Usability and Security of Text Passwords on Mobile Devices

William Melicher, Darya Kurilova,* Sean M. Segreti, Pranshu Kalvani, Richard Shay, Blase Ur, Lujo Bauer, Nicolas Christin, Lorrie Faith Cranor, Michelle L. Mazurek†

Carnegie Mellon University †University of Maryland
{billy, ssegreti, pranshu, rshay, bur, lbauer, nicolasc, lorrie}@cmu.edu
*darya@cs.cmu.edu †mmazurek@umd.edu

ABSTRACT

Recent research has improved our understanding of how to create strong, memorable text passwords. However, this research has generally been in the context of desktops and laptops, while users are increasingly creating and entering passwords on mobile devices. In this paper we study whether recent password guidance carries over to the mobile setting. We compare the strength and usability of passwords created and used on mobile devices with those created and used on desktops and laptops, while varying password policy requirements and input methods. We find that creating passwords on mobile devices takes significantly longer and is more error prone and frustrating. Passwords created on mobile devices are also weaker, but only against attackers who can make more than 10^{13} guesses. We find that the effects of password policies differ between the desktop and mobile environments, and suggest ways to ease password entry for mobile users.

Author Keywords

Passwords; Password-composition policies; Usable Security; Security policy; Mobile devices; Authentication

ACM Classification Keywords

H.1.2. Human factors: User/Machine Systems; K.6.5. Authentication: Security and Protection

INTRODUCTION

Extensive prior work has examined how password-composition policies affect password strength and usability on laptops and desktops (e.g., [39, 45, 48]). However, in the last decade, the electronic-device landscape has significantly changed. Mobile devices, such as smartphones and tablets, have surged in popularity and are now used not only for calling and sending text messages, but also for email, web surfing, social networking, and banking [18]. Most of these activities require authentication, typically in the form of text passwords. Unfortunately, due to the size of virtual keys and the effort required to navigate between keyboard pages, text entry on mobile devices is time consuming and error prone [29, 37, 42]. Conventional wisdom and prior work suggests that this problem has important effects on the usability and security of text passwords on mobile devices [24, 51].

At the same time, choosing difficult-to-guess passwords remains important. Offline attacks, in which attackers steal databases of hashed passwords and then "crack" these passwords by making up to trillions of guesses, are common (e.g., [3, 5, 21, 36]). These attacks can be particularly damaging because users often reuse passwords, sometimes with small modifications, across different accounts [20]. Offline attacks also continue to be successful despite efforts to use slow hashes to limit attackers' ability to crack stolen passwords [23].

As text passwords have been studied primarily in the context of laptops and desktops, the precise impact of mobile device usability on the security of passwords created and used on mobile devices with respect to offline guessing attacks remains largely unknown.

In this paper, we compare the usability and security of text passwords created or used on phones or tablets (referred to as *mobile devices*) to those created and used on laptop or desktop computers (referred to as *traditional devices*) in the context of offline guessing attacks. We then revisit recent recommendations for password-composition policies and study whether these recommendations hold for mobile devices. Finally, we investigate whether some of the usability drawbacks of password entry on mobile devices can be mitigated by improved password-entry methods.

To compare password behavior on mobile and traditional devices, we conducted a 2,709-participant, between-subjects, online study. We investigated several recently recommended password-composition policies, as well as different password-input methods on mobile devices. We measured usability via self-reported user sentiment and observed behavior (e.g., entry time), and password strength using simulated offline-guessing attacks.

Prior research speculated that passwords created on mobile devices could be more vulnerable to guessing attacks than those created on traditional devices [50]. Consistently with this, we found that passwords created on mobile devices had fewer uppercase and special characters, with special characters more often grouped together. Perhaps surprisingly, we observed no difference between the strength of passwords created on mobile and traditional devices against attackers making up to 10^{13} guesses. We did find, however, that against

CHI'16, May 07-12, 2016, San Jose, CA, USA
ACM 978-1-4503-3362-7/16/05.
http://dx.doi.org/10.1145/2858036.2858384

stronger attackers making up to 10^{16} guesses passwords created on mobile devices were significantly weaker (by 32%) than those created on traditional devices.

In contrast, the results of our usability evaluation of passwords on mobile devices matched expectations: participants on mobile devices took 20% longer to enter passwords and failed to create passwords due to typing mistakes twice as often, resulting in increased user frustration compared to participants who used traditional devices. Participants also reported both creating and using passwords on mobile devices regularly, despite being frustrated by these usability issues.

We also explored how password usability and security on mobile devices are affected by different input methods. These included displaying passwords in plain text and allowing automatic text-entry tools, including word completion, spell check, and Swype, a faster text entry system for mobile devices. Making passwords visible improved both their usability and, unexpectedly, their resistance to offline guessing (while also making them potentially more vulnerable to shoulder-surfing and screen-capture attacks). Contrary to our expectations, the addition of automatic text-entry tools, in combination with password visibility, did not appear to provide an additional usability improvement.

We found that password-composition policy recommendations for traditional devices need adjustment to apply to mobile devices. While the relative strengths of passwords created under each password policy appear similar across device types, the mobile environment affects the relative usability of password policies differently. Recent work found that two specific policies—which require passwords to be at least 12 or 16 characters long and comply with other requirements—offer a promising balance between usability and security [44]. We find the usability gap between these two policies to be notably narrower on mobile devices than on traditional devices. This bolsters the case for requiring passwords to be at least 16 characters long (while limiting other requirements), leading us to amend previous policy recommendations for situations when passwords may be created or used on mobile devices.

BACKGROUND AND RELATED WORK

In this section, we review prior work on password entry and creation for mobile and traditional devices, including previous work studying password policies on traditional devices. We then review different text-entry methods for passwords and alternative authentication methods on mobile devices.

Text Passwords on Mobile Devices

In the work most closely related to ours, Zezschwitz et al. found that passwords created on mobile devices are shorter and contain fewer symbols and uppercase letters [50]. They studied how and why mobile devices make it more difficult to enter passwords and examined how this affects security based on password composition and manual inspection of passwords. In contrast, our work measures the difference in security between passwords created on mobile and traditional devices by the more accurate method of simulating guessing attacks. In addition, we examine the effect of different password policies on the usability of password creation on mobile devices and explore input methods and cross-device aspects that could influence password security and usability.

Other studies have found that entering text using touchscreen devices affects typing and results in passwords with significantly lower entropy [26,54]. Maydebura et al. found that participants had more difficulty typing passwords on their mobile devices while walking, which is a common use case for mobile passwords [38]. Bao et al. found that users who primarily work with traditional devices type passwords more slowly on mobile devices and that this may influence password security [2]. Greene et al. evaluated the usability of complex, randomly generated passwords on mobile devices and found that entering the password on mobile devices increases the time to enter the password and the number of errors during password entry [25]. Consequently, Greene et al. proposed a system that permutes characters in randomly generated passwords to minimize switching between different keyboard layouts [24]. They note that this would decrease entropy, and recommend restoring entropy by adding more characters. Schaub et al. assessed the usability of various mobile keyboards [42]. This included assessing changes in entry time and error rate caused by entering passwords on mobile devices. They found that the alternate function keys on virtual keyboards may increase the time to enter passwords while also increasing the risk of a shoulder-surfing attack.

Password Policies

A variety of work has explored the effects of different password composition policies on the strength and usability of passwords [34,44]; research suggests that more stringent policies can result in users fulfilling these requirements in predictable ways [9, 28, 34, 45]. Other research found that policies that emphasize length, rather than character complexity, have both security and usability benefits over policies that have a shorter length requirement with more character complexity [31, 34]. However, these previous studies were based on passwords created on traditional devices, such as laptops and desktops, and it is unclear whether these findings still hold on mobile devices. We follow a similar methodology to these studies for collecting and analyzing passwords, but we investigate the extent to which the constraints of mobile devices impact the security of password-composition policies. The previously studied usability drawbacks of mobile devices suggest that the ideal password policy on traditional devices may not be the same as on mobile devices.

