpenWall Mangled Wordlists, which is a cracking dictionary.[1] In our scoring system, 4 points were awarded for each character in the password, and an additional 17 points were awarded each for the inclusion of an uppercase character, a digit, and a symbol; 17 points were deducted if the password contained no lowercase letters. A second unique digit, symbol, or uppercase character would add an additional 8 points, while a third would add an additional 4 points. Passing the dictionary check conferred 17 points. Therefore, passwords such as *P4$sword*, *gT7fas#g*, and *N!ck1ebk* would fill the meter with a score of exactly 100. In addition, passwords that were hybrids of the two policies, such as a 13-character password meeting Comprehensive8 except containing no symbols, could also fill the meter.



LiveMail

Create a password

Account Password

A strong password helps prevent unauthorized access to your email account.

Type new password: ••••••••••••••••

8-character minimum; case sensitive

Password strength: Poor. Consider adding a digit or making your password longer.

Retype new password:

□ Make my password expire every 72 days.

Save

Figure 2: An example of the password creation page. The password meter's appearance and scoring varied by condition.

4.2 Conditions

Our 15 conditions fall into four main categories. The first category contains the two conditions to which we compared the others: having no password meter and having a baseline password meter. Conditions in the next category differ from the baseline meter in only one aspect of visual presentation, but the scoring remains the same. In contrast, conditions in the third category have the same visual presentation as the baseline meter, but are scored differently. Finally, we group together three conditions that differ in multiple dimensions from the baseline meter. In addition, we collectively refer to *half-score*, *one-third-score*, *text-only half-score*, and *text-only half-score* as the *stringent* conditions throughout the paper. Each participant was assigned round-robin to one condition.

4.2.1 Control Conditions

No meter. This condition, our control, uses no visual feedback mechanism. 26 of the Alexa Top 100 websites provided no feedback on password strength, and this condition allows us to isolate the effect of the visual feedback in our other conditions.

[1] http://www.openwall.com/wordlists/

Baseline meter. This condition represents our default password meter. The score is the higher of the scores derived from comparing the password to the Basic16 and Comprehensive8 policies, where a password meeting either policy fills the bar. The color changes from red to yellow to green as the score increases. We also provide a suggestion, such as "Consider adding a digit or making your password longer." This condition is a synthesis of meters we observed in the wild.

4.2.2 Conditions Differing in Appearance

Three-segment. This condition is similar to *baseline meter*, except the continuously increasing bar is replaced with a bar with three distinct segments, similar to meters from Google and Mediafire.

Green. This condition is similar to *baseline meter*, except instead of changing color as the password score increases, the bar is always green, like Twitter's meter.

Tiny. This condition is similar to *baseline meter*, but with the meter's size decreased by 50% horizontally and 60% vertically, similar to the size of Google's meter.

Huge. This condition is similar to *baseline meter*, but with the size of the meter increased by 50% horizontally and 120% vertically.

No suggestions. This condition is similar to *baseline meter*, but does not offer suggestions for improvement.

Text-only. This condition contains all of the text of *baseline meter*, but has no visual bar graphic.

4.2.3 Conditions Differing in Scoring

Half-score. This condition is similar to *baseline meter*, except that the password's strength is displayed as if it had received half the rating. A password that would fill the *baseline meter* meter only fills this condition's meter half way, allowing us to study nudging the participant toward a stronger password. A password with 28 characters, or one with 21 characters that included five different uppercase letters, five different digits, and five different symbols, would fill this meter.

One-third-score. This condition is similar to *half-score*, except that the password's strength is displayed as if it had received one-third the rating. A password that would fill the *baseline meter* meter only fills one-third of this condition's meter. A password containing 40 characters would fill this meter.

Nudge-16. This condition is similar to *baseline meter*, except that only the password score for the Basic16 policy is calculated, allowing us to examine nudging the user toward a specific password policy.

Nudge-comp8. As with *nudge-16*, this condition is similar to *baseline meter*, except that only the password score for Comprehensive8 is calculated.

4.2.4 Conditions Differing in Multiple Ways

Text-only half-score. As with *text-only*, this condition contains all of the text of *baseline meter*, yet has no bar. Furthermore, like *half-score*, the password's strength is displayed as if it had received only half the score.

Bold text-only half-score. This condition mirrors *text-only half-score*, except the text is displayed in bold.

Bunny. In place of a bar, the password score is reflected in the speed at which an animated Bugs Bunny dances. When the score is 0, he stands still. His speed increases with the score; at a score of 100, he dances at 20 frames per second; at a score of 200, he reaches his maximum of 50 frames per second. This condition explores a visual feedback mechanism other than a traditional bar.

4.3 Mechanical Turk

Many researchers have examined using MTurk workers for human-subjects research and found it to be a convenient source of high-quality data [5, 10, 20, 35]. MTurk enables us to have a high volume of participants create passwords, on a web site we control, with better population diversity than would be available in an on-campus laboratory environment [5]. MTurk workers are also more educated, more technical, and younger than the general population [17].

4.4 Statistical Tests

All statistical tests use a significance level of $\alpha = .05$. For each variable, we ran an omnibus test across all conditions. We ran pairwise contrasts comparing each condition to our two control conditions, *no meter* and *baseline meter*. In addition, to investigate hypotheses about the ways in which conditions varied, we ran planned contrasts comparing *tiny* to *huge*, *nudge-16* to *nudge-comp8*, *half-score* to *one-third-score*, *text-only* to *text-only half-score*, *half-score* to *text-only half-score*, and *text-only half-score* to *bold text-only half-score*. If a pairwise contrast is not noted as significant in the results section, it was not found to be statistically significant. To control for Type I error, we ran contrasts only where the omnibus test was significant. Further, we corrected contrasts for multiple testing, accounting for the previous contrasts. We applied multiple testing correction to the p-values of the omnibus tests when multiple tests were run on similar variables, such as the Likert response variables measuring user attitudes.

We analyzed quantitative data using Kruskal-Wallis for the omnibus cases and Mann-Whitney U for the pairwise cases. These tests, identified in our results as K-W and MWU, respectively, are analogues of the ANOVA and t-tests without the assumption of normality. We analyze categorical data for equality of proportions with χ^2

tests for both the omnibus and pairwise cases. All multiple testing correction used the Holm-Bonferroni method, indicated as HC throughout the paper.

4.5 Calculating Guess Numbers

We evaluated the strength of passwords created in each condition using a guess-number calculator (see Section 2.4), allowing us to approximate passwords' resistance to automated cracking. Using a password guess calculator similar to that used by Kelley et al. [18], we calculate the guessability of passwords in three different attack scenarios. This calculator simulates the password-cracking algorithm devised by Weir et al. [39], which makes guesses based on the structures, digits, symbols, and alphabetic strings in its training data. The calculator was set to only consider guesses with minimum length 8. For training, we used several "public" datasets, including leaked sets of cracked passwords. In Section 7.2, we discuss ethical issues of using leaked data.

Training data included 40 million passwords from the OpenWall Mangled Wordlist,[2] 32 million leaked passwords from the website RockYou [36], and about 47,000 passwords leaked from MySpace [27]. We augmented the training data with all strings harvested from the Google Web Corpus,[3] resulting in a dictionary of 14 million alphabetic strings.

