This work was presented at 9th Symposium On Usable Privacy and Security (SOUPS 2013), July 24-26, 2013, Newcastle, UK and available for downloading at http://cups.cs.cmu.edu/soups/2013/posters/soups13_posters-final19.pdf.

Poster: Visual Password Checker

Kyriakos Kafas
University of Cambridge, UK
kyriakos.kafas@gmail.com

Nouf Aljaffan
University of Surrey, UK
n.aljaffan@surrey.ac.uk

Shujun Li
University of Surrey, UK
shujun.li@surrey.ac.uk

1. INTRODUCTION

Nowadays, static textual passwords are still the most commonly used technique for user authentication. It has been well known that a usability-security dilemma exists for users' choices of static textual passwords: to resist many attacks especially offline dictionary attacks users need to choose stronger passwords, however, stronger passwords are normally difficult to remember so users often end up with weak but more memorizable passwords.

There have been many proposed (partial) solutions to the above usability-security dilemma about static textual passwords, such as enforcing strong password policies, using computer generated (stronger) passwords, using dynamic passwords generated by a hardware device or sent via an out-of-band (OOB) channel, moving to multi-factor user authentication. This poster focuses on a solution working at the user interface level: proactive password checkers (or meters) which persuade and educate users to choose stronger passwords via *immediate* feedback of the security strength of the passwords chosen by users. Proactive password checkers can work with other solutions e.g. to enforce a password policy by adapting the password strength estimate to the specific policy. A password checker may also be used as an offline tool for users to evaluate the strength of their current passwords [1][2].

Proactive password checkers have been widely deployed on the user registration pages of many web sites. Figure 1 shows one used by Google. All password checkers we are aware of show the estimated security strength as a 1-D bar with a textual description and/or a numeric strength score. Some password checkers also provide hints about how to choose stronger passwords, but very few [1] reveal detail about the underlying password strength estimation algorithm.

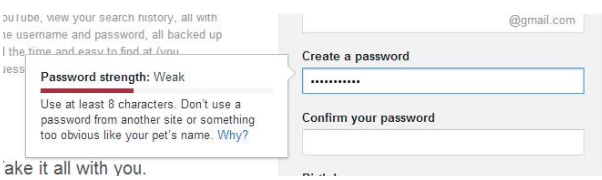

Figure 1. Google's proactive password checker in use.

One of the problems caused by the simple user interface of existing password checkers is that users often have little clue about why a password is weak and how to further improve it. Users often attempt to try more choices until successfully escaping from reported "weak" passwords. In other words, even when a password checker does help define stronger passwords, users are still not well educated about different kinds of risks behind password security. When an inappropriate password strength estimator is used, this may mislead the user to use seemingly strong but actually weak passwords [3]. The use of different password strength estimator by different password checkers can cause confusions to users. Yet another problem about existing password checkers is that the visual information is basically redundant since the 1-D bar carries the same amount of information as a numeric score, implying that the power of information visualization is not fully explored.

This poster presents a novel scheme for designing password checkers which provides users with immediate feedback about multiple threats detected on the current password choice through 2-D visual guidance in order to influence users to define stronger passwords based on the visual feedback. It provides the following new features that cannot be found in existing password checkers: 1) it visualizes *multiple* threats detected on the current password choice simultaneously, 2) it makes use of a 2-D visual space to show detected threats in a structural approach, 3) it uses a public and standard password strength estimator defined by NIST [5], 4) it provides an open interface to add more static dictionaries; 5) it supports a "smart" dictionary to cover password composition rules, 6) it supports personalized dictionaries through information gathered from the user's social network accounts, 7) it provides detailed information about *each* detected threat to better educate users, 8) it is a pure client side solution so can be easily integrated into any web site with minimum change to the server, 9) being much more complicated than any existing password checker, it is still fairly fast and can run in real time even from resource-constrained devices like smart phones. We name the proposed scheme Visual Password Checker (VPC) to highlight its making effective use of a 2-D visual space.

In the following, we describe the general design of the VPC framework and then our prototype implementation. Some ongoing and future work will be given which concludes the poster abstract.

2. SYSTEM DESIGN

The basic idea behind VPC is to extend the simple 1-D bar used by all existing password checkers to a 2-D space in which different threats detected for the current password choice are visualized. Three types of threats are considered in the current design of VPC: brute force attacks, static dictionary attacks, rule-based dictionary attacks, and personalized dictionary attacks. To show different threats in a more structural and user-friendly way, we render the whole 2-D space as a radar screen where the center of the screen shows the current password under evaluation and detected threats are shown around the center according to the level of risks: the higher the risk is, the closer the threat is placed to the center. Specifically, for weak passwords identified through dictionary attacks, we place each weak password on a circle whose radius is equal to the editing distance between the current password and the detected weak password. The threat related to brute force attacks is indicated by the password guessing entropy defined in Appendix A.2.1 of NIST SP 800-63-1 [5], which corresponds to the password strength shown by other existing password checkers. The password guessing entropy is shown on the x-axis of the 2-D radar screen and its distance to the center is proportional to its value. To make the system more reconfigurable, we design an interface allowing the end user to add new static dictionaries. A rule-based "smart" dictionary is also included to cover password composition rules which are combined with static dictionaries to detect more sophisticated but still weak passwords. Furthermore, personalized dictionaries are supported by collecting information from users' social network accounts (activated by the users manually by logging into their accounts). To differentiate different threats from each other, we use different shapes for different types of threats and different colors for different risk levels. A diagrammatic view of the design is shown in Figure 2.