Recent work questions the need to study offline attacks, since the combination of well-implemented systems security and using slow hash functions to protect passwords should, in principle, make offline attacks unlikely and, even if they succeed, make passwords difficult for attackers to recover [19]. In practice, however, successful offline attacks are common [3,5,21,36], and implementing better password-composition policies could significantly lessen the impact of such attacks.

Password Entry Methods

Previous work has explored using different text-entry methods to promote better passwords. Jakobsson and Akavipat suggested the use of "fastwords," in which multi-word

passphrases with auto-correction and auto-completion are used to make password entry faster on mobile and full-sized keyboards. They found that this method made users on mobile devices chose passwords that are more secure, in addition to making password entry more usable, when participants were given instruction during creation [29]. We also study how features such as auto-correction and auto-completion can improve passwords on mobile devices, but do not compare passwords to the "fastwords" system. Komanduri et al. studied how making passwords visible during creation impacted their security and usability, and found that plain text visibility was not a significant factor contributing to password usability and strength [33]. However, that work primarily explored using visibility in the context of giving users data-driven feedback about password strength on traditional devices and did not explore the effect that visibility might have on password creation. In addition, many commercial systems, such as Windows 8, Android, and iOS, have already adopted the practice of enabling passwords to be optionally or partially visible during their user input. While password entry on mobile devices may be susceptible to shoulder surfing attacks [42], it is still useful to explore whether the usability benefits of visibility reduce users' frustration resulting in stronger passwords.

Text Password Alternatives on Mobile Devices

A number alternatives to text passwords have been considered for authentication on mobile devices, such as graphical patterns [51], fingerprint and face unlock mechanisms [4], recognition-based graphical passwords [16], authentication mechanisms that use the front and the back of mobile devices [13], and authentication mechanisms that use all of these approaches [43]. Password managers have the potential to be useful on mobile devices but have not yet proved to be usable and are distrusted by users [10, 30]. Although all these mechanisms are promising, especially for unlocking phones [27], they are unlikely to replace all uses of text passwords, particularly for services that are accessed from both mobile and traditional devices. Therefore, quantifying and examining how best to tune the usability and security of text passwords on mobile devices remains an important topic.

METHODOLOGY

To examine password creation and use on mobile and traditional devices, we conducted a two-part online study using Amazon's Mechanical Turk crowdsourcing service.

Online Study

In the first part of the study, participants created a password on either a mobile (e.g., a phone or tablet) or traditional (e.g., a laptop or desktop) device, under varying password policies and using varying input methods. Later, participants were asked to return for the second part in which participants reentered the password. At each step, participants took a short survey about the process. All participants were at least 18 years old and located in the United States. Participants were recruited for a two-part study and paid $1.00 for completing Part One and a $1.50 bonus for returning a few days later and to complete Part Two. The study protocol was approved by the Carnegie Mellon University Institutional Review Board.

Part One

Participants were presented with a hypothetical scenario in which their main email service provider had been attacked and their account was compromised. They were further asked to imagine that their email provider is requiring them to change their password. To proceed to the main task, participants were required to prove they had a mobile device, such as a smartphone or tablet. All participants were required to complete this step, regardless of whether they were assigned to a condition that required the use of a mobile device, in order to make the participants' demographics as well as the participants' experiences across conditions more homogeneous. We verified possession of a mobile device by checking the participant's browser HTTP User-Agent string; participants were allowed any number of tries. Until a mobile device was verified, participants could not proceed to the rest of the study.

After verification, participants were assigned to one of eleven conditions in a round-robin fashion. Depending on the condition, participants either continued the survey on a mobile device or were asked to switch to a traditional device. We verified that participants used the correct device using the browser's User-Agent string. Participants were then asked to create a password adhering to the password-composition requirements of their condition. After creating a password, participants reentered it for confirmation. There were no differences between the interfaces that participants saw in the mobile and non-mobile conditions beyond varying textual description of password requirements. In conditions with different input methods, the interface was the same as in other conditions, except for how the textual input was shown and treated in the text fields for password entry. Participants then took a survey collecting demographic information and information about password creation. Afterwards, participants were again prompted to enter their newly created password to encourage them to remember it.

Part Two

Two days after completing Part One, participants received an email with a link to continue the study for a bonus payment. After following the link, participants were asked to enter the password they created in Part One. Depending on the condition, the use of a mobile device was again verified via the HTTP User-Agent string. Participants who could not remember their password could select "I forgot my password" to receive a recovery email. After five incorrect attempts the correct password was displayed. After password entry, participants completed a final survey collecting information about the process as well as general password behaviors.

Conditions

Study conditions differed in three dimensions: 1) the *password policy* with which the created password must comply, 2) whether *text-entry features* were enabled, and 3) the *type of device* used for password creation in Part One and for password re-entry in Part Two.

Password Policy

Each participant created a password under one of the five following password-composition policies. These policies were

chosen for comparison with previous work on password policies for traditional devices [28, 44]. Our 8-character policy is similar to one of the main password policies proposed by NIST in 2006 [9]. Our choice to study longer policies was motivated by recent work that evaluated a range of such policies [44]. That work found two password policies requiring many characters, the 12-character and 16-character policies used in our experiment, to be more usable and more secure than the widely used 8-character policy.

basic20 Passwords must be at least 20 characters long.

3class8 Passwords must be at least 8 characters long and contain at least three character classes (lowercase letters, uppercase letters, digits, and special characters).

3class12 Passwords must be at least 12 characters long and contain at least three character classes.

2word16 Passwords must be at least 16 characters long and contain at least two words (blocks of lowercase or uppercase letters) separated by a non-letter sequence.

3word20 Passwords must be at least 20 characters long and contain at least three words (blocks of lowercase or uppercase letters) separated by a non-letter sequence.

Input Methods

Along with traditional password input—a textbox that displays characters as dots and does not allow automatic text-entry tools—we considered two variations:

Visible Participants saw their password as they typed it, whereas traditionally passwords are obfuscated. We hypothesized that making passwords visible could make it easier to notice typos. In this condition, passwords were always visible during creation and re-entry.

Autotools Participants were able to see their password and to use automatic text-entry tools, such as Swype, autocorrection, autocapitalization, autocomplete, and spell check. We hypothesized that automatic text-entry tools allow entering passwords more quickly and accurately [29]. While assignment to this condition was experimentally controlled, participants may or may not have used these tools during the experiment. We chose to not force users to take advantage of these tools in part because that is not technically feasible in a remote field study, and so we felt that voluntary use would more accurately model a real situation.

Type of Devices

The type of device participants used was varied during both password creation and re-entry. Participants were told at password-creation which type of device they would use for re-entry, simulating a scenario in which users know how they will use the password later and can balance security and usability accordingly. This is a best-case scenario; in practice, users creating a password may not think about using it on other devices later. We consider: **Mobile to mobile (MM)**, **Traditional to mobile (TM)**, and **Traditional to Traditional (TT)**. While a scenario in which users create passwords on mobile devices and re-enter them on traditional devices is interesting, we prioritized the above scenarios, which we believe are more likely in practice.

Tested Conditions

It was not feasible to explore each possible combination in our condition matrix in a full factorial design. We chose 11 conditions that we deemed most relevant to answering our research questions. Condition names (e.g., "2word16-visible-MM") are the combination of the policy short-hand, additional features if applicable, and device type.

3class8MM, **3class8TT**, **3class12MM**, **3class12TT**, **2word16MM**, **2word16TT**. These three condition pairs allow us to directly compare passwords created on traditional devices with those created on mobile devices.

basic20MM, **3word20MM**. Together with the previous MM conditions, these allow us to additionally investigate the effects of length and complexity in the mobile setting.