In the *weak attacker* scenario, we consider an attacker with limited computational resources who can make 500 million (5×10^8) guesses. In the *medium attacker* scenario, we consider an attacker with greater resources who can make 50 billion (5×10^{10}) guesses. Finally, in the *strong attacker* scenario, we examine what percentage of passwords would have been guessed within the first 5 trillion (5×10^{12}) guesses. John the Ripper[4], a popular password cracker, can crack 500 million hashed passwords in about an hour on a modern desktop machine. Five trillion guesses would require a botnet of several hundred machines working for several days.

5 Results

From January to April 2012, 2,931 people completed the initial task, and 2,016 of these subjects returned for the second part of the study. We begin our evaluation by comparing characteristics of passwords created in each condition, including their length and the character classes used. Next, we simulate a cracking algorithm to evaluate what proportion of passwords in each condition would be cracked by adversaries of varying strength. We then examine the usability of these passwords, followed by data about the process of password creation. Finally, we discuss participant demographics and potential interaction effects. In Section 6, we provide additional results on participants' attitudes and reactions.

5.1 Password Characteristics

The presence of almost any password meter significantly increased password length. In conditions that scored passwords stringently, the meter also increased the use of digits, uppercase letters, and symbols. The length of the passwords varied significantly across conditions, as did the number of digits, uppercase characters, and symbols contained in each password (HC K-W, p<.001). Table 1 displays the characteristics of passwords created.

Length The presence of any password meter except *text-only* resulted in significantly longer passwords. Passwords created with *no meter* had a mean length of 10.4, and passwords created in the *text-only* condition had a mean length of 10.9, which was not significantly different. Passwords created in the thirteen other conditions with meters, with mean length ranging from 11.3 to 14.9 characters, were significantly longer than in *no meter* (HC MWU, p≤.014).

Furthermore, passwords created in *half-score*, with mean length 14.9, and in *nudge-16*, with mean length 13.0, were significantly longer than those created in *baseline meter*, which had mean length 12.0 (HC MWU, p≤.017). On the other hand, passwords created in *text-only*, with mean length 10.9, were significantly shorter than in *baseline meter* (HC MWU, p=.015). Although passwords created in *one-third-score* had mean length 14.3, they had a high standard deviation (8.1) and did not differ significantly from *baseline meter*.

Digits, Uppercase Characters, and Symbols Compared to *no meter*, passwords in five conditions contained significantly more digits: *half-score*, *one-third-score*, *nudge-comp8*, *bold text-only half-score*, and *bunny* (HC MWU, p<.028). In each of these five conditions, passwords contained a mean of 3.2 to 3.4 digits, compared to 2.4 digits in *no meter*. The mean number of digits in all other conditions ranged from 2.5 to 3.1.

In three of these conditions, *half-score*, *one-third-score*, and *bold text-only half-score*, passwords on average contained both more uppercase letters and more symbols (HC MWU, p<.019) than in *no meter*. In these three conditions, the mean number of uppercase characters ranged from 1.4 to 1.5 and the mean number of symbols ranged from 0.8 to 1.0, whereas passwords created in *no meter* contained a mean of 0.8 uppercase characters and 0.3 symbols. Furthermore, passwords created in

[2] `http://www.openwall.com/wordlists/`

[3] `http://googleresearch.blogspot.com/2006/08/all-our-n-gram-are-belong-to-you.html`

[4] `http://www.openwall.com/john/`

Table 1: A comparison across conditions of the characteristics of passwords created: the *length*, number of *digits*, number of *uppercase* letters, and number of *symbols*. For each metric, we present the mean, the standard deviation (SD), and the median. Conditions that differ significantly from *no meter* are indicated with an asterisk (*). Conditions that differ significantly from *baseline meter* are indicated with a dagger (†).

Metric	no meter (*)	baseline meter (†)	three-segment	green	tiny	huge	no suggestion	text-only	half-score	one-third-score	nudge-16	nudge-comp8	text-only half	bold text-only half	bunny
Length		*	*	*	*	*	*	†	*,†	*	*,†	*	*	*	*
Mean	10.4	12.0	11.5	11.3	11.4	11.6	11.4	10.9	14.9	14.3	13.0	11.6	12.3	13.0	11.2
SD	2.9	3.7	3.8	3.6	3.2	3.3	3.5	3.2	7.3	8.1	3.7	3.5	6.1	5.5	3.1
Median	9	11	10	10	11	11	11	10	12.5	12	12	11	10.5	11	10
Digits									*	*		*		*	*
Mean	2.4	2.7	2.8	2.6	2.7	2.5	3.0	2.5	3.3	3.4	3.2	3.3	3.1	3.2	3.3
SD	2.8	2.6	2.6	2.5	2.3	2.2	2.8	2.3	3.0	3.2	3.4	2.8	3.5	3.0	3.0
Median	2	2	2	2	3	2	2	2	3	3	3	3	2	3	3
Uppercase									*	*				*,†	
Mean	0.8	0.8	0.9	0.8	0.6	1.0	0.7	0.9	1.5	1.4	0.5	0.8	1.2	1.5	0.8
SD	2.0	1.8	1.7	2.0	1.4	2.3	1.5	1.7	3.4	3.2	1.3	1.5	2.2	2.5	1.5
Median	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.5
Symbols									*	*			*	*	
Mean	0.3	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.8	1.0	0.5	0.5	0.6	0.9	0.4
SD	0.7	1.0	0.8	1.1	0.7	0.8	0.8	0.7	1.6	2.7	1.3	1.0	1.2	1.7	0.7
Median	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

text-only half-score had significantly more symbols, 0.6 on average, than *no meter*, although the mean number of digits did not differ significantly.

While most participants used digits in their passwords, uppercase characters and symbols were not as common. In nearly all conditions, the majority of participants did not use any uppercase characters in their password despite the meter's prompts to do so. In addition, fewer than half of participants in any condition used symbols.

5.2 Password Guessability

We evaluated the strength of passwords based on their "guessability," which is the number of guesses an adversary would need to guess that password, as detailed in Section 2.4. We considered three adversaries: a *weak attacker* with limited resources who makes 500 million (5×10^8) guesses, a *medium attacker* who makes 50 billion (5×10^{10}) guesses, and a *strong attacker* who makes 5 trillion (5×10^{12}) guesses. Table 2 and Figure 3 present the proportion of passwords cracked by condition.

We found that all conditions with password meters appeared to provide a small advantage against attackers of all three strengths. In all fourteen conditions with meters, the percentage of passwords cracked by all three adversaries was always smaller than in *no meter*, although most of these differences were not statistically significant. The only substantial increases in resistance to cracking were provided by the two stringent meters with visual bars, *half-score* and *one-third-score*.

A *weak adversary* cracked 21.0% of passwords in the *no meter* condition, which was significantly larger than the 5.8% of passwords cracked in the *half-score* condition and the 4.7% of passwords cracked in *one-third-score* (HC χ^2, $p<0.001$). Furthermore, only 7.8% of passwords were cracked in *bunny*, which was also significantly less than in *no meter* (HC χ^2, $p=0.008$). Between 9.5% and 15.3% of passwords were cracked in all other conditions with meters, none of which were statistically significantly different than *no meter*.

