YourPassword: Applying Feedback Loops to Improve Security Behavior of Managing Multiple Passwords

Tiffany Hyun-Jin Kim* H. Colleen Stuart† Hsu-Chun Hsiao* Yue-Hsun Lin*
Leon Zhang* Laura Dabbish* Sara Kiesler*
* Carnegie Mellon University † Johns Hopkins University

ABSTRACT

Various mechanisms exist to secure users' passwords, yet users continue to struggle with the complexity of multiple password management. We explore the effectiveness of a feedback loop to improve users' password management. We introduce YourPassword, a web-based application that uses feedback to inform users about the security of their password behavior. YourPassword has two main components: a password behavior checker that converts password strengths into numerical scores and a dashboard interface that visualizes users' overall password behavior and provides visual feedback in real time. YourPassword not only provides a total score on all passwords, but also visualizes when passwords are too similar to each other. To test the efficacy of YourPassword, we conducted a between-subjects experiment and think-aloud test with 48 participants. Participants either had access to YourPassword, an existing commercial password checker, or no password tool (control condition). YourPassword helped participants improve their password behavior as compared with the commercial tool or no tool.

Categories and Subject Descriptors

K.6.5 [**MANAGEMENT OF COMPUTING AND INFORMATION SYSTEMS**]: Security and Protection—*Authentication*; H.5.2 [**INFORMATION INTERFACES AND PRESENTATION (e.g., HCI)**]: User Interfaces—*User-centered design*; H.1.2 [**MODELS AND PRINCIPLES**]: User/Machine Systems—*Human factors*

Keywords

Authentication; Feedback Loops; Password Management

1. INTRODUCTION

Password-based online authentication is a primary way users log in securely to various websites. According to a study conducted in 2007, a typical user types approximately eight passwords each day [9]. The same study suggests that users are reusing the same or very similar passwords on multiple websites, maintaining 25 passwords but actively using about 7 distinct passwords. Reuse of passwords increases the security risks of password breach, but people have trouble recalling many distinct passwords.

Password managers of various types attempt to help people manage multiple different passwords without having to remember them (e.g., LastPass,[1] SplashID[2]). Although these password managers are useful, they are vulnerable in different ways: reliant on the strength of a master password, open to physical attacks, or subject to usability and convenience issues because of lack of portability.

We take a different approach and explore the effectiveness of a feedback loop to improve users' multiple password management behavior. Previous research in behavioral science suggests that a feedback loop can provide people with timely information about their actions and opportunities to improve them [11]. We argue that by making people more aware of their own behavior and how it is linked to their security, we can lead them to make improvements.

In this paper, we present the YourPassword system, which explores how to apply feedback loops to remind users of their own password security behavior and encourage them to choose multiple dissimilar passwords. YourPassword algorithms measure the strength of the user's passwords and translate them into easily understandable scores. Our algorithm considers password similarities and reuse among different websites.

YourPassword is a web-based dashboard application implemented as a Chrome browser extension. It provides password information and advice to users based on the algorithms we developed. We conducted a user study to measure the usability and usefulness of YourPassword. Our results suggest that the feedback loops in YourPassword helped users become more aware of their password choices and create more secure passwords.

2. PROBLEM DEFINITION

Our goal is to encourage users to create unique, dissimilar, and strong passwords for different websites, and to periodically update their passwords. Doing so is a challenge because users favor convenient, simple, and memorable passwords. In addition, overly salient indicators may overwhelm people, making them feel inadequate and unmotivated. At the same time, overly subtle indicators will not catch people's

ASIA CCS'14, June 4–6, 2014, Kyoto, Japan.
Copyright 2014 ACM 978-1-4503-2800-5/14/06 ...$15.00.
http://–enter the whole DOI string from rightsreview form confirmation.

[1] `https://lastpass.com/`

[2] `http://splashdata.com/splashid/`

attention. We provide a feedback loop that gives people information about the effectiveness of their own password choices in a way that will encourage them to exert more care over their password management behavior.

Assumptions. We assume that we can gather relevant password information for all the user's websites and analyze the user's passwords in real time. We also assume that the gathered passwords are securely stored on the user's local machine and that our application does not leak stored passwords. In Sections 4.2 and 6, we discuss an alternative approach to relax our assumptions.

Desired properties. The following usability and performance properties are desired for an effective password behavior indicator: easy to use and intuitive, motivating (to choose stronger and more unique passwords), timely (catching the password information in real time), and accurate.