2word16TM. In combination with 2word16MM and 2word16TT, this allows us to further examine the effect of changing the type of device used for creating and entering passwords. We focus on the 2word16 policy because it was recently recommended for traditional devices [44].

2word16-autotools-MM, **2word16-visible-MM**. In combination with 2word16MM and 2word16TT, these allow us to further investigate differences in password usability and security caused by varying input methods. We chose 2word16 as the password policy for this investigation because we hypothesized that, as a word-based condition, it would most benefit from automatic text-entry tools.

Measuring Usability

We measured a variety of observed (objective) and perceived (subjective) usability metrics for each condition.

Time to create. The time elapsed between loading the password-creation web page and submitting a password. This was measured on the participant's machine to avoid measuring network latency. Times greater than two standard deviations above the mean (suggesting the participant was away from the device) were not included.

Creation attempts, re-entry attempts, and reasons for failures. The number of attempts the participant needed to create a conforming password, the number of attempts the participant needed to enter the correct password during Part Two, and which composition requirements the participant did not meet during failed creation attempts.

Difficulty of password entry. Self-reported agreement with the statement, "I found it difficult to enter the password I created on this device" on a five-point Likert scale.

Copy-pasting. Whether participants pasted passwords in the browser's password-entry field during re-entry in Part Two.

Number of deletions. The number of characters participants deleted during the password-creation process.

Password storage. Self-reported incidence of storing password (e.g., writing down or saving in the browser).

Measuring Password Strength

We used guessability—how many guesses a particular cracking algorithm takes to guess a password—to measure the

strength of the collected passwords. Recent work advocates using this metric, which simulates an attacker trying guesses based on expected probability, rather than other metrics such as entropy [6, 7, 31, 52]. In this metric, each password is assigned a guess number describing how many guesses an attacker would need to guess the password. Moreover, we follow recent work by using multiple attack methods to simulated a skilled attacker [11, 14, 41, 49]. Different attacks require different resources, leading to variation in feasible guessing cutoffs. We analyze the results separately to be able to attribute different results at the two cutoffs to either the effect of different attacks or effect of the number of guesses. There is no guarantee that this measurement of strength simulates all attackers; indeed, an attacker with more knowledge about how these passwords were created may be able to make better guesses.

Probabilistic Context-Free Grammar Attack

One component of our approach is an algorihm developed by Weir at al., which uses a training set of existing passwords to generate a probabilistic context-free grammar (*PCFG*) to guess passwords in order of their predicted probability [53]. We used a modified version of Weir's technique, which can accommodate long passwords by tokenizing training and guessing data along string boundaries [44]. For each condition, we generated at least 10^{13} guesses with this attack.

oclHashcat Attack

To augment the PCFG attack and emulate a common, real-world attacker with access to multiple cracking methods, we ran the popular GPU-based cracking software oclHashcat [46]. oclHashcat was chosen instead of other off-the-shelf cracking tools because of its popularity in industry and password-cracking competitions [22, 35, 40], and with actual attackers. Its high-speed GPU implementation makes the computation of high guess numbers more feasible than is possible using the PCFG algorithm. Our configuration was based on similar work studying, in part, how to effectively configure different cracking methods [49]. oclHashcat can be configured to guess passwords in a number of ways; we configured it to perform a mangled dictionary attack (making 10^{13} guesses to match the PCFG guess number) and a combinator attack (making 6.8×10^{15} guesses), which we found to be the most effective in our preliminary experiments. This is more than the PCFG attack because of the technical infeasibility of generating large numbers of guesses using PCFG.

Statistical Tests

We used pairwise, regression, and omnibus tests: Omnibus tests were used to determine when pairwise tests were appropriate. Pairwise tests were used when comparing two conditions. Regression tests were used to estimate the relationship between several variables, such as the device type used, the password policy, and additional password input methods.

For analyzing the guessability of passwords, we followed recent passwords research [39] in using Cox regression, a survival-analysis technique that compares hazard distributions that are associated with time [12]. This technique accounts both for the guess numbers assigned to cracked passwords and for those passwords that remained uncracked after our attacks, allowing us to include all available guessing data in our statistical comparisons. To prevent overfitting in our regression model, we used Akaike Information Criterion (AIC)—a measure of the quality of a statistical model—in combination with standard backward elimination to remove extraneous interaction terms [1]. We remove one interaction factor at a time from the model to minimize the AIC.

For omnibus comparisons, we used Kruskal-Wallis tests for quantitative data and Chi-squared tests for categorical data. For pairwise comparisons, we used Mann-Whitney U tests for quantitative data and Chi-squared tests for categorical data. We use non-parametric statistical tests to avoid making assumptions about the probability distributions of our data. To compare mobile (MM) and traditional (TT) conditions, we used linear regression (abbreviated Lin.R) for quantitative data and logistic regressions (abbreviated Log.R) for categorical data. In these regressions, we included data for 3class8MM, 3class8TT, 3class12MM, 3class12TT, 2word16MM, and 2word16TT conditions. The 2word16TM condition was excluded from this analysis because participants in that condition had to use both types of devices and were notified about that upfront, which may have influenced their choices of passwords. Regressions predicted dependent variables based on three binary variables: whether the password was made in a mobile condition, 3class8 condition or 2word16 condition. All tests used a significance level of $\alpha = 0.05$. To control for type I error, we applied Holm-Bonferroni correction.

Ecological Validity and Limitations

Although analyzing real-world passwords would be ideal, it is infeasible to collect such passwords and associated usability data under controlled conditions. The use of MTurk for high-quality human-subjects data has been validated both generally [8, 15, 32, 47] and specifically for passwords studies [17, 39]. Specifically, Mazurek et al. found that passwords from realistic online password studies, including studies employing similar methodology to ours, can approximate those used for real-life, high-value accounts [39]. Additionally, the biasing effects sometimes attributed to the use of simulated data apply to all participants and are likely to have similar impact across conditions. In comparison to other available resources (e.g., password leaks), simulated password studies provide the best source of experimentally controlled data when real data is not available. Therefore, while not ideal, the use of simulated data is a practical alternative to the use of real data and is useful to guide design of password systems.

Our study had several other limitations, many of which are common to online passwords studies. Our data set is comprised exclusively of participants from the United States, and our results may not generalize to other populations. To distinguish between traditional and mobile devices, we used user-agent strings sent to us by the participants' web browsers. However, modern web browsers allow spoofing the user-agent string; it is possible some participants used this feature to participate in the study using devices that did not match their assigned condition. All of these limitations should have similar impacts across all conditions, allowing for effective

comparisons. However, in addition to these limitations, participants were potentially able to choose a convenient time and a comfortable environment in which to complete study tasks, so our results may represent a best-case scenario for usability and security. This is especially relevant for users on mobile devices who may see exacerbated usability issues in reality due to non-ideal situations, for example, entering a password while walking.

RESULTS

We next compare passwords in different conditions using metrics for strength and usability, and report on the perceptions and behaviors of participants in all conditions. We report results from both Part One and Part Two independently. For Part One measurements we include all participants who completed Part One; for Part Two measurements we include all participants who completed Part Two.

Participant and Demographics

A total of 2,709 participants completed Part One of our study. Of those, 2,025 participants (75%) returned for Part Two. The difference in the return rates for Part Two among conditions was not statistically significant. Participants' ages ranged from 18 to 72 years with a median of 26, and 50% of participants reported as male and 49% as female (A "Prefer not to answer" option was provided). Our participants were required to be from the United States, and 21% reported having a degree or job in a computer-science-related field. Each condition had between 232 and 273 participants during Part One. The vast majority of mobile devices were either Android phones (49%) or iPhone (41%) devices.