In the *medium adversary* scenario, significantly more passwords were cracked in the *no meter* condition than in the *half-score* and *one-third-score* conditions (HC χ^2, $p \leq 0.017$). 35.4% of the passwords in the *no meter* condition were cracked, compared with 19.5% of passwords in *half-score* and 16.8% of passwords in *one-third-score*. None of the other conditions differed significantly from *no meter*; between 23.7% and 34.4% of passwords were cracked in these conditions.

The *half-score* and *one-third-score* meters were again significantly better than *no meter* against a *strong adversary*. In *no meter*, 46.7% of passwords were cracked, compared with 26.3% in *half-score* and 27.9% in *one-third-score* (HC χ^2, $p \leq 0.005$). Between 33.7% and

46.2% of passwords in all other conditions were cracked.

After the completion of the experiment, we ran additional conditions to explore how meters consisting of only a visual bar, without accompanying text, would compare to text-only conditions and conditions containing both text and visual features. Since this data was collected two months after the rest of our data, we do not include it in our main analyses. However, passwords created in these conditions performed similarly to equivalent text-only conditions and strictly worse than equivalent conditions containing both a bar and text. For instance, a strong adversary cracked 48.3% of passwords created with the *baseline meter* bar without its accompanying text and 33.0% of passwords created with the *half-score* bar without its accompanying text.

5.3 Password Memorability and Storage

To gauge the memorability of the passwords subjects created, we considered the proportion of subjects who returned for the second day of our study, the ability of participants to enter their password both minutes after creation and a few days after creation, and the number of participants who either reported or were observed storing or writing down their password.

$2,016$ of our participants, 68.8%, returned and completed the second part of the study. The proportion of participants who returned did not differ significantly across conditions (χ^2, p=0.241).

Between the 68.8% of participants who returned for the second part of the study and the 31.2% of participants who did not, there were no significant differences in the length of the passwords created, the number of digits their password contained, or the percentage of passwords cracked by a *medium* or *strong* attacker. However, the *weak* attacker cracked a significantly higher percentage of passwords created by subjects who did not return for the second part of the study than passwords created by participants who did return (HC χ^2, p<.001). 14.5% of passwords created by subjects who did not return and 9.5% of passwords created by subjects who did return were cracked. Participants who returned for the second part of the study also had more uppercase letters and more symbols in their passwords (K-W, p<.001). Participants who returned had a mean of 1.0 uppercase letters and 0.6 symbols in their passwords, while those who did not had a mean of 0.8 uppercase letters and 0.5 symbols.

Participants' ability to recall their password also did not differ significantly between conditions, either minutes after creating their password (χ^2, p=0.236) or at least two days later (χ^2, p=0.250). In each condition, 93% or more of participants were able to enter their password correctly within three attempts minutes after creating the password. When they received an email two days later to return and log in with their password, between 77% and 89% of the subjects in each condition were able to log in successfully within the first three attempts.

As an additional test of password memorability, we asked participants if they had written their password down, either electronically or on paper, or if they had stored their password in their browser. Furthermore, we captured keystroke data as they entered their password, which we examined for evidence of pasting in the password. If a participant answered affirmatively to either question or pasted the password into the password field, he or she was considered as having stored the password. Overall, 767 participants (38.0% of those who returned) reported that they had stored or written down their password. 78 of these 767 participants were also observed to have pasted in their password. An additional 32 participants (1.6%) were observed pasting in their password even thought they had said they had not stored it.

The proportion of participants storing their passwords did not differ across conditions (χ^2, p=0.364). In each condition, between 33% and 44% of participants were observed pasting in a password or reported writing down or storing their password.

5.4 Password Creation Process

Based on analysis of participants' keystrokes during password creation, we found that participants behaved differently in the presence of different password meters. Password meters seemed to encourage participants to reach milestones, such as filling the meter or no longer having a "bad" or "poor" password. The majority of participants who saw the most stringent meters changed their mind partway into password creation, erasing what they had typed and creating a different password. Table 3 presents this numerical data about password creation.

Most participants created a new password for this study, although some participants reused or modified an existing password. Between 57% and 71% of subjects in each condition (63% overall) reported creating an entirely new password, between 15% and 26% (21% overall) reported modifying an existing password, between 9% and 19% (14% overall) reported reusing an existing password, and fewer than 4% (2% overall) used some other strategy. The proportion of participants reporting each behavior did not vary significantly across conditions (χ^2, p=.876).

Participants in *nudge-16*, *bunny*, and all four stringent conditions took longer to create their password than those in *no meter* (HC χ^2, p<.001). The mean password creation time, measured from the first to the last keystroke in the password box, was 19.9 seconds in the *no meter* condition. It was 60.8 seconds for *half-score*, 59.8 seconds for *one-third-score*, 57.1 seconds

Table 2: A comparison of the percentage of passwords in each condition cracked by weak (5×10^8 guesses), medium (5×10^{10} guesses), and strong adversaries (5×10^{12} guesses). Each cell contains the percentage of passwords cracked in that threat model. Conditions that differ significantly from *no meter* are indicated with an asterisk (*).

Adversary	no meter (*)	baseline meter (†)	three-segment	green	tiny	huge	no suggestion	text-only	half-score	one-third-score	nudge-16	nudge-comp8	text-only half	bold text-only half	bunny
Weak % Cracked	21.0	11.1	10.3	12.0	10.7	9.6	11.0	15.1	* 5.8	* 4.7	15.3	10.3	9.5	11.4	* 7.8
Medium % Cracked	35.4	27.2	26.6	30.0	30.0	31.0	25.9	34.4	* 19.5	* 16.8	25.0	23.7	24.2	25.7	28.1
Strong % Cracked	46.7	39.4	39.4	45.5	42.1	41.6	39.3	46.2	* 26.3	* 27.9	33.7	39.2	34.7	35.6	40.1