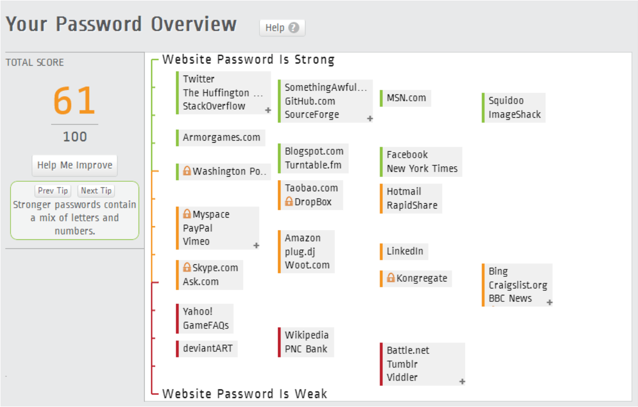
Adversary model. Attackers may attempt to guess as many passwords as possible to gain access to users' sensitive information. Accessing one password may provide exponential benefits to the attacker if that password is reused on multiple websites.

3. PASSWORD BEHAVIOR CHECKER

We designed a password scoring mechanism that considers a multitude of parameters to evaluate a password's security level. Unlike existing mechanisms [4, 5, 7, 13, 14, 23, 24], our mechanism adjusts the score based on the similarity to all the passwords of a user and the corresponding website's sensitivity level: we apply stricter rules for those passwords for highly sensitive websites (e.g., bank, email) compared to the passwords for moderately sensitive websites (e.g., forum, classifieds).

Our password behavior checker is composed of 5 modules:

- **Individual password strength:** This module checks if the password has a certain number of characters (e.g., minimum 8 characters for sensitive websites), contains dictionary words or common passwords,[3] contains uppercase characters, contains special characters, and is composed of unique characters.
- **Password reuse:** This module checks if the same password is reused on multiple websites.
- **Website sensitivity:** This module checks if the same password is reused on multiple websites with different sensitivity levels.
- **Password encryption:** This module checks if the password is transmitted without encryption.
- **Password similarity:** This module calculates how similar a user's password is to his/her other passwords.



Stronger passwords contain a mix of letters and numbers.

Figure 1: YourPassword dashboard.

For each site u, we denote pw_u, the password for u, and ℓ_u, the sensitivity level of u. Our password behavior checker (PBC) computes a score S_u for this site given pw_u and ℓ_u as well as the passwords and sensitivity levels of other sites. Specifically, $S_u = PBC(pw_u, \ell_u, \{pw_i, \ell_i | \forall i \neq u\})$. S_u is the combination of the normalized scores from all 5 modules, and this score can be potentially updated whenever the user enters any password (since our algorithm considers similarities of all of his/her passwords). For most modules (except password similarity), website's sensitivity levels affect the generosity of the password scores as follows:

$$S_i \leq S_j \quad \forall \ell_i > \ell_j, pw_i = pw_j.$$

That is, if two sites have the same password but different sensitivity levels, the more sensitive site has a lower score.

Determining websites' sensitivity levels is a non-trivial task due to differences in individuals: a website that Alice considers to be sensitive may not be sensitive to Bob. On the other hand, any websites that are vulnerable to security and privacy breaches should be labeled as sensitive. Hence, YOURPASSWORD provides a default list of sensitive websites, such as financial corporations that are prone to phishing attacks.[4] Furthermore, YOURPASSWORD enables a user to update the list to include websites that (s)he considers to be sensitive.

The password behavior checker multiplies the fractional scores from the above five modules and outputs two scores: (1) a normalized score (0-100) for a recently entered password and (2) an average score for all of a user's passwords.

4. USER INTERFACE

The YOURPASSWORD interface visualizes a user's overall password behavior for all websites along with strength and uniqueness scores for each website's password.