Usage of Mobile Devices

Participants reported general difficulties using passwords on mobile devices: 52% of participants reported that they "avoid entering passwords on mobile devices," 23% reported that they have changed a password due to difficulties entering it on a mobile device in the past, and 20% have passwords that they use specifically on mobile devices. However, despite difficulties with using passwords on mobile devices, participants reported that they both create and enter passwords on mobile devices fairly often. 82% of participants reported creating more than one password on a mobile device and 37% reported creating more than three passwords on a mobile device. 75% of participants report entering passwords on mobile devices more than a few times every week, with some entering passwords more often. In addition, 55% of participants reported using their mobile device for banking, which provides evidence that these passwords can be of high value.

Security Results

To evaluate password strength, we calculated guess numbers for two kinds of guessing attacks: a Hashcat attack and a PCFG attack. Guessability results are shown in Figures 1 and 2. On each graph, the x-axis shows guess numbers and the y-axis shows the percent of passwords correctly guessed after a given number of guesses. Curves closer to the x-axis represent stronger password sets, with a higher resistance to guessing attacks. The summary of results of our Cox regression, after backward elimination, is shown in Table 1. These results show that the composition policy and the password-input method each have significant effects.

Our analysis shows that passwords created on mobile devices are neither weaker nor stronger than those created on desktop devices for attackers attempting 10^{13} guesses in both the Hashcat and PCFG attacks. However, when considering results up to the maximum number of guesses that the Hashcat attack made, 6.8×10^{15}, mobile passwords were 32% more likely (Cox regression, $p < 0.024$) than traditional passwords to be guessed. Mobile-to-traditional comparison plots are shown in Figures 1a and 2a. The effect on password strength of using a mobile device was more pronounced for the 3class12 and 2word16 policies; for the 3class8 policy, creating and entering on a mobile device had a limited effect on strength. We hypothesize that the impact of using a mobile device is greater for longer password-composition policies.

The password-composition policies we tested were chosen in part because they were found, on traditional devices, to offer different levels of resistance to guessing (e.g., [44]). The resistance to guessing, per policy, of the passwords created on mobile devices is shown in Figure 1b and 2b. Overall, we found that the relative ordering of policies by strength is similar to the ordering on traditional devices, both as measured in prior work [44] and by our conditions for traditional devices. At the cutoff of our guessing attacks, 2word16, basic20, and 3word20 were the strongest policies. Of the policies previously recommended for use in practice, 2word16 was consistently stronger than 3class12 and 3class12 was consistently stronger than 3class8.

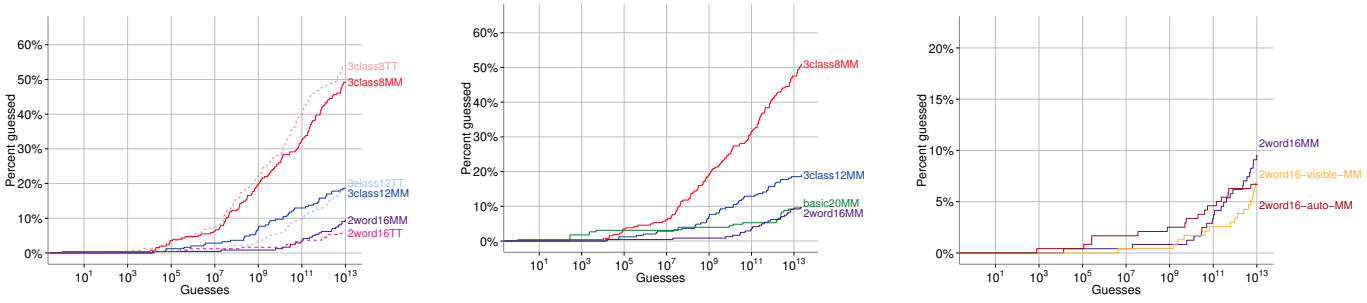
When examining the impact of the password-input method on security, we found, unexpectedly, that making passwords visible during creation (2word16-visible-MM) resulted in significantly stronger passwords across both types of simulated attackers compared to passwords that were not visible (2word16MM). Depending on the type of attack, visible passwords were from 2.71 to 2.61 times less likely to be guessed than passwords that are not visible during creation (Cox regression, $p < 0.001$). Adding additional text-entry tools, on the other hand, provided no additional benefit beyond that provided by visibility.

Usability Results

Table 2 summarizes the quantitative results. We discuss these results in the context of four dimensions: (1) differences between mobile and traditional passwords, (2) among password policies on mobile devices, (3) password input methods, and (4) scenarios in which participants enter passwords created on desktop devices on a mobile device.

Mobile and Traditional Passwords

To compare passwords on mobile devices and passwords on traditional devices, we considered participants in the 3class8MM, 3class8TT, 3class12MM, 3class12TT, 2word16MM, and 2word16TT conditions. The 2word16TM condition was excluded from this analysis because participants in that condition had to use both types of devices and were notified about that upfront, which may have influenced their choices of passwords.

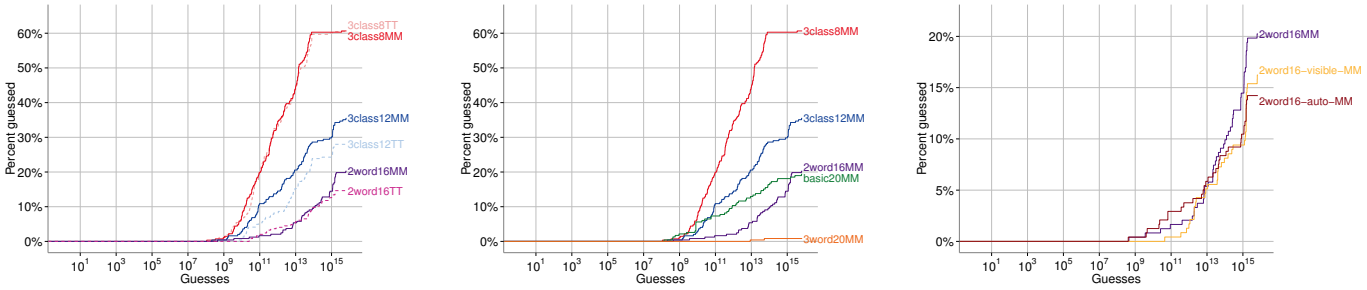


(a) Mobile and traditional condition pairs.

(b) Mobile conditions. Note that PCFG did not make any successful guesses against 3word20.

(c) Conditions that differ in input methods.

Figure 1: Guessability of passwords across conditions using a PCFG attack.



(a) Mobile and traditional condition pairs.

(b) Mobile conditions.

(c) Conditions that differ in input methods.

Figure 2: Guessability of passwords across conditions using a Hashcat attack.

Overall, our results suggest that password creation and entry is less usable on mobile devices than on traditional devices. Participants spent more time creating passwords on mobile devices than on traditional devices (MM: 18.8s, TT: 23.4s, Lin.R, $F(3, 1501) = 22.4$, $p < 0.001$), and participants entering passwords on mobile devices failed more often in the password creation task in Part One (MM: 1.77 attempts, TT: 1.54 attemps, Lin.R, $F(3, 1506) = 50.82$, $p < 0.001$). Participants on mobile devices spent less time entering their passwords on the first attempt during re-entry (MM: 13.0, TT: 15.3s, Lin.R, $F(3, 992) = 16.9$, $p < 0.001$), but made about the same number of re-entry attempts (MM: 1.73, TT: 1.71, Lin.R, $F(3, 1012) = 1.232$, $p < 0.30$). During creation, failure to create a password in mobile conditions was more often due to a mismatched verification password (TT: 6% vs. MM: 12% of attempts) suggesting that participants that used mobile devices may have mistyped their password more often. In addition, participants in mobile conditions reported more difficulty entering their passwords at a significantly higher rate than those in traditional conditions (MM: 24%, TT: 15%, Log.R, $p < 0.001$).