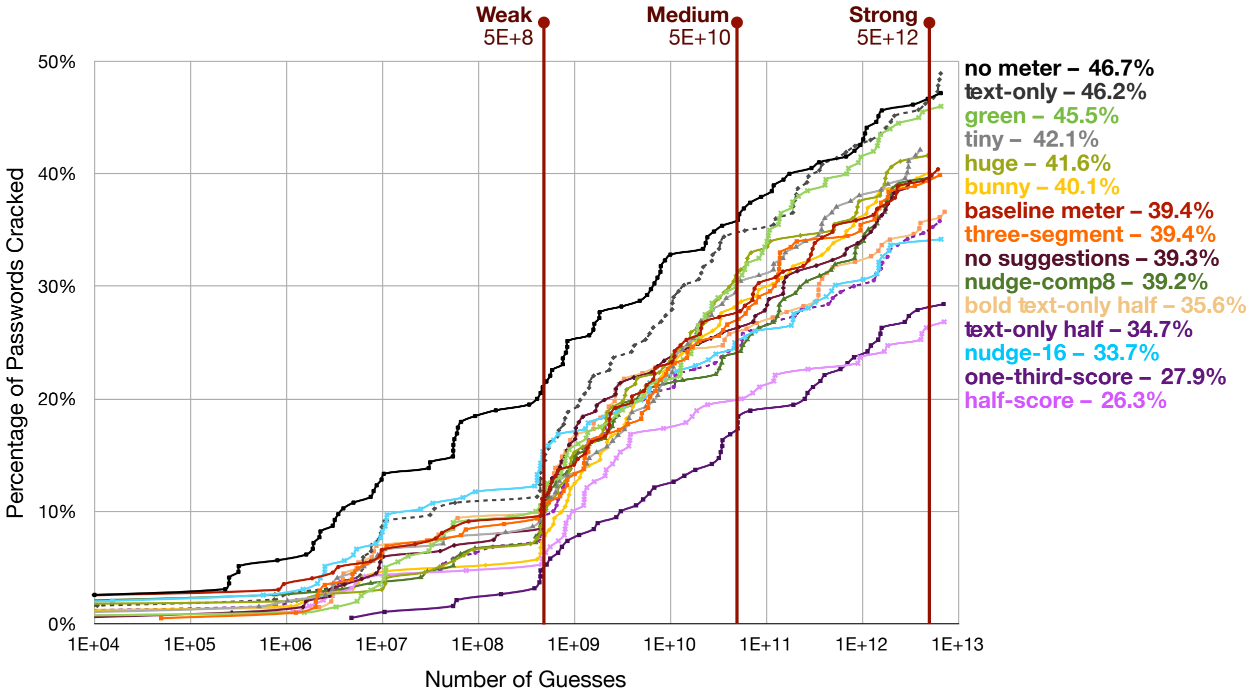


Figure 3: This graph contrasts the percentage of passwords that were cracked in each condition. The x-axis, which is logarithmically scaled, indicates the number of guesses made by an adversary, as described in Section 2.4. The y-axis indicates the percentage of passwords in that condition cracked by that particular guess number.

for *bold text-only half-score*, 38.5 seconds for *text-only half-score*, 33.1 seconds for *nudge-16*, and 30.4 seconds for *bunny*. Compared also to the *baseline meter* meter, where mean password creation time was 23.5 seconds, participants took significantly longer in the *half-score*, *one-third-score*, and *bold text-only half-score* conditions (HC χ^2, p<.008). The mean time of password creation ranged from 21.0 to 26.6 seconds in all other conditions.

Password meters encouraged participants both to avoid passwords that the meter rated "bad" or "poor" and to create passwords that filled the meter. Had there been a password meter, 24.1% of passwords created in *no meter* would have scored "bad" or "poor," which was significantly higher than the 12.0% or fewer of passwords in all non-stringent conditions other than *no suggestions* and *nudge-16* rated "bad" or "poor" (HC χ^2, p$\leq$0.035). Had *no meter* contained a password meter, 25.1% of passwords created would have filled the meter. A larger proportion of passwords in all non-stringent conditions other than *no suggestions* and *nudge-16* filled the meter (HC

Table 3: A comparison across conditions of password creation: the percentage of participants who completely *filled the password meter* or equivalently scored "excellent" in text-only conditions, the percentage of participants whose password received a score of *"bad" or "poor"*, the *time* of password creation (first to last keystroke), the number of *deletions* (characters deleted after being entered) in the password creation process, the percentage of participants who *changed their password* (initially entering a valid password containing at least 8 characters before completely deleting it and entering a different password), and the *edit distance* between the initial password entered and the final password saved, normalized by the length of the final password. Conditions differing significantly from *no meter* are indicated with an asterisk (*), while those differing significantly from *baseline meter* are marked with a dagger (†).

Metric	no meter (*)	baseline meter (†)	three-segment	green	tiny	huge	no suggestion	text-only	half-score	one-third-score	nudge-16	nudge-comp8	text-only half	bold text-only half	bunny
Filled Meter		*	*	*	*	*		*	*,†	*,†	†	*	*,†	*,†	*
% of participants	(25.1)	48.5	53.2	42.5	48.2	52.8	37.3	46.2	9.0	1.6	24.5	46.9	3.2	5.0	48.4
"Bad" or "Poor"		*	*	*	*	*	*		*,†	*,†	†	*	*,†	*,†	*
% of participants	(24.1)	9.1	10.3	12.0	9.6	8.1	7.5	13.4	58.4	93.7	37.2	9.8	76.3	67.8	8.3
Time (seconds)									*,†	*,†	*		*	*,†	*
Mean	19.9	23.5	22.7	21.0	21.5	25.8	24.7	24.8	60.8	59.8	33.1	26.6	38.5	57.1	30.4
SD	28.4	22.7	23.6	22.2	23.2	28.9	36.6	29.4	75.7	84.9	33.2	30.2	49.8	150.0	36.9
Median	10.6	15.6	14.0	13.7	13.1	14.7	13.0	14.0	39.1	34.2	23.2	13.8	23.5	32.8	19.8
Deletions									*,†	*,†	*,†		*,†	*,†	*
Mean	5.3	6.2	7.5	5.8	6.2	7.8	5.5	7.8	23.8	22.9	12.1	8.1	14.6	23.1	10.7
SD	10.7	10.2	13.7	12.4	10.8	11.3	8.4	11.9	29.0	26.6	16.2	13.3	19.3	26.9	17.2
Median	0	0	0	0	1	2	0	0	13.5	13	8	1	8	13.5	5
Changed PW									*,†	*,†	*,†		*,†	*,†	*,†
% of participants	14.4	18.7	25.6	16.5	23.9	23.4	25.9	25.8	52.6	52.6	40.3	24.7	35.8	51.0	34.9
Norm. Edit Dist.									*,†	*,†	*,†		*,†	*,†	*,†
Mean	0.10	0.09	0.47	0.09	0.14	0.12	0.15	0.17	0.37	0.45	0.27	0.15	0.27	0.35	0.28
SD	0.29	0.23	4.84	0.28	0.30	0.31	0.37	0.36	0.42	1.22	0.38	0.36	0.43	0.47	0.70
Median	0	0	0	0	0	0	0	0	0.15	0.11	0	0	0	0.08	0

χ^2, $p \leq 0.006$). In each of these conditions, 42.5% or more of the passwords filled the meter. While the proportion of passwords in *nudge-16* and the four stringent conditions reaching these thresholds was significantly lower than *baseline meter*, the proportions would have been higher than *baseline meter* were the *baseline meter* scoring algorithm used in those conditions.

During the password creation process, participants in all four stringent conditions, as well as in *nudge-16*, made more changes to their password than in *no meter* or *baseline meter*. We considered the number of deletions a participant made, which we defined as the number of characters that were inserted into the password and then later deleted. In the four stringent conditions and in *nudge-16*, the mean number of deletions by each participant ranged from 12.1 to 23.8 characters. In contrast, significantly fewer deletions were made in *no meter*, with a mean of 5.3 deletions, and *baseline meter*, with a mean of 6.2 deletions (HC MWU, $p<0.001$). The *bunny* condition, with a mean of 10.7, also had significantly more deletions than *no meter* (HC MWU, $p=0.004$).