4.1 YourPassword Description

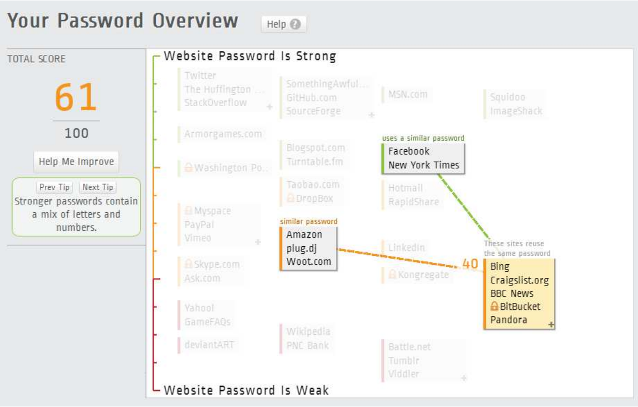
YOURPASSWORD is a browser dashboard that displays:

- An overall score for a user's passwords based on their relative strength,
- Websites that share the same passwords,
- Visual groupings of websites with similar passwords,
- An individual score for each password based on its strength,
- A visual hierarchy, such that website groups with high scores are shown at top of the screen,
- A "Help me improve" button that displays multiple pieces of information to help users improve their password scores and security.

Figure 1 illustrates the YOURPASSWORD interface. The main display chart visualizes the user's overall password behavior. The scoring feedback is designed to range from 0 to 100 to make it easy for users to understand. To reinforce differences, we apply colors to differentiate score groupings

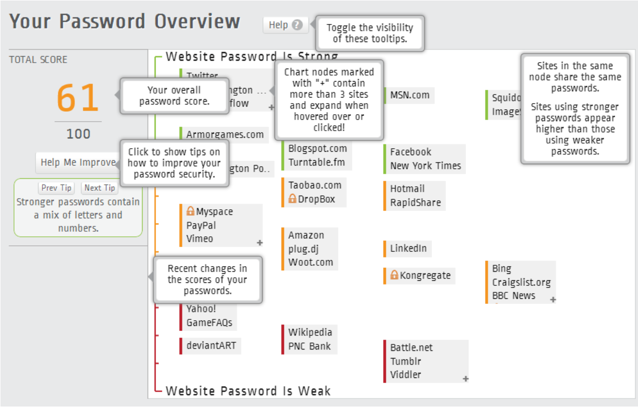
[3] For our prototype, we used the top 100 Adobe passwords that were recently exposed in November 2013.

[4] http://docs.apwg.org/reports/APWG_GlobalPhishingSurvey_2H2011.pdf



Stronger passwords contain a mix of letters and numbers.

Figure 2: Individual node in the dashboard. When a user hovers over a node, the dashboard grays out the background and emphasizes the node, displaying the password's score along with links to other nodes with similar passwords.



Toggle the visibility of these tooltips.

Your overall password score.

Chart nodes marked with "+" contain more than 3 sites and expand when hovered over or clicked!

Sites in the same node share the same passwords.

Sites using stronger passwords appear higher than those using weaker passwords.

Click to show tips on how to improve your password security.

Stronger passwords contain a mix of letters and numbers.

Recent changes in the scores of your passwords.

Figure 3: Help button. When a user clicks the "Help" button, tool tips for each section are displayed to help users understand the dashboard and interpret their current password behavior.

(e.g., scores from 0 – 39, 40 – 69, 70 –100 are in red, orange, and green, respectively).

Websites that share the same password are grouped together and displayed in the same node. In Figure 1, the user has a unique password for MSN.com but shares the same password for Facebook and New York Times. (A node marked with the "+" sign indicates that a password is shared on more than 3 websites.) Sensitive websites are represented with a lock icon.

We apply the same coloring scheme to individual nodes to help users understand the relationship between the groups of websites and their password strengths. We also display an individual score for each password. When the user hovers over a node, it grays out the background and emphasizes the node, and the score for the password is displayed on the left side of the node. YourPassword also draws links to other nodes with similar passwords when a user hovers over a node (Figure 2). The "Help" button on top of the dashboard toggles the visibility of the tool tips (Figure 3).

The advice for improvement includes suggestions to update a specific website's password and reminders to update passwords regularly. The advice is dynamically updated as users enter their passwords, reducing potential habituation.

4.2 Implementation

We implemented YourPassword as a Google Chrome browser extension. The extension consists of three main modules – password extractor (PE), score generator (SG), and password database (PD).

Tracking is automatically accomplished by the PE module that monitors a user's login activities. After the installation, the PE module records the user's passwords at login pages by parsing the HTML source in the background and looking for the input field of a password type. When the user clicks the login button, the PE module captures the information $T = \{pwd, fn, d\}$, which includes the entered password (pwd), the password field name (fn), and the visited domain address (d), and then waits for a submission callback. Users may choose Chrome's incognito mode if they do not desire PE to track the credentials of particular websites.