Other results likely reflect general difficulty with text entry on mobile devices. Fewer participants on mobile devices than on traditional devices copy-pasted their passwords (MM: 1.64%, TT: 8.71%, Log.R, $p < 0.001$), and participants in mobile conditions deleted fewer characters during password creation (MM: 3.25 deletions, TT: 5.54 deletions, Lin.R, $F(3, 1506) = 21.44$, $p < 0.001$). Participants in mobile conditions also stored their passwords less often (MM: 25.1%, TT: 30.7%, Log.R, $p < 0.03$).

We found no statistically significant differences between mobile and traditional passwords with respect to annoyance with password creation, reported difficulty in remembering passwords, or time spent in password re-entry.

Visibility and Password Input Methods

To examine the effect of visibility and automatic text-entry tools (e.g., Swype and iOS's autocorrect feature), we compared results from the 2word16MM, 2word16-visible-MM, and 2word16-auto-MM conditions.

Making passwords visible improved usability on two of the objective usability metrics, but had no difference on participant sentiment. The visible condition required fewer deletions during creation (2word16MM: 4.79 deletions, 2word16-visible-MM: 2.62 deletions, Kruskal-Wallis, $H(1) = 6.28$, $p < 0.012$) and fewer creation attempts (2word16MM: 2.10 attempts, 2word16-visible-MM: 1.76 attempts, Kruskal-Wallis, $H(1) = 11.85$, $p < 0.001$).

Only a fraction of participants in the 2word16-auto-MM condition used automatic text-entry tools. Analyzing keypress logging information and comparing it to the content of the password-entry field, we detected that only 35 of 239 participants (14.6%) in this condition used an automatic text-entry tool. Some participants used mobile devices which did not support them, and others did not choose to use these features.

Factor	PCFG up to 10^{13} guesses	Hashcat up to 10^{13} guesses	Hashcat up to 6.8×10^{15} guesses
Device			
Mobile is	1.05× stronger than traditional	1.04× weaker than traditional	**1.32× weaker than traditional**
Policy			
3class8 is	**3.59× weaker than 3class12**	**2.68× weaker than 3class12**	**2.93× weaker than 3class12**
2word16 is	**2.40× stronger than 3class12**	**2.04× stronger than 3class12**	**2.17× stronger than 3class12**
basic20 is	**1.88× stronger than 3class12**	1.34× stronger than 3class12	**2.04× stronger than 3class12**
3word20 is	0× likely to guess as 3class12	**23.1× stronger than 3class12**	**38.5× stronger than 3class12**
Input methods			
Autotool is	same strength as visible	0.86× weaker than visible	1.10× stronger than visible
Visible is	**2.71× stronger than obfuscated**	**2.62× stronger than obfuscated**	**2.66× stronger than obfuscated**

Table 1: Summary of results of Cox regression by factors. Statistically significant findings are in bold, and their cells are colored: red (darker) cells indicate cases when the factor is stronger than the baseline and blue (lighter) cells indicate cases when the factor is weaker than the baseline.

We tested for usability differences between the 2word16-auto-MM and 2word16-visible-MM conditions; and also among 2word16-auto-MM participants based on whether they did or did not use automatic text-entry tools. In neither case, however, were the differences statistically significant. The largest observed differences were for annoyance and deleting a space during password creation: More 2word16-auto-MM participants who used the tools found password creation annoying than those who did not use the automatic text-entry tools (did not use autotools 56%, used autotools 74%). More participants who used the tools also deleted a space during password creation than those in the same condition (did not use autotools 8%, used autotools 23%).

Traditional Creation, Mobile Entry

The 2word16TM condition was designed to examine creation on a traditional device in combination with entry on a mobile device. We were expecting that participants who created their password on a traditional device might find it more difficult to enter that password on a mobile device than passwords created on mobile devices.

We compared Part One metrics between 2word16TM and 2word16TT, and Part Two metrics between 2word16TM and 2word16MM. Neither comparison showed any statistically significant difference, although password creation was slightly slower and involved more deletions in 2word16TM than in 2word16TT (see Table 2).

Comparing Previously Recommended Policies

Recent work that focused on traditional devices recommended the 3class12 and 2word16 policies, with 3class12 being slightly more usable during creation and 2word16 offering more security [44]. Here we compare the relative performance of those two policies on traditional devices with their relative performance on mobile devices.

In terms of security, our findings roughly matched those of the previous work [44]: in our study, under all types of attacks, passwords created under the 2word16 policy were about two times stronger than those created under the 3class12 policy according to the Cox regression. The regression did not show additional significant effects on this difference between 2word16 and 3class12 based on whether passwords were created on a mobile or a traditional device.

In terms of usability on traditional devices, our results were also similar to the findings of the previous work in that 2word16 is less usable than 3class12 on some (but not all) metrics [44]. We found 3class12TT to perform better than 2word16TT in the time and number of attempts needed to create a password, the reported difficulty of creating a password, and the number of deletions during password creation.

On mobile devices, however, 3class12 loses this advantage, with no significant difference between the two conditions in either failure to create or reported difficulty, both of which previously favored 3class12. In fact, 3class12MM significantly outperformed 2word16MM on only two usability metrics: time to create passwords (23.3s to 29.3s, Kruskal-Wallis, $H(1) = 10.3$, $p < 0.003$) and the number of deletions during creation (2.36 to 4.79, Kruskal-Wallis, $H(1) = 8.00$, $p < 0.03$). This result shows that the usability gap between 3class12 and 2word16 is narrower on mobile devices.

Syntactic Properties of Passwords

Prior work found syntactic features of passwords correlated with resistance to guessing, e.g., using more character classes and spreading them throughout a password is correlated with stronger passwords [39]. We examined the syntactic properties of passwords we collected, including length, number of character classes used, and the number and distribution throughout the password of characters from different character classes. If a password had characters from a particular character class, we categorized their location as 'beginning,' 'end,' 'middle,' or 'spread.' Beginning, end, and middle mean that all characters from that class were clustered at the beginning, in the middle, or at the end of the password. Spread means that there are at least two clusters of characters from that class separated by at least one character of a different type. For example, the password "P@ssword!!1" is classified as uppercase beginning, lowercase middle, digits end, and special characters spread. We define *bunching* as any

Condition Name	Creation Time (s)	Creation Attempts	Deletions	Entry Time (s)	Re-entry Attempts	Stored (%)	Pasted (%)
3class8TT	12.5	1.20	3.25	13.7	1.73	25.6	3.94
3class8MM	18.3	1.43	2.70	10.3	1.55	20.6	1.03
3class12TT	20.5	1.50	5.79	14.4	1.77	33.0	10.0
3class12MM	23.3‡	1.81	2.36‡	13.7	1.93	27.8	1.14
2word16TT	23.8	1.92	7.70	18.0	1.68	33.5	12.4
2word16TM	28.0	1.90	9.57	18.7	1.65	37.0	0
2word16MM	29.3‡	2.10*	4.79*‡	15.3	1.66	27.4	2.79
2word16-autotools-MM	31.2	1.97	4.54	11.3	1.45	33.7	0.59
2word16-visible-MM	33.9	1.76*	2.62*	16.3	1.45	29.9	1.72
basic20MM	31.8	1.84	6.38	17.8	1.80	30.4	1.75
3word20MM	44.4	2.38	7.95	21.8	1.82	41.2	2.35
aggregateTT	18.8†	1.54†	5.54†	15.3	1.73	30.7†	8.71†
aggregateMM	23.4†	1.77†	3.25†	13.0	1.71	25.1†	1.64†

Table 2: Password usability metrics by condition. "Creation Time" refers to the time it took participants to create a password, given in seconds. "Creation Attempts" represents the mean of the number of attempts it took participants to create a password. "Deletions" represents of the number of characters participants deleted when creating a password. "Re-entry attempts" represent the number of attempts participants made to enter their password in Part Two. "Stored" refers to the percent of participants who recorded their password on paper or electronically. "Pasted" refers to the percent of participants who copy-pasted their passwords using a software clipboard or browser autofill. † means that that metric is significantly different between mobile and traditional devices. ‡ means that that metric is significantly different between 2word16MM and 3class12MM. * means that that factor is statistically different between 2word16MM and 2word16-visible-MM. aggregateTT is a combination of the data from 3class8TT, 3class12TT, and 2word16TT; aggregateMM is a combination of the data from 3class8MM, 3class12MM, and 2word16MM.

classification other than spread. For example, the password "Pa$$word" has bunched special characters.