We further analyzed the proportion of participants who changed their password, finding significantly more changes occurring in the stringent conditions, as well as in *nudge-16* and *bunny*. Some participants entered a password containing eight or more characters, meeting the stated requirements, and then completely erased the password creation box to start over. We define the *initial password* to be the longest such password containing eight or more characters that a participant created before starting over. Similarly, we define the *final password* to be the password the participant eventually saved. We considered participants to have changed their password if they created an initial password, completely erased the password field, and saved a final password that differed by one edit or more from their initial password.

More than half of the participants in *half-score*, *one-third-score*, and *bold text-only half-score* changed their password during creation. Similarly, between 34.9% and 40.3% of *nudge-16*, *text-only half-score*, and *bunny* participants changed their password. The proportion of participants in these six conditions who changed their password was greater than the 14.4% of *no meter* participants and 18.7% of *baseline meter* participants who did so (HC χ^2, p≤.010). Across all conditions, only 7.7% of final passwords consisted of the initial password with additional characters added to the end; in a particular condition, this percentage never exceeded 16%.

These changes in the password participants were creating resulted in final passwords that differed considerably from the initial password. We assigned an edit distance of 0 to all participants who did not change their password. For all other participants, we computed the Levenshtein distance between the initial and final password, normalized by the length of the final password. The mean normalized edit distance between initial and final passwords ranged from 0.27 to 0.45 in the six aforementioned conditions, significantly greater than *no meter*, with a mean of 0.10, and *baseline meter*, with a mean of 0.09 (HC MWU, p<.003).

We also compared the guessability of the initial and final passwords for participants whose initial password, final password, or both were guessed by the strong adversary. 86.1% of the 43 such changes in *half-score* resulted in a password that would take longer to guess, as did 83.8% of 37 such changes in *text-only half-score*. In contrast, 50% of 18 such changes in *baseline meter* and between 56.7% and 76.7% such changes in all other conditions resulted in passwords that would take longer to guess. However, these differences were not statistically significant.

5.5 Participant Demographics

Participants ranged in age from 18 to 74 years old, and 63% percent reported being male and 37% female.[5] 40% percent reported majoring in or having a degree or job in computer science, computer engineering, information technology, or a related field; 55% said they did not. Participants lived in 96 different countries, with most from India (42%) and the United States (32%). Because many of our password meters used a color scheme that includes red and green, we asked about color-blindness; 3% of participants reported being red-green color-blind, while 92% said they were not, consistent with the general population [30].

The number of subjects in each condition ranged from 184 to 202, since conditions were not reassigned if a participant did not complete the study. There were no statistically significant differences in the distribution of participants' gender, age, technology background, or country of residence across experimental conditions.

However, participants who lived in different countries created different types of passwords. We separated participants into three groups based on location: United States, India, and "the rest of the world." Indian subjects' passwords had mean length 12.2, U.S. subjects' passwords had mean length 11.9, and all other subjects' passwords had mean length 12.1 (HC K-W, p=0.002). Furthermore, Indian subjects' passwords had a mean of 0.9 uppercase letters, and both U.S. subjects' and all other subjects' passwords had a mean of 1.0 uppercase letters (HC K-W, p<0.001). While the percentage of passwords cracked by a *weak* or *medium* attacker did not differ significantly between the three groups, a lower percentage of the passwords created by Indian participants than those created by American participants was cracked by a *strong* adversary (HC χ^2, p=.032). 42.3% of passwords created by subjects from the U.S., 35.5% of passwords created by subjects from India, and 38.8% of passwords created by subjects from neither country were cracked by a *strong* adversary. However, the guessing algorithm was trained on sets of leaked passwords from sites based in the U.S., which may have biased its guesses.

6 Participants' Attitudes and Perceptions

We asked participants to rate their agreement on a Likert scale with fourteen statements about the password creation process, such as whether it was fun or annoying, as well as their beliefs about the password meter they saw. We also asked participants to respond to an open-ended prompt about how the password meter did or did not help. We begin by reporting participants' survey responses, which reveal annoyance among participants in the stringent conditions. The *one-third-score* condition and text-only stringent conditions also led participants to believe the meter gave an incorrect score and to place less importance on the meter's rating. The distribution of responses to select survey questions is shown in Figure 4. We then present participants' open-ended responses, which illuminate strategies for receiving high scores from the meter.

6.1 Attitudes Toward Password Meters

In a survey immediately following password creation, a higher percentage of participants in the stringent conditions found password creation to be annoying or difficult than those in *baseline meter*. A larger proportion of subjects in the four stringent conditions than in either the *no meter* or *baseline meter* conditions agreed that creating a password in this study was annoying (HC χ^2, p≤.022). Similarly, a higher percentage of subjects in the *half-score* and *bold text-only half-score* found creating a password difficult than in either the *no meter* or *baseline meter* conditions (HC χ^2, p≤.012). Creating a password was also considered difficult by a higher percentage of

[5] We offered the option not to answer demographic questions; when percentages sum to less than 100, non-answers make up the remainder.

subjects in *one-third-score* and *text-only half-score* than in *baseline meter* (HC χ^2, p≤.003), although these conditions did not differ significantly from *no meter*.

Participants in the stringent conditions also found the password meter itself to be annoying at a higher rate. A higher percentage of subjects in all four stringent conditions than in *baseline meter* agreed that the password-strength meter was annoying (HC χ^2, p≤.007). Between 27% and 40% of participants in the four stringent conditions, compared with 13% of *baseline meter* participants, found the meter annoying.

Participants in the two stringent conditions without a visual bar felt that they did not understand how the meter rated their password. 38% of *text-only half-score* and 39% of *bold text-only half-score* participants agreed with the statement, "I do not understand how the password strength meter rates my password," which was significantly greater than the 22% of participants in *baseline meter* who felt similarly (HC χ^2, p≤.015). 32% of *half-score* participants and 34% of *one-third-score* participants also agreed, although these conditions were not statistically significantly different than *baseline meter*.

The *one-third-score* condition and both text-only stringent conditions led participants to place less importance on the meter. A smaller proportion of *one-third-score*, *text-only half-score*, and *bold text-only half-score* participants than *baseline meter* subjects agreed, "It's important to me that the password-strength meter gives my password a high score" (HC χ^2, p≤.021). 72% of *baseline meter* participants, yet only between 49% and 56% of participants in those three conditions, agreed. In all other conditions, between 64% and 78% of participants agreed. Among these conditions was *half-score*, in which 68% of participants agreed, significantly more than in *one-third-score* (HC χ^2, p=.005).

More participants in those same three conditions felt the meter's score was incorrect. 42-47% of *one-third-score*, *text-only half-score*, and *bold text-only half-score* participants felt the meter gave their password an incorrect score, significantly more than the 21% of *baseline meter* participants who felt similarly (HC χ^2, p≤.001). Between 12% and 33% of participants in all other conditions, including *half-score*, agreed; these conditions did not differ significantly from *baseline meter*.

6.2 Participant Motivations

Participants' open-ended responses to the prompt, "Please explain how the password strength meter helped you create a better password, or explain why it was not helpful," allowed some participants to explain their thought process in reaction to the meter, while others discussed their impressions of what makes a good password.