The login information T is first saved at a temporary database, until a redirect page is loaded. If the redirect page has a password input field, we assume a login failure and delete the entry from the temporary database. Otherwise, we assume a successful login and load the temporary database into memory. When the login is successful, T is sent to the SG module using the Chrome extension API call `chrome.extension.sendRequest`. The SG module then calculates the score based on the currently entered password and all the previously stored passwords. To ensure privacy and secrecy of the passwords, the PD module encrypts and stores in the user's local machine the passwords in cleartext, total scores, individual module scores, T, and the website address.

Limitations. Some websites may prevent YourPassword from capturing passwords, for example by enabling javascripts. One possible solution is to ask users to manually enter passwords and their corresponding websites to YourPassword. Doing so would reduce the usability and convenience of appliance.

5. EVALUATION

We conducted a user study to evaluate whether YourPassword improves users' password behavior.

5.1 Method

The study was a between-subjects experiment in which participants played the role of a twin sibling, Robin, who suspects that eight personal accounts are potentially compromised by hackers. We gave the participant *Robin's* accounts and their corresponding passwords (Table 1). We asked the participant to improve the security of Robin's passwords. Two sets of passwords were similar to each other, and we reused the same password for two accounts. Six passwords, slightly modified, were among the most common in 2012.[5] Two were strings of random characters. We explained that hackers may steal private and sensitive information, including personal emails, private photos, private purchase history, personal medical history, as well as bank, credit card, and financial investment information. We emphasized that all Robin's accounts are equally important and sensitive, and have the same level of security risk.

The experiment had three conditions:

1. Control condition (CC): Participants were asked to up-

[5] `http://www.cnn.com/2012/10/25/tech/web/worst-passwords-2012/`

Table 1: A list of accounts and passwords for the experiment. We had two sets of passwords that were similar to each other, and the same password for two accounts.

Service	Password	Similar	Same
Amazon	`!iloveyou`	①	
American Express	`Abcd123`	②	
Bank of Oklahoma	`passw0rd`		✓
Etrade	`eswr@U1ayP`		
Facebook	`passw0rd`		✓
Gmail	`@w43df2rxTL6^`		
MyChart@Johns Hopkins	`abcd123`	②	
PayPal	`iloveyou!`	①	

date Robin's passwords. They were given no hints about the original or updated password strengths.

2. Microsoft password checker condition (MC): Microsoft's application allows users to check the strength of a password.[6] Participants in MC were asked to suggest new password(s), check each of them using Microsoft's password checker, and record the final passwords when they were happy with their strengths. They were allowed to try passwords multiple times until they were satisfied.
3. YourPassword condition (YC): Participants were asked to use YourPassword. After seeing the overall security of Robin's current passwords from the YourPassword application, they were asked to suggest new password(s), check each of them using YourPassword, and record the final passwords when they were satisfied. As in MC, they could repeat multiple times.

We hypothesized that participants in the YourPassword condition (YC) would suggest more password changes, stronger passwords, and more unique passwords.

5.2 Sample

We recruited 48 participants from Carnegie Mellon University's participant pool service. We also recruited local residents and students. We randomly assigned participants to each of the three conditions. Participants included 26 men and 22 women, ranging in age from 18 to 46 years old. Thirty-five were students, and of these students, 27 were majoring in engineering or computer science.

5.3 Procedure

A pretest asked participants if they had accounts with the websites listed in Table 1, if they thought it would be important to protect their personal data, how many unique passwords they currently had, and what strategy they used to remember their passwords. Participants were then randomly assigned to a condition and asked to follow the instructions described above. They were asked to think aloud and explain which password(s) they would update and their reason(s) for doing so. After the participants had reported their final password selections, they were asked to recall their final passwords for all eight accounts, including the passwords they did not suggest changing. We also asked participants whether they applied a policy to secure Robin's accounts, and what strategy they would use to remember the passwords they observed or created during the study.

5.4 General Observations

We observed that participants had fundamental security knowledge to create secure passwords *individually*. For example, all participants followed their own password policies to create a new password and proposed to update passwords that did not include a number, an uppercase character, or a symbol. However, without proper feedback, all participants in CC and MC conditions reused the same password on multiple websites, whereas all participants in YC proposed distinct passwords during our experiment.