Using this analysis, we found that participants in mobile conditions tended to bunch the special characters in their passwords more often than those in traditional conditions (MM: 91%, TT: 87%, Log.R, $p < 0.004$). Consistent with prior work [50], passwords in mobile conditions tended to have fewer uppercase (MM: 1.64 chars, TT: 1.89 chars, Lin.R, $F(3, 1506) = 20.38$, $p < 0.02$), special characters (MM: 0.88 chars, TT: 1.00 chars, Lin.R, $F(3, 1506) = 8.79$, $p < 0.03$), and digits (MM: 3.15, TT: 2.93, Lin.R F(3, 1506) = 20.85, $p < 0.042$) than those in traditional conditions. These findings provide some explanation for why that work found passwords in the mobile conditions to be weaker; as well as why our analysis finds mobile passwords to be weaker than traditional passwords against a strong attacker. At the same time, the differences, even when statistically significant, are small, which is consistent with our finding that the difference in strength between traditional and mobile passwords for attackers who made up to 10^{13} guesses is not statistically significant.

DISCUSSION

We next discuss our findings and their implications for the design of password systems for mobile devices.

Impaired Usability and Security on Mobile Devices

Entering passwords on mobile devices presents many usability challenges. Typing special characters is particularly burdensome, as users must change the keyboard 'depth' to access special characters; typing uppercase letters and digits incurs similar additional effort. This extra effort can be observed in our results: we found that passwords created on mobile devices contained fewer uppercase letters and fewer special characters. We also observed more bunching of special characters on mobile devices, perhaps because users do not want to access the special character keyboard more than once.

Also consistent with the expectation that password entry on mobile devices is more difficult than on traditional devices was our finding that users make more mistakes when creating and entering their passwords on mobile devices. We also saw reduced incidence of coping mechanisms that might ameliorate usability problems, such as password storage and copy-pasting, perhaps because these operations can be more difficult on mobile devices. Many people may not know how to use the copy-paste functionality on mobile devices, and it can be more difficult to electronically save passwords on mobile devices than on traditional devices.

These usability limitations can lead to a decrease in security, as passwords with fewer non-lowercase characters that are also less widely distributed throughout the password are easier for attackers to guess. However, perhaps surprisingly, for both PCFG and Hashcat attacks, the effect of usability disadvantages on the security of mobile passwords is insignificant under 10^{13} guesses. For attackers who can make more than 10^{13} guesses, the effect on security is substantial, with mobile passwords 32% easier to guess than traditional passwords.

Policy Recommendations

Previous research on text passwords has evaluated the security and usability trade-offs of different password policies on traditional devices [28, 34, 45]. However, the specific usability challenges of mobile devices cause the balance of usability and security to change. Recent work identified 2word16 and

3class12 as policies that provide both reasonable usability and good security against a range of attackers, with 3class12 providing more usability during password creation and 2word16 more security to a guessing attack [44]. Our data for traditional devices confirms this finding.

On mobile devices, however, the trade-offs are a bit different: 3class12 no longer provides as much security improvement over 3class8 as it did on traditional devices, while 2word16 remains strong. In addition, 3class12's advantage in usability over 2word16 narrowed, with no significant difference between the two policies in either creation failures or reported creation difficulty. Taken together, these results suggest that for systems with many mobile users, there is little reason to prefer 3class12 to 2word16, unless even minor improvements in usability are worth the more substantial loss of security.

Impact of Extra Text-Entry Features

We examined two types of additional text-entry features: password visiblility and enabling automatic text-entry tools.

Making Passwords Visible

We experimented with password visibility in the hope that visual feedback would help prevent users from mistyping their passwords. This hypothesis was confirmed, as users who could see their passwords in plain text made fewer creation attempts and deleted fewer characters during creation. In addition, visibility resulted in passwords overall more resistant to guessing, perhaps because users were willing to attempt more complex passwords when they could see them. In a real-world scenario, the increase in guessing resistance needs to be weighed against a higher potential for shoulder-surfing and screen capture attacks.

Automatic Text-Entry Tools

We hoped that allowing automatic text-entry tools, such as Swype, would allow users to enter passwords more quickly and reliably on mobile devices, an idea also considered by others [29]. Because actual use of these tools was not compulsory for participants, and therefore not experimentally controlled, our study of this aspect is exploratory. However, to our surprise, those who used these tools reported more frustration with password entry, deleted more characters during creation, and required more creation attempts. These findings were not statistically significant, perhaps because so few people actually used these automatic text-entry methods. It is possible that these tools do not align well with users' intentions when creating passwords or that users were not expecting to be able to use these features. For example, in our study, users in this category frequently deleted spaces, which are automatically inserted by many automated tools. It is also possible that using Swype in conjunction with other password policies, which better allow what these tools are designed for, might provide benefits. However, in the policy we studied, use of automated tools provided no overall security advantage, and some passwords in this group were guessed early.

While more research is necessary to explore alternative methods of password entry, if automatic text-entry tools are to be used for password entry, users need to be instructed about how best to take advantage of them [29]. Indeed, the approach we took yielded insight: the straightforward application of such tools actually causes inconvenience in a realistic situation. It might be even better, however, to modify these tools to operate differently in password fields than in standard text fields, for example, by not automatically inserting spaces.

CONCLUSION

Using simulations of real-world password-guessing attacks and surveys, we compared the security and usability properties of passwords based on whether they were created on mobile or traditional devices. We also examined the effects of several popular or recently recommended password-composition policies, as well as the impact of text-entry tools, such as making passwords visible and auto-completion tools. Our study is the first to examine the difference between mobile and traditional passwords with this degree of preciseness, breadth of factors, and scale.

We found and quantified a loss in usability when passwords are created or entered on mobile devices. Similarly, we found that passwords created on mobile devices are substantially weaker against a strong attacker, although against a less strong attacker they are not discernibly different in guessability from traditional passwords.

Based on our findings about password policies recommended for traditional devices—3class12 and 2word16—we suggest 2word16 (and not 3class12) for systems on which many users will be authenticating with mobile devices; in such a setting, 3class12 provides only negligible usability benefits over 2word16, but is roughly half as resistant to guessing attacks.

Finally, we explored how the usability problems of text-password entry on mobile devices can be mitigated by different input methods. Making passwords visible showed particular promise at encouraging users to make passwords both stronger and more usable. Augmenting that approach by allowing the use of automatic text-entry tools did not further help usability or improve security.

Based on our findings, we make the following recommendations to system designers:

- For systems with many mobile users where security is important and usability during creation is less so (e.g., mobile banking where users are likely to create the password on a traditional device), use a 2word16 password policy.
- For systems with many mobile users where usability during creation is important (e.g., shopping websites that require creating password before a purchase) use a 3class12 password policy.
- For systems where shoulder surfing attacks are uncommon, consider visible password creation or allow users to optionally enable password visibility during creation.
- To avoid user frustration, disable text-entry tools, e.g., Swype and spelling auto-correction, in password fields.

ACKNOWLEDGEMENTS

The authors thank Saranga Komanduri for his help. This research was supported in part by NSF grants DGE-0903659 and CNS-1116776 and gifts from the PNC Center for Financial Services Innovation and Microsoft Research.