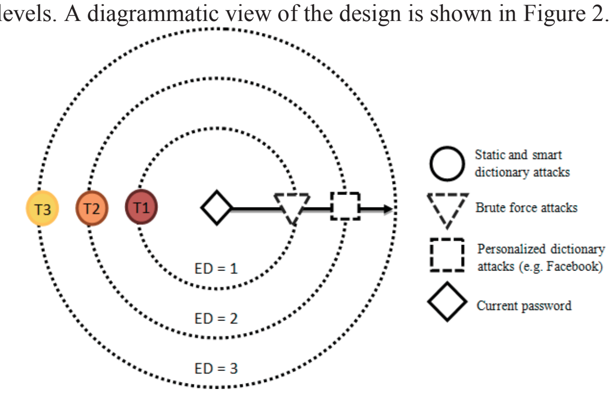

Figure 2. The visual user interface design of VPC.

The whole radar screen is also colored differently to visualize the overall password strength estimated from all threats detected: red denotes high risk, blue denotes medium risk, and green denotes low risk where no any threat is shown on the radar screen. Furthermore, to better educate users about all threats detected and provide tailored recommendations, each visualized threat is associated with a hidden tooltip control which will be made visible when the user moves her mouse on the threat.

3. PROTOTYPE IMPLEMENTATION

To demonstrate the feasibility of the above design of VPC, we implemented a prototype system using pure client-side web programming techniques including HTML5, CSS and JavaScript. While HTML5 is relatively new, the elements we used have been well supported by most modern web browsers.

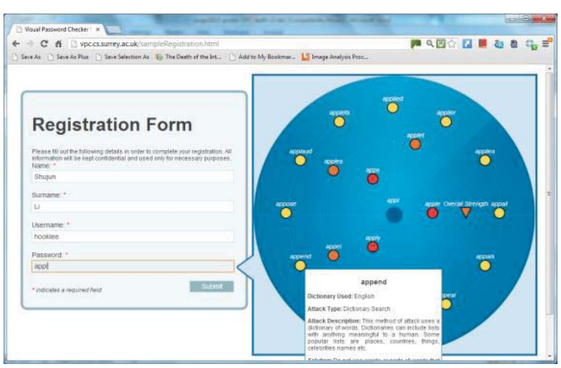

Figure 3. A snapshot of the VPC prototype in use.

One major implementation issue is how to make VPC very responsive meaning that the calculation needed should be very computationally light even with very large dictionaries. This was achieved by representing each dictionary as a tree where each valid word is labeled with a "final" flag if it is not a leaf node. A web-based tool was also developed to convert a textual static dictionary to the required tree format. When comparing the current password with entries in a dictionary, Levenshtein distance is used to calculate the edit distance. To limit the size of the 2-D radar screen, the prototype shows weak passwords with an edit distance up to 3. The function of personalized dictionaries is demonstrated by incorporating Facebook's API of single-sign on and personal information extraction. Figure 3 shows a snapshot of the VPC prototype in use on a registration page. The prototype system has been tested on five widely used web browsers and also on several mobile devices. To adapt to smaller screens of mobile devices, the 2-D radar screen can rescale automatically. The prototype is available for testing at http://vpc.cs.surrey.ac.uk.

4. ONGOING AND FUTURE WORK

We are currently working on or plan to work on the following tasks to further improve the VPC design and implementation:

- Adding support on more password composition rules such as 1) rules based on regular expressions, 2) more rules used by password crackers, 3) new rules extracted from real world.
- Identifying and visualizing multiple weak password segments (part of the password found in dictionaries).
- Adding strength estimated by invoking password crackers used by real-world password crackers.
- Replacing the NIST password guessing entropy estimator by a more accurate one such as that in [6].
- Adding password strength based on peer pressure [7].
- Showing weak passwords that would appear if the user deletes a few characters at the end (which are actually weak passwords appearing on previous screens).
- Improving the coloring scheme to allow smoother transition between different risk levels.
- A user study on the actual performance of VPC on real users.

5. REFERENCES

[1] The Password Meter. http://www.passwordmeter.com/.

[2] Intel Corporation. How Strong is Your Password? https://www-ssl.intel.com/content/www/us/en/forms/passwordwin.html.

[3] D. Goodin. Why Intel's "How Strong is Your Password?" site can't be trusted. Ars Technica, http://arstechnica.com/security/2013/05/why-intels-how-strong-is-your-password-site-cant-be-trusted/, 8 May 2013.

[4] 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 *Proc. CHI 2013*, pages 2379-2388, ACM, 2013.

[5] W. E. Burr, D. F. Dodson, E. M. Newton, R. A. Perlner, W. T. Polk, S. Gupta, and E. A. Nabbus. Electronic authentication guidelines. NIST SP 800-63-1, 2011.

[6] C. Castelluccia, M. Dürmuth, and D. Perito. Adaptive password-strength meters from Markov models. In *Proc. NDSS 2012*, Article Number 6-3, Internet Society, 2012.

[7] A. Sotirakopoulos. Influencing user password choice through peer pressure. Master's Thesis, University of British Columbia, LERSSE-THESIS-2011-004, http://lersse-dl.ece.ubc.ca/record/270, December 2011.