Some participants in CC and MC conditions reused the same or similar passwords for services that are linked for usage. For example, two participants, one in CC and one in MC, reused the same or similar passwords for those services dealing with banking or credit card information (i.e., Amazon, American Express, Bank of Oklahoma, and PayPal). Also, one participant in MC mentioned that she would reuse the same password for Amazon, Facebook, and Gmail, since she logs into these services using the same email address.

Some participants did not trust the strength measurement using Microsoft's password checker, because they believed that their password policies would generate secure passwords. An interesting observation was that a number of participants used the initials of the services to create new passwords for memorability: not only these participants generated passwords that are meaningful to them but also linked with the services to tie individual passwords to the corresponding services.

Participants in YC immediately started to update the same or similar passwords after examining the feedback interface. On average, participants made 1.8 attempts ($\sigma = 0.71$, max $= 9$) to update the passwords for the accounts they wished to update. All participants tried hard to increase both individual password scores and overall score. However, they gave up after multiple attempts if the scores did not increase significantly.

5.5 Results

The results of an ANOVA test support the hypothesis that the feedback-based YourPassword interface (YC) improved participants' password behavior as compared with response to the commercial individual password checker (MC) or getting no feedback (CC). There were no effects of participants' education level, major field of study, occupation, age, or gender.

As shown in Table 2, participants' overall security level, use of unique passwords, and dissimilar passwords were significantly higher when they used the YourPassword application. The CC and MC conditions did not differ.

During the post-test, we measured how many final passwords participants could remember correctly. Memorability is an important issue because users may avoid passwords that are difficult to recall. We did not find statistically significant differences in the number of final passwords that participants remembered in different conditions. We also gave a 5-point Likert scale question to measure how useful participants found the application. Participants found both YourPassword and the commercial password checker somewhat useful but their usefulness scores were not significantly different. For the YC condition, we also asked how useful each feature in YourPassword was to the participants. On average, participants using YourPassword found the following features at least somewhat helpful: the total score ($\mu = 3.75, \sigma = .88$), individual score ($\mu = 4.58, \sigma = .56$), passwords similarity indicators ($\mu = 4.19, \sigma = .56$), grouping of the reused passwords ($\mu =$

[6] https://www.microsoft.com/security/pc-security/password-checker.aspx

Table 2: Summary of ANOVA test results for overall security, password uniqueness, similarity, memorability, usefulness, and likeliness to use to analyze the efficacy of **YourPassword** (YC) compared to individual password checker (MC) and no feedback (CC) ($N = 48$). The higher mean that is statistically significant from the others is highlighted in bold.

	Overall min: 0, max: 100		Uniqueness min: 0, max: 8		Similarity min: 0, max: 1		Memorability min: 0, max: 8		Usefulness min:1, max:5		Likeliness min:1, max:5	
	μ	$\sigma_{\bar{X}}$	μ	$\sigma_{\bar{X}}$	μ	$\sigma_{\bar{X}}$	μ	$\sigma_{\bar{X}}$	μ	σ	μ	σ
YC	**77.83**	**1.75**	**8.00**	**.00**	**.97**	**.01**	2.44	.70	4.28	.63	**4.31**	**.66**
MC	44.38	6.96	6.71	.49	.73	.06	4.06	.68	3.97	1.11	3.00	1.31
CC	44.46	8.54	5.60	.68	.71	.08	2.93	.71				
Results	$F(2,45) = 9.17$ $p < .0005$		$F(2,45) = 6.18$ $p = .004$		$F(2,45) = 5.74$ $p = .006$		$F(2,45) = 1.48$ $p = .239$		$F(1,31) = .96$ $p = .34$		$F(1,31) = 12.97$ $p = .001$	

μ: mean, $\sigma_{\bar{X}}$: standard error, σ: standard deviation

$4.59, \sigma = .91$), and placing passwords along the y-axis based on their scores ($\mu = 4.59, \sigma = .37$). Finally we asked participants in the MC and YC studies how likely it was that they would use the application to analyze their own password behavior. Participants were statistically significantly more likely to use YourPassword compared with the commercial password checker.

At the end of the study, we asked participants to share their strategies for remembering complex passwords. They mentioned writing passwords on paper, and sending themselves emails or text messages with passwords or password hints. Twenty participants said that they enable their browsers to remember their passwords. Just a few participants said that they would avoid storing their passwords for sensitive websites, such as banks or credit cards.