REFERENCES

1. H. Akaike. 1974. A New Look at the Statistical Model Identification. *Automatic Control, IEEE Transactions on* 19, 6 (1974), 716–723. http://dx.doi.org/doi:10.1109/TAC.1974.1100705

2. Patti Bao, Jeffrey Pierce, Stephen Whittaker, and Shumin Zhai. 2011. Smart Phone Use by Non-Mobile Business Users. In *Proc. International Conference on Human Computer Interaction with Mobile Devices and Services*. http://doi.acm.org/10.1145/2037373.2037440

3. BBC. 2015. Ashley Madison: What's in the leaked accounts data dump? (August 2015). http://www.bbc.com/news/technology-33986228

4. Chandrasekhar Bhagavatula, Blase Ur, Kevin Iacovino, Su Mon Kywe, Lorrie Faith Cranor, and Marios Savvides. 2015. Biometric Authentication on iPhone and Android: Usability, Perceptions, and Influences on Adoption. In *Proc. Network and Distributed Systems Symposium Workshop on Usable Security*. http://dx.doi.org/10.14722/usec.2015.23003

5. Joseph Bonneau. 2010. The Gawker hack: how a million passwords were lost. (2010). http://www.lightbluetouchpaper.org/2010/12/15/the-gawker-hack-how-a-million-passwords-were-lost/.

6. Joseph Bonneau. 2012a. Statistical metrics for individual password strength. In *20th International Workshop on Security Protocols*. http://www.jbonneau.com/doc/B12-SPW-statistical_password_strength_metrics.pdf

7. Joseph Bonneau. 2012b. The science of guessing: analyzing an anonymized corpus of 70 million passwords. In *Proc. IEEE Symposium on Security and Privacy*. http://www.jbonneau.com/doc/B12-IEEESP-analyzing_70M_anonymized_passwords.pdf

8. Michael Buhrmester, Tracy Kwang, and Samuel D. Gosling. 2011. Amazon's Mechanical Turk: A New Source of Inexpensive, Yet High-Quality, Data? *Perspectives on Psychological Science* 6, 1 (2011), 3–5. http://pps.sagepub.com/content/6/1/3.abstract

9. William E. Burr, Donna F. Dodson, and W. Timothy Polk. 2006. *Electronic Authentication Guideline*. Technical Report. NIST. http://csrc.nist.gov/publications/nistpubs/800-63/SP800-63V1_0_2.pdf

10. Sonia Chiasson, Paul C van Oorschot, and Robert Biddle. 2006. A Usability Study and Critique of Two Password Managers.. In *Proc. USENIX Security Symposium*. http://dl.acm.org/citation.cfm?id=1267336.1267337

11. Yiannis Chrysanthou. 2013. *Modern Password Cracking: A hands-on approach to creating an optimised and versatile attack*. Master's thesis. Royal Holloway, University of London. https://www.ma.rhul.ac.uk/static/techrep/2013/MA-2013-07.pdf

12. D. R. Cox. 1972. Regression Models and Life-Tables. *Journal of the Royal Statistical Society. Series B (Methodological)* 34 (1972), 187–220. http://www.jstor.org/stable/2985181

13. Alexander De Luca, Marian Harbach, Emanuel von Zezschwitz, Max-Emanuel Maurer, Bernhard Ewald Slawik, Heinrich Hussmann, and Matthew Smith. 2014. Now You See Me, Now You Don't: Protecting Smartphone Authentication from Shoulder Surfers. In *Proc. CHI'14: 32nd Annual ACM Conference on Human Factors in Computing Systems*. 10. http://doi.acm.org/10.1145/2556288.2557097

14. M. Dell'Amico, P. Michiardi, and Y. Roudier. 2010. Password Strength: An Empirical Analysis. In *Proc. INFOCOM*. http://dx.doi.org/10.1109/INFCOM.2010.5461951

15. Julie S. Downs, Mandy B. Holbrook, Steve Sheng, and Lorrie Faith Cranor. 2010. Are your participants gaming the system? Screening Mechanical Turk workers. In *Proc. CHI'10: 28th Annual ACM Conference on Human Factors in Computing Systems*. http://doi.acm.org/10.1145/1753326.1753688

16. Paul Dunphy, Andreas P. Heiner, and N. Asokan. 2010. A Closer Look at Recognition-based Graphical Passwords on Mobile Devices. In *Proc. Symposium on Usable Privacy and Security*. Article 3, 12 pages. http://doi.acm.org/10.1145/1837110.1837114

17. Sascha Fahl, Marian Harbach, Yasemin Acar, and Matthew Smith. 2013. On the ecological validity of a password study. In *Proc. Symposium on Usable Privacy and Security*. http://doi.acm.org/10.1145/2501604.2501617

18. Adrienne Porter Felt and David Wagner. 2011. Phishing on Mobile Devices. In *Proc. Web 2.0 Security & Privacy*. http://www.w2spconf.com/2011/papers/felt-mobilephishing.pdf

19. Dinei Florêncio, Cormac Herley, and Baris Coskun. 2007. Do strong web passwords accomplish anything?. In *Proc. USENIX HotSec*. http://dl.acm.org/citation.cfm?id=1361419.1361429

20. Megan Geuss. 2015. Mozilla: data stolen from hacked bug database was used to attack Firefox. (Sep 2015). http://arstechnica.com/security/2015/09/mozilla-data-stolen-from-hacked-bug-database-was-used-to-attack-firefox/

21. Dan Goodin. 2012a. 8 million leaked passwords connected to LinkedIn, dating website. (Jun 2012). http://arstechnica.com/security/2012/06/8-

million-leaked-passwords-connected-to-linkedin/

22. Dan Goodin. 2012b. Why passwords have never been weaker—and crackers have never been stronger. *Ars Technica*. (August 2012). http://arstechnica.com/security/2012/08/passwords-under-assault/.

23. Dan Goodin. 2015. Once seen as bulletproof, 11 million+ Ashley Madison passwords already cracked. (Sep 2015). http://arstechnica.com/security/2015/09/once-seen-as-bulletproof-11-million-ashley-madison-passwords-already-cracked/

24. Kristen K. Greene, Joshua Franklin, and John Kelsey. 2015. Tap On, Tap Off: Onscreen Keyboards and Mobile Password Entry. http://www.shmoocon.org/speakers#tappedout. (2015). ShmooCon.

25. Kristen K. Greene, Melissa A. Gallagher, Brian C. Stanton, and Paul Y. Lee. 2014. I Can't Type That! P@$$w0rd Entry on Mobile Devices. In *Proc. Human Computer Interaction International*. http://dx.doi.org/10.1007/978-3-319-07620-1_15

26. S. M. Taiabul Haque, Matthew Wright, and Shannon Scielzo. 2013. Passwords and interfaces: towards creating stronger passwords by using mobile phone handsets. In *Proc. Workshop on Security and Privacy in Smartphones and Mobile Devices at ACM Conference on Computer and Communications Security*. http://doi.acm.org/10.1145/2516760.2516767

27. Marian Harbach, Emanuel von Zezschwitz, Andreas Fichtner, Alexander De Luca, and Matthew Smith. 2014. It's a Hard Lock Life: A Field Study of Smartphone (Un)Locking Behavior and Risk Perception. In *Proc. Symposium On Usable Privacy and Security*. https://www.usenix.org/conference/soups2014/proceedings/presentation/harbach

28. Philip Inglesant and M. Angela Sasse. 2010. The True Cost of Unusable Password Policies: Password Use in the Wild. In *Proc. CHI'10: 28th Annual ACM Conference on Human Factors in Computing Systems*. http://doi.acm.org/10.1145/1753326.1753384

29. Markus Jakobsson and Ruj Akavipat. 2012. Rethinking Passwords to Adapt to Constrained Keyboards. In *Proc. Mobile Security Technologies*. http://www.markus-jakobsson.com/fastwords.pdf