6.2.1 Reactions to the Password Meter

Some participants noted that they changed their behavior in response to the meter, most commonly adding a different character class to the end of the password. One participant said the meter "motivated [him] to use symbols," while another "just started adding numbers and letters to the end of it until the high score was reached." Participants also said that the meter encouraged or reminded them to use a more secure password. One representative participant explained, "It kept me from being lazy when creating my password. [I] probably would not have capitalized any letters if not for the meter."

Other participants chose a password before seeing the meter, yet expressed comfort in receiving validation. For instance, one representative participant noted, "The password I ultimately used was decided on before hand. However, whilst I was typing and I saw the strength of my password increase and in turn felt reassured."

However, a substantial minority of participants explained that they ignore password meters, often because they believe these meters discourage passwords they can remember. One representative participant said, "No matter what the meter says, I will just use the password I chose because it's the password I can remember. I do not want to get a high score for the meter and in the end have to lose or change my password." Some participants expressed frustration with meters for not understanding this behavior. For instance, one participant explained, "I have certain passwords that I use because I can remember them easily. I hate when the meter says my password is not good enough– it's good enough for me!"

Participants also reported embarrassment at poor scores, fear of the consequences of having a weak password, or simply a desire to succeed at all tasks. One participant who exemplifies the final approach said, "I wanted to make my password better than just 'fair,' so I began to add more numbers until the password-strength meter displayed that my password was 'good.' I wanted to create a strong password because I'm a highly competitive perfectionist who enjoys positive feedback." In contrast, another participant stated, "Seeing a password strength meter telling me my password is weak is scary."

6.2.2 Impressions of Password Strength

Participants noted impressions of password strength that were often based on past experiences. However, the stringent conditions seemed to violate their expectations.

Most commonly, subjects identified a password containing different character classes as strong. One representative participant said, "I am pretty familiar with password strength meters, so I knew that creating a password with at least 1 number/symbol and a mixture of upper

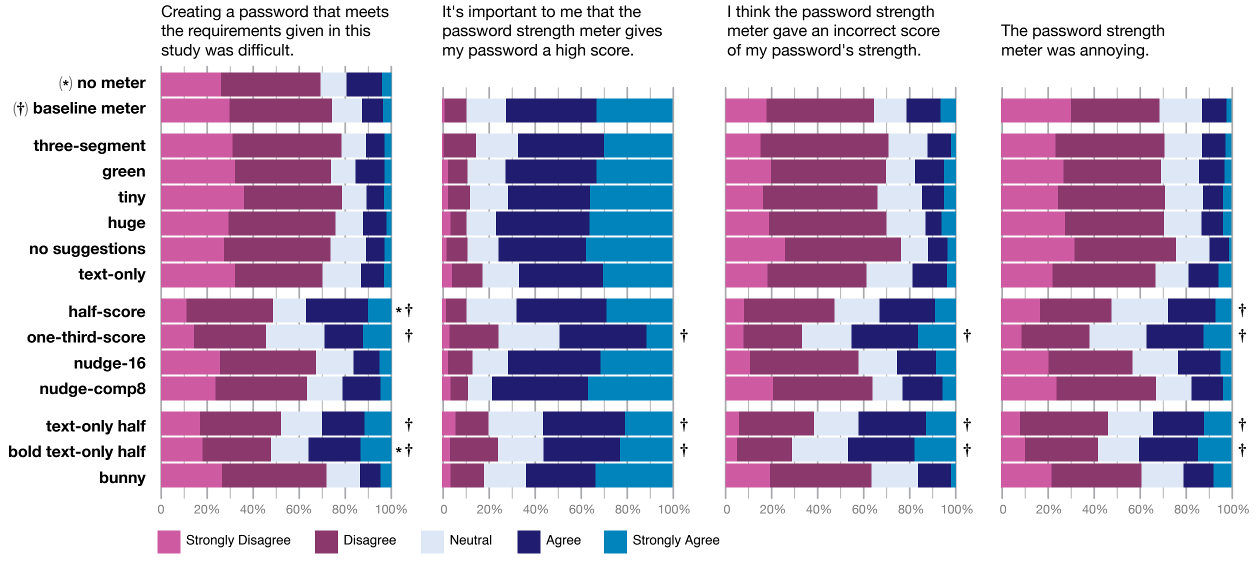


Figure 4: These charts depict participants' agreement or disagreement with the statement above each chart. Each color represents the proportion of participants in that condition who expressed a particular level of agreement of disagreement with the statement. Conditions in which the proportion of participants agreeing with a statement differed significantly from *no meter* are indicated with an asterisk (*), while those that differed significantly from *baseline meter* are marked with a dagger (†). Participants in *no meter* did not respond to questions about password meters.

and lower case letters would be considered strong." Participants also had expectations for the detailed algorithm with which passwords were scored, as exemplified by a participant who thought the meter "includes only English words as predictable; I could have used the Croatian for 'password123' if I wanted."

The stringent conditions elicited complaints from participants who disagreed with the meter. For example, one participant was unsure how to receive a good score, saying, "No matter what I typed, i.e. how long or what characters, it still told me it was poor or fair." Another participant lamented, "Nothing was good enough for it!" Some participants questioned the veracity of the stringent meters. For instance, a *one-third-score* participant said, "I have numbers, upper/lower case, and several symbols. It's 13 characters long. It still said it was poor. No way that it's poor." Other participants reused passwords that had received high scores from meters in the wild, noting surprise at the stringent meters' low scores. Some participants became frustrated, including one who said the *one-third-score* meter "was extremely annoying and made me want to punch my computer."

The *bunny* received mixed feedback from participants. Some respondents thought that it sufficed as a feedback mechanism for passwords. For instance, one subject said, "I think it was just as helpful as any other method I have seen for judging a password's strength...I do think the dancing bunny is much more light-hearted and fun." However, other participants found the more traditional bar to be more appropriate, including one who said *bunny* "was annoying, I am not five [years old]."

6.2.3 Goals for the Password Meter

Participants stated two primary goals they adopted while using the password meter. Some participants aimed to fill the bar, while others hoped simply to reach a point the meter considered not to be poor. Those participants who aimed to fill the bar noted that they continued to modify their password until the bar was full, citing as motivation the validation of having completed their goal or their belief that a full bar indicated high security.

Participants employing the latter strategy increased the complexity of their password until the text "poor" disappeared. One participant noted, "It gave me a fair score, so I went ahead with the password, but if it would have given me a low score I would not have used this password." A number of participants noted that they didn't want to receive a poor rating. One representative participant said, "I didn't want to have poor strength, while I didn't feel I needed something crazy."

Some participants also identified the bar's color as a factor in determining when a password was good enough. Some participants hoped to reach a green color, while others simply wanted the display not to be red. One participant aiming towards a green color said, "I already chose a fairly long password, but I changed a letter in it to an uppercase one to make it turn green." Another participant expressed, "I knew that I didn't want to be in the red, but being in the yellow I thought was ok."

7 Discussion

We discuss our major findings relating to the design of effective password meters. We also address our study's ethical considerations, limitations, and future work.