Our evaluation suggests that participants had sufficient security knowledge to create secure passwords individually. All participants followed their own password policies to create a new password. Several participants used the initials of the services to create new passwords for memorability. Participants also proposed updating passwords that did not include a number, an uppercase character, or a symbol. However, without feedback, all participants in the CC and MC conditions reused the same password on multiple websites, whereas all participants in YC proposed distinct passwords during our experiment. Because YourPassword suggests that users avoid reusing passwords, and our participants just did that, this feature may be especially valuable.

6. DISCUSSION

Regardless of its usability, any password manager must itself be safe. As is true of other password managers, YourPassword's current implementation stores encrypted passwords in a database so that YourPassword can accurately measure the similarity between a new password and old passwords. Unlike commercial password managers that ask users to generate a master encryption key, YourPassword's encryption key is a randomly generated string that is difficult for an attacker to break, and the user does not need to remember the key. Thus, the current implementation of YourPassword provides a better level of security compared with other password managers in the sense that the passwords are protected using a full-length random secret key that is hard to break even if the encrypted passwords are leaked. YourPassword can further increase the level of protection by storing the encryption key in sealed storage using the trusted computing technology. However, neither password managers nor YourPassword can defend against a strong adversary capable of stealing the encryption key.

We might further minimize the attack surface of YourPassword by storing the hashes of the passwords instead of storing the encrypted passwords. We can use two types of hash functions: a cryptographic hash function (e.g., SHA256) for exact matching, and a Locality Sensitive Hashing (LSH) function for fuzzy matching. Exact matching is needed to detect password reuse, whereas fuzzy matching is needed to evaluate the similarity between passwords. We can obtain an estimate of the distance between two passwords by using multiple LSH hashes with different threshold values. Since LSH is probabilistic, this modified version of YourPassword would eliminate the risk of master key leakage at the cost of the accuracy of a score. However, in contrast to cryptographic hash algorithms, LSH algorithms may suffer from preimage attacks – the attacker recovers the input from the hash value in short time (e.g., much fewer than 2^L attempts, where L is the hash length) – due to the additional distance information leaked to the attacker. We leave it as future work to formally investigate the trade-offs between security and score accuracy.

7. RELATED WORK

Feedback loops. Websites provide password strength meters to provide real-time feedback on the strength of a password as a user types. Such meters encourage users to create longer passwords [8,20]. YourPassword is different from this previous work in two aspects: YourPassword visualizes the relationship among all the user's passwords and provides feedback on the use of similar passwords.

Password vulnerabilities. Compromising less sensitive sites in order to attack sites with high security has been observed in websites using email addresses as user identifiers [3]. Several researchers have analyzed the actual deployment of such an attack [15,21], and many schemes have been developed to evaluate password strength based on attack resistance [4–7, 14, 23, 24]. A recent study revealed that long passwords with no other restrictions provide resistance to guessing attacks, and that the dictionary check relies heavily on the choice of dictionary [13]. We designed our password strength checking algorithms to incorporate findings from these previous studies.

Human factors for password mechanism. Adams and Sasse emphasized the importance of considering human factors when designing security mechanisms, including passwords [1]. They noted the importance of making system security visible to users, such as providing feedback during password construction process. Users experience difficulties in creating and remembering attack-resistant passwords under strict policies [12, 16–19, 22]. Rather than aiming to replace passwords, our goal is to help users become aware of their password behavior based on feedback loops and encourage them to improve their password behavior.

Prior studies indicate that the majority of users reuse their passwords across multiple websites [9, 10]. A large-scale password study also revealed that strong passwords are used at fewer sites on average compared to weak passwords [9]. Based on these findings, YourPassword aims to help users avoid sharing the same password on multiple websites by visualizing such information.

A recent proposal uses stories and pictures to help users create memorable passwords [2]. Such a scheme can be combined with YourPassword to help users create secure, unique, and memorable passwords.

8. CONCLUSION

People want to improve their password management behavior but they need help in doing so. Although various password managers have been developed to help users manage multiple, possibly unique passwords for various websites, they face security vulnerabilities and may not adequately change user behavior. Our solution is to apply feedback loops, the effectiveness of which has been demonstrated in the past, such as in speed-limit control systems [11]. By providing a non-invasive reminder about users' overall password behavior, they become more aware of their current actions and are encouraged to update weak and reused passwords to improve their security. The results of our experimental evaluation suggest further research to help users improve their own password behavior without relying on complex and possibly vulnerable external management tools.