30. Ambarish Karole, Nitesh Saxena, and Nicolas Christin. 2011. A Comparative Usability Evaluation of Traditional Password Managers. In *Proc. International Conference on Information Security and Cryptology*. http://dl.acm.org/citation.cfm?id=2041036.2041056

31. Patrick Gage Kelley, Saranga Komanduri, Michelle L. Mazurek, Rich Shay, Tim Vidas, Lujo Bauer, Nicolas Christin, Lorrie Faith Cranor, and Julio Lopez. 2012. Guess again (and again and again): Measuring password strength by simulating password-cracking algorithms. In *Proc. IEEE Symposium on Security and Privacy*. http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6234434

32. Aniket Kittur, Ed H. Chi, and Bongwon Suh. 2008. Crowdsourcing User Studies With Mechanical Turk. In *Proc. CHI'08: 26th Annual ACM Conference on Human Factors in Computing Systems*. http://doi.acm.org/10.1145/1357054.1357127

33. Saranga Komanduri, Richard Shay, Lorrie Faith Cranor, Cormac Herley, and Stuart Schechter. 2014. Telepathwords: Preventing Weak Passwords by Reading Users' Minds. In *Proc. USENIX Security Symposium*. https://www.usenix.org/conference/usenixsecurity14/technical-sessions/presentation/komanduri

34. Saranga Komanduri, Richard Shay, Patrick Gage Kelley, Michelle L. Mazurek, Lujo Bauer, Nicolas Christin, Lorrie Faith Cranor, and Serge Egelman. 2011. Of Passwords and People: Measuring the Effect of Password-Composition Policies. In *Proc. CHI'11: 29th Annual ACM Conference on Human Factors in Computing Systems*. http://doi.acm.org/10.1145/1978942.1979321

35. KoreLogic. 2010-. "Crack Me If You Can" - DEFCON 2014. http://contest-2013.korelogic.com. (2010-).

36. Steve Kovach. 2014. We Still Don't Have Assurance From Apple That iCloud Is Safe. (September 2014). http://www.businessinsider.com/apple-statement-on-icloud-hack-2014-9

37. Tara Matthews, Jeffrey Pierce, and John Tang. 2009. No Smart Phone is an Island: The Impact of Places, Situations, and Other Devices on Smart Phone Use. In *IBM Research Report*. http://domino.research.ibm.com/library/cyberdig.nsf/papers/F5FD878B5B062ACA85257635004EC3F5/%24File/rj10452.pdf

38. Sergey Maydebura, Dong Hyun Jeong, and Byunggu Yu. 2013. Understanding Environmental Influences on Performing Password-Based Mobile Authentication. In *Proc. Information Reuse and Integration*. http://dx.doi.org/10.1109/IRI.2013.6642543

39. Michelle L. Mazurek, Saranga Komanduri, Timothy Vidas, Lujo Bauer, Nicolas Christin, Lorrie Faith Cranor, Patrick Gage Kelley, Richard Shay, and Blase Ur. 2013. Measuring Password Guessability for an Entire University. In *Proc. ACM Conference on Computer and Communication Security*. http://doi.acm.org/10.1145/2508859.2516726

40. PHDays. 2013. "Hash Runner"– Positive Hack Days. http://2013.phdays.com/program/contests/. (2013).

41. Ashwini Rao, Birendra Jha, and Gananand Kini. 2013. Effect of Grammar on Security of Long Passwords. In *Proc. ACM Conference on Data and Application Security and Privacy*. http://doi.acm.org/10.1145/2435349.2435395

42. Florian Schaub, Ruben Deyhle, and Michael Weber. 2012. Password Entry Usability and Shoulder Surfing Susceptibility on Different Smartphone Platforms. In *Proc. International Conference on Mobile and Ubiquitous Multimedia*. Article 13. http://doi.acm.org/10.1145/2406367.2406384

43. Roland Schlöglhofer and Johannes Sametinger. 2012. Secure and Usable Authentication on Mobile Devices. In *Proc. International Conference on Advances in Mobile Computing & Multimedia*. 6. http://doi.acm.org/10.1145/2428955.2429004

44. Richard Shay, Saranga Komanduri, Adam L. Durity, Phillip (Seyoung) Huh, Michelle L. Mazurek, Sean M. Segreti, Blase Ur, Lujo Bauer, Nicolas Christin, and Lorrie Faith Cranor. 2014. Can Long Passwords Be Secure and Usable?. In *Proc. CHI'14: 32nd Annual ACM Conference on Human Factors in Computing Systems*. 10. http://doi.acm.org/10.1145/2556288.2557377

45. Richard Shay, Saranga Komanduri, Patrick Gage Kelley, Pedro Giovanni Leon, Michelle L. Mazurek, Lujo Bauer, Nicolas Christin, and Lorrie Faith Cranor. 2010. Encountering Stronger Password Requirements: User Attitudes and Behaviors. In *Proc. Symposium on Usable Privacy and Security*. http://doi.acm.org/10.1145/1837110.1837113

46. Jens Steube. 2009-. Hashcat Advanced Password Recovery. https://hashcat.net/oclhashcat/. (2009-).

47. Michael Toomim, Travis Kriplean, Claus Pörtner, and James Landay. 2011. Utility of Human-Computer Interactions: Toward a Science of Preference Measurement. In *Proc. CHI'11: 29th Annual ACM Conference on Human Factors in Computing Systems*. http://doi.acm.org/10.1145/1978942.1979277

48. Blase Ur, Patrick Gage Kelly, Saranga Komanduri, Joel Lee, Michael Maass, Michelle Mazurek, Timothy Passaro, Richard Shay, Timothy Vidas, Lujo Bauer, Nicolas Christin, and Lorrie Faith Cranor. 2012. How Does Your Password Measure Up? The Effect of Strength Meters on Password Creation. In *Proc. USENIX Security Symposium*. http://dl.acm.org/citation.cfm?id=2362793.2362798 .

49. Blase Ur, Sean M. Segreti, Lujo Bauer, Nicolas Christin, Lorrie Faith Cranor, Saranga Komanduri, Darya Kurilova, Michelle L. Mazurek, William Melicher, and Richard Shay. 2015. Measuring Real-World Accuracies and Biases in Modeling Password Guessability. In *24th USENIX Security Symposium (USENIX Security 15)*. USENIX Association, Washington, D.C., 463–481. https://www.usenix.org/conference/usenixsecurity15/technical-sessions/presentation/ur

50. Emanuel von Zezschwitz, Alexander De Luca, and Heinrich Hussmann. 2014. Honey, I Shrunk the Keys: Influences of Mobile Devices on Password Composition and Authentication Performance. In *Proc. NordiCHI'14: 12th Annual Nordic Conference on Human-Computer Interaction*. 10. http://doi.acm.org/10.1145/2639189.2639218

51. Emanuel von Zezschwitz, Paul Dunphy, and Alexander De Luca. 2013. Patterns in the Wild: A Field Study of the Usability of Pattern and PIN-Based Authentication on Mobile Devices. In *Proc. International Conference on Human-Computer Interaction with Mobile Devices and Services*. 10. http://doi.acm.org/10.1145/2493190.2493231

52. Matt Weir, Sudhir Aggarwal, Michael Collins, and Henry Stern. 2010. Testing Metrics for Password Creation Policies by Attacking Large Sets of Revealed Passwords. In *ACM Conference on Computer and Communication Security*. http://doi.acm.org/10.1145/1866307.1866327

53. M. Weir, S. Aggarwal, B. de Medeiros, and B. Glodek. 2009. Password Cracking Using Probabilistic Context-Free Grammars. In *Proc. IEEE Symposium on Security & Privacy*. http://dx.doi.org/10.1109/SP.2009.8

54. Yulong Yang, Janne Lindqvist, and Antti Oulasvirta. 2014. Text Entry Method Affects Password Security. *Computing Research Repository* (2014). http://arxiv.org/abs/1403.1910