7.1 Effective Password Meters

At a high level, we found that users do change their behavior in the presence of a password-strength meter. Seeing a password meter, even one consisting of a dancing bunny, led users to create passwords that were longer. Although the differences were generally not statistically significant, passwords created in all 14 conditions with password meters were cracked at a lower rate by adversarial models of different strengths.

However, the most substantial changes in user behavior were elicited by stringent meters. These meters led users to add additional character classes and make their password longer, leading to significantly increased resistance to a guessing attack. Furthermore, more users who saw stringent meters changed the password they were creating, erasing a valid password they had typed and replacing it with one that was usually harder to crack.

Unfortunately, the scoring systems of meters we observed in the wild were most similar to our non-stringent meters. This result suggests that meters currently in use on popular websites are not aggressive enough in encouraging users to create strong passwords. However, if all meters a user encountered were stringent, he or she might habituate to receiving low scores and ignore the meter, negating any potential security benefits.

There seems to be a limit to the stringency that a user will tolerate. In particular, the *one-third-score* meter seemed to push users too hard; *one-third-score* participants found the meter important at a lower rate and thought the meter to be incorrect at a higher rate, yet their passwords were comparable in complexity and cracking-resistance to those made by *half-score* participants. Were meters too stringent, users might just give up.

Tweaks to the password meter's visual display did not lead to significant differences in password composition or user sentiment. Whether the meter was tiny, monochromatic, or a dancing bunny did not seem to matter. However, an important factor seemed to be the combination of text and a visual indicator, rather than only having text or only having a visual bar. Conditions containing text without visual indicators, run as part of our experiment, and conditions containing a visual bar without text, run subsequently to the experiment we focus on here, were cracked at a higher rate and led to less favorable user sentiment than conditions containing a combination of text and a visual indicator.

In the presence of password-strength meters, participants changed the way they created a password. For instance, the majority of participants in the stringent conditions changed their password during creation. Meters seemed to encourage participants to create a password that filled the meter. If that goal seemed impossible, participants seemed content to avoid passwords that were rated "bad" or "poor." In essence, the password meter functions as a progress meter, and participants' behavior echoed prior results on the effects progress meters had on survey completion [8]. Meters whose estimates of password strength mirrored participants' expectations seemed to encourage the creation of secure passwords, whereas very stringent meters whose scores diverged from expectations led to less favorable user sentiment and an increased likelihood that a participant would abandon the task of creating a strong password.

We also found many users to have beliefs regarding how to compose a strong password, such as including different character classes. Because users' understanding of password strength appears at least partially based on experience with real-world password-strength meters and password-composition policies, our results suggest that wide-scale deployment of more stringent meters may train users to create stronger passwords routinely.

7.2 Ethical Considerations

We calculated our guessability results by training a guess-number calculator on sets of passwords that are publicly and widely available, but that were originally gathered through illegal cracking and phishing attacks. It can be argued that data acquired illegally should not be used at all by researchers, and so we want to address the ethical implications of our work. We use the passwords alone, excluding usernames and email addresses. We neither further propagate the data, nor does our work call significantly greater attention to the data sets, which have been used in several scientific studies [4, 9, 18, 38, 39]. As a result, we believe our work causes no additional harm to the victims, while offering potential benefits to researchers and system administrators.

7.3 Limitations

One potential limitation of our study is its ecological validity. Subjects created passwords for an online study, and they were not actually protecting anything valuable with those passwords. Furthermore, one of the primary motivations for part of the MTurk population is financial compensation [17], which differs from real-world motivations for password creation. Outside of a study, users would create passwords on web pages with the logos and insignia of companies they might trust, perhaps making them more likely to heed a password meter's suggestions. On the other hand, subjects who realize they are participating in a password study may be more likely to think carefully about their passwords and pay closer attention to the password meter than they otherwise would. We did ask participants to imagine that they were creating passwords for their real email accounts, which prior work has shown to result in stronger passwords [21]. Because our results are based on comparing passwords between conditions, we believe our findings about how meters compare to one another can be applied outside our study.

Our study used a password-cracking algorithm developed by Weir et al. [39] in a guess-number calculator implemented by Kelley et al. [18] to determine a password's guessability. We did not experiment with a wide variety of cracking algorithms since prior work [18, 38, 42] has found that this algorithm outperformed alternatives including John the Ripper. Nevertheless, the relative resistance to cracking of the passwords we collected may differ depending on the choice of cracking algorithm.

Furthermore, the data we used to train our cracking algorithm was not optimized to crack passwords of particular provenance. For instance, passwords created by participants from India were the most difficult to crack. The data with which we trained our guessing algorithm was not optimized for participants creating passwords in languages other than English, which may have led to fewer of these passwords being cracked; prior work by Kelley et al. [18] found that the training set has a substantial effect on the success of the guessing algorithm we used.

7.4 Future Work

Further research in password-strength meters may involve continued examination of the structure and composition of passwords created with meters. The presence of a meter caused changes in users' behavior, with over 50% of participants in three of the four stringent meter conditions erasing a valid 8-character password they had already entered and entering a new, different password. The strategies users employed both initially and after this shift deserve further investigation, both to suggest directions for user feedback and to uncover patterns that can improve techniques for cracking passwords.

In addition, we have certainly not exhausted the space of possible password-strength meters. Although we have found that the score conveyed to the user is a more important factor than the visual display, it is possible that either subtle or substantial variations to the scoring algorithm (e.g., representing a password's likelihood [7]) or to the textual feedback provided to users may increase the usability and security of the resulting passwords. Furthermore, there seems to be a limit to how stringent a meter can be. Alternate scoring algorithms, improved text feedback, and the degree of stringency that leads to the best tradeoff between usability and security for passwords thus appear to be fertile ground for future work.

8 Conclusion

We have conducted the first large-scale study of password-strength meters, finding that meters did affect user behavior and security. Meters led users to create longer passwords. However, unless the meter scored passwords stringently, the resulting passwords were only marginally more resistant to password cracking attacks.

Meters that rated passwords stringently led users to make significantly longer passwords that included more digits, symbols, and uppercase letters. These passwords were not observed to be less memorable or usable, yet they were cracked at a lower rate by simulated adversaries making 500 million, 50 billion, and 5 trillion guesses. The most stringent meter annoyed users, yet did not provide security benefits beyond those provided by slightly less stringent meters. The combination of a visual indicator and text outperformed either in isolation. However, the visual indicator's appearance did not appear to have a substantial impact.

Despite the added strength that these more stringent meters convey, we observed many more lenient meters deployed in practice. Our findings suggest that, so long as they are not overly onerous, employing more rigorous meters would increase security.

9 Acknowledgments

This research was supported by NSF grants DGE-0903659 and CNS-1116776, by CyLab at Carnegie Mellon under grants DAAD19-02-1-0389 and W911NF-09-1-0273 from the Army Research Office, by Air Force Research Lab Award No. FA87501220139, and by a gift from Microsoft Research.

References

[1] ADAMS, A., SASSE, M. A., AND LUNT, P. Making passwords secure and usable. In *Proc. HCI on People and Computers XII* (1997).

[2] BISHOP, M., AND KLEIN, D. V. Improving system security via proactive password checking. *Computers & Security 14*, 3 (1995), 233–249.