9. ACKNOWLEDGMENTS

We gratefully thank Payas Gupta, Ruogu Kang, Peter Kinnard, Adrian Perrig, and Akshay Udiavar for their insightful feedback and help with the interface design and implementation. We also thank anonymous reviewers for their valuable comments. This research was supported by NSF under awards CNS-1040801 and CNS-1221006.

10. REFERENCES

[1] A. Adams and M. A. Sasse. Users are not the Enemy. *Communications of the ACM*, December 1999.

[2] J. Blocki, M. Blum, and A. Datta. Naturally Rehearsing Passwords. In *Proceedings of ASIACRYPT*, 2013.

[3] J. Bonneau and S. Preibusch. The Password Thicket: Technical and Market Failures in Human Authentication on the Web. In *Proceedings of WEIS*, 2010.

[4] W. E. Burr, D. F. Dodson, and W. T. Polk. Electronic Authentication Guideline. Technical report, NIST, 2006.

[5] C. Castelluccia, M. Durmuth, and D. Perito. Adaptive Password-Strength Meters from Markov Models. In *Proceedings of NDSS*, 2012.

[6] A. Das, J. Bonneau, M. Caesar, N. Borisov, and X. Wang. The Tangled Web of Password Reuse. In *Proceedings of NDSS*, 2014.

[7] M. Dell'Amico, P. Michiardi, and Y. Roudier. Password Strength: An Empirical Analysis. In *Proceedings of INFOCOM*, 2010.

[8] S. Egelman, A. Sotirakopoulos, I. Muslukhov, K. Beznosov, and C. Herley. Does My Password Go up to Eleven? The Impact of Password Meters on Password Selection. In *Proceedings of CHI*, 2013.

[9] D. Florencio and C. Herley. A Large-Scale Study of Web Password Habits. In *Proceedings of WWW*, 2007.

[10] S. Gaw and E. W. Felten. Password Management Strategies for Online Accounts. In *Proceedings of SOUPS*, 2006.

[11] T. Goetz. Harnessing the Power of Feedback Loops. Wired Magazine, June 2011.

[12] P. Inglesant and M. A. Sasse. The True Cost of Unusable Password Policies: Password Use in the Wild. In *Proceedings of CHI*, 2010.

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

[14] A. Narayanan and V. Shmatikov. Fast Dictionary Attacks on Passwords Using Time-Space Tradeoff. In *Proceedings of CCS*, 2005.

[15] B. Prince. Twitter Details Phishing Attacks Behind Password Reset. eWeek, January 2010.

[16] R. W. Proctor, M.-C. Lien, K.-P. L. Vu, E. E. Schultz, and G. Salvendy. Improving Computer Security for Authentication of Users: Influence of Proactive Password Restrictions. *Behavior Research Methods, Instruments, & Computers*, 2002.

[17] R. Shay and E. Bertino. A Comprehensive Simulation Tool for the Analysis of Password Policies. *International journal of Information Security*, 2009.

[18] R. Shay, A. Bhargav-Spantzel, and E. Bertino. Password Policy Simulation and Analysis. In *Proceedings of ACM Workshop on Digital Identity Management*, 2007.

[19] J. M. Stanton, K. R. Stam, P. Mastrangelo, and J. Jolton. Analysis of End User Security Behaviors. *Computer & Security*, 2005.

[20] B. Ur, P. Kelley, S. Komanduri, J. Lee, M. Maass, M. L. Mazurek, T. Passaro, R. Shay, T. Vidas, L. Bauer, N. Christin, and L. F. Cranor. How Does Your Password Measure Up? The Effect of Strength Meters on Password Creation. In *Proceedings of USENIX*, 2012.

[21] A. Vance. If your Password is 123456, Just Make It HackMe. The New York Times, January 2010.

[22] K.-P. L. Vu, R. W. Proctor, A. Bhargav-Spantzel, B.-L. B. Tai, J. Cook, and E. E. Schultz. Improving Password Security and Memorability to Protect Personal and Organizational Information. *International Journal of Human-Computer Studies*, 2007.

[23] M. Weir, S. Aggarwal, B. de Medeiros, and B. Glodek. Password Cracking using Probablistic Context-Free Grammars. In *Proceedings of IEEE Symposium on Security and Privacy*, 2009.

[24] Y. Zhang, F. Monrose, and M. K. Reiter. The Security of Modern Password Expiration: An Algorithmic Framework and Empirical Analysis. In *Proceedings of CCS*, 2010.