[3] BONNEAU, J. The science of guessing: Analyzing an anonymized corpus of 70 million passwords. In *Proc. IEEE Symposium on Security and Privacy* (2012).

[4] Bonneau, J., Just, M., and Matthews, G. What's in a name? Evaluating statistical attacks on personal knowledge questions. In *Proc. Financial Crypto* (2010).

[5] Buhrmester, M., Kwang, T., and Gosling, S. D. Amazon's Mechanical Turk: A new source of inexpensive, yet high-quality, data? *Perspectives on Psychological Science 6*, 1 (2011), 3–5.

[6] Burr, W. E., Dodson, D. F., and Polk, W. T. Electronic authentication guideline. Tech. rep., NIST, 2006.

[7] Castelluccia, C., Dürmuth, M., and Perito, D. Adaptive password-strength meters from Markov models. In *Proc. NDSS* (2012).

[8] Conrad, F. G., Couper, M. P., Tourangeau, R., and Peytchev, A. The impact of progress indicators on task completion. *Interacting with computers 22*, 5 (2010), 417–427.

[9] Dell'Amico, M., Michiardi, P., and Roudier, Y. Password strength: An empirical analysis. In *Proc. INFOCOM* (2010).

[10] Downs, J. S., Holbrook, M. B., Sheng, S., and Cranor, L. F. Are your participants gaming the system? Screening Mechanical Turk workers. In *Proc. CHI* (2010).

[11] Few, S. *Information Dashboard Design: The Effective Visual Communication of Data.* O'Reilly Media, Inc., 2006.

[12] Florêncio, D., and Herley, C. A large-scale study of web password habits. In *Proc. WWW* (2007).

[13] Forget, A., Chiasson, S., van Oorschot, P., and Biddle, R. Improving text passwords through persuasion. In *Proc. SOUPS* (2008).

[14] Herley, C. So long, and no thanks for the externalities: The rational rejection of security advice by users. In *Proc. NSPW* (2009).

[15] Herley, C., and Van Oorschot, P. A research agenda acknowledging the persistence of passwords. *IEEE Security & Privacy*, 99 (2011).

[16] Inglesant, P., and Sasse, M. A. The true cost of unusable password policies: Password use in the wild. In *Proc. CHI* (2010).

[17] Ipeirotis, P. G. Demographics of Mechanical Turk. Tech. Rep. CeDER-10-01, New York University, March 2010.

[18] Kelley, P. G., Komanduri, S., Mazurek, M. L., Shay, R., Vidas, T., Bauer, L., Christin, N., Cranor, L. F., , and Lopez, J. Guess again (and again and again): Measuring password strength by simulating password-cracking algorithms. In *Proc. IEEE Symposium on Security and Privacy* (2012).

[19] Kessler, D. A., Mande, J. R., Scarbrough, F. E., Schapiro, R., and Feiden, K. Developing the "nutrition facts" food label. *Harvard Health Policy Review 4*, 2 (2003), 13–24.

[20] Kittur, A., Chi, E. H., and Suh, B. Crowdsourcing user studies with Mechanical Turk. In *Proc. CHI* (2008).

[21] Komanduri, S., Shay, R., Kelley, P. G., Mazurek, M. L., Bauer, L., Christin, N., Cranor, L. F., and Egelman, S. Of passwords and people: Measuring the effect of password-composition policies. In *Proc. CHI* (2011).

[22] Kotadia, M. Gates predicts death of the password, Feb. 2004. `http://news.cnet.com/2100-1029-5164733.html`.

[23] Leyden, J. Office workers give away passwords for a cheap pen, Apr. 2003. `http://www.theregister.co.uk/2003/04/18/office_workers_give_away_passwords/`.

[24] Loewenstein, G. F., and Haisley, E. C. The economist as therapist: Methodological ramifications of 'light' paternalism. In *The Foundations of Positive and Normative Economics*. Oxford University Press, 2008.

[25] Milman, D. A. Death to passwords, Dec. 2010. `http://blogs.computerworld.com/17543/death_to_passwords`.

[26] Proctor, R. W., Lien, M.-C., Vu, K.-P. L., Schultz, E. E., and Salvendy, G. Improving computer security for authentication of users: Influence of proactive password restrictions. *Behavior Research Methods, Instruments, & Computers 34*, 2 (2002), 163–169.

[27] Schneier, B. Myspace passwords aren't so dumb, Dec. 2006. `http://www.wired.com/politics/security/commentary/securitymatters/2006/12/72300`.

[28] Shay, R., Kelley, P. G., Komanduri, S., Mazurek, M. L., Ur, B., Vidas, T., Bauer, L., Christin, N., and Cranor, L. F. Correct horse battery staple: Exploring the usability of system-assigned passphrases. In *Proc. SOUPS* (2012).

[29] Shay, R., Komanduri, S., Kelley, P. G., Leon, P. G., Mazurek, M. L., Bauer, L., Christin, N., and Cranor, L. F. Encountering stronger password requirements: User attitudes and behaviors. In *Proc. SOUPS* (2010).

[30] Shevell, S. K., Ed. *The Science of Color*. Elsevier, 2003.

[31] Sotirakopoulos, A., Muslukov, I., Beznosov, K., Herley, C., and Egelman, S. Motivating users to choose better passwords through peer pressure. In *Proc. SOUPS (Poster Abstract)* (2011).

[32] Stanton, J. M., Stam, K. R., Mastrangelo, P., and Jolton, J. Analysis of end user security behaviors. *Comp. & Security 24*, 2 (2005), 124–133.

[33] Summers, W. C., and Bosworth, E. Password policy: The good, the bad, and the ugly. In *Proc. WISICT* (2004).

[34] Thaler, R., and Sunstein, C. *Nudge: Improving decisions about health, wealth, and happiness*. Yale University Press, 2008.

[35] Toomim, M., Kriplean, T., Pörtner, C., and Landay, J. Utility of human-computer interactions: Toward a science of preference measurement. In *Proc. CHI* (2011).

[36] Vance, A. If your password is 123456, just make it hackme. New York Times (New York edition), Jan. 21, 2010.

[37] Vu, K.-P. L., Proctor, R. W., Bhargav-Spantzel, A., Tai, B.-L. B., and Cook, J. Improving password security and memorability to protect personal and organizational information. *Int. J. of Human-Comp. Studies 65*, 8 (2007), 744–757.

[38] Weir, M., Aggarwal, S., Collins, M., and Stern, H. Testing metrics for password creation policies by attacking large sets of revealed passwords. In *Proc. CCS* (2010).

[39] Weir, M., Aggarwal, S., de Medeiros, B., and Glodek, B. Password cracking using probabilistic context-free grammars. In *Proc. IEEE Symposium on Security and Privacy* (2009).

[40] Wogalter, M., and Vigilante, Jr., W. Effects of label format on knowledge acquisition and perceived readability by younger and older adults. *Ergonomics 46*, 4 (2003), 327–344.

[41] Yan, J. J. A note on proactive password checking. In *Proc. NSPW* (2001).

[42] Zhang, Y., Monrose, F., and Reiter, M. K. The security of modern password expiration: An algorithmic framework and empirical analysis. In *Proc. CCS* (2010).