PSV (Password Security Visualizer): From Password Checking to User Education

Nouf Aljaffan[1,2], Haiyue Yuan[1], and Shujun Li[1]

[1] University of Surrey, Guildford, United Kingdom
[2] King Saud University, Riyadh, Kingdom of Saudi Arabia
{n.aljaffan, haiyue.yuan, shujun.li}@surrey.ac.uk

Abstract. This paper presents the Password Security Visualizer (PSV), an interactive visualization system specifically designed for password security education. PSV can be seen as a reconfigurable "box" containing different proactive password checkers (PPCs) and visualizers of password security information, allowing it to be used like a "many in one" or "hybrid" PPC. PSV can provide many new features that do not exist in traditional PPCs, thus having a greater potential to achieve its goals of educating users. Using purely client-side Web-based technologies, we implemented a prototype of PSV as an open-source software tool on a 2-D animated canvas. To evaluate the actual performance of our implemented PSV prototype against traditional PPCs, we conducted a semi-structured interview involving 20 human participants. Our qualitative analysis of the results showed that PSV was considered the most informative and recommended by most participants as a good educational tool. To the best of our knowledge, PSV is the first system combining different PPCs together for user education, and the user study is the first of this kind on comparing educational effectiveness of different PPCs (and PPC-like password security tools such as PSV).

Keywords: Password, security, visualization, password strength, password checker, password strength meter, password cracking

1 Introduction

Despite being older than half a century, passwords remain the mostly-used form for user authentication, which can be attributed to their simplicity (ease to use) and cost effectiveness. Because the pervasive use of passwords, they are frequently targeted in cyber attacks and many large-scale password leakage incidents have been reported especially in recent years [9, 22]. Password strengthening technologies such as password hashing and salting have been developed to provide more protection on passwords stored on the server side, but human users remain a weak link because they often choose weak passwords to compromise security for usability, thus making password cracking much more effective [11, 17, 31, 32].

In order to avoid the use of weak passwords by human users, many technologies have been developed to assist users and network administrators. Password

checkers are among the most widely-used technologies for this purpose. Password checkers are software tools used to check the strength of given passwords in order to detect and/or prevent use of weak passwords. There are two types of password checkers: proactive password checkers (PPCs) and reactive password checkers (RPCs). PPCs are client-side tools interacting with end users when they are creating passwords and giving immediate feedback on the user interface to inform users about the password strength. They are often combined with password policies so that known weak passwords are banned. RPCs are server-side tools performing regular scans of the password database by launching simulated password cracking attempts. Detected weak passwords by RPCs will be sent to network administrators and/or affected users for actions. In this paper, we focus on PPCs because they can offer more opportunities to educate end users directly.

A PPC needs to work with one or more password strength metering (PSM) algorithms, each of which normally returns a numerical or categorical value indicating the overall strength of a given password[3]. Many researchers use the term "PSM" (password strength meters) or simply "password meters" for PPCs, which can lead to confusion. In this paper, we used the term PSM for the underlying ("invisible") algorithms calculating password strength and PPC for the ("visible") software system with a clear user interface, empowered by one or more PSMs, to inform users about the strength of a given password.

PPCs are normally not designed for educational purposes, but can achieve such goals as a natural byproduct (e.g., by repeatedly using PPCs a user can naturally gain knowledge about password security). Insights learned from research work on PPCs and PSMs [4,5,25–27] have suggested that educating users about password security and attacks is an important aspect to make PPCs more effective, but very few tools have been developed and evaluated for this purpose.

This paper tries to fill the gap between password checking and user education by presenting Password Security Visualizer (PSV), an *interactive* visualization system specifically designed for password security education. PSV extends the main concepts behind all PPCs to a *reconfigurable* "box" containing different proactive password checkers and other non-PPC tools for visualizing useful information around the security of a given password, where "reconfigurable" refers to the capability of adding new PPCs into and removing existing ones from the PSV "box". Although being designed as an educational tool, PSV can still be used like a normal PPC, with much richer information about the security of the given password. To some extent, in addition to being a password security education tool, PSV can also be seen a "many in one" or "hybrid" PPC.[4] At the user interface (UI) level, PSV can be designed in many different ways, two of which will be explained in this paper. Using purely client-side Web-based technologies, we implemented one possible PSV design as an open-source software tool on a 2-D animated canvas. We followed some educational principles to design and

[3] In principle, a PSM algorithm can return more than one value each representing a different aspect of the password strength. Such algorithms are however very rare.

[4] We originally developed PSV as Visual Password Checker (VPC) [10], which was later extended/renamed to be more education-oriented rather than yet another PPC.

implement PSV, so we hoped it can a greater potential to achieve its goals of educating users. To evaluate the actual performance of the PSV prototype, we conducted a semi-structured interview with 20 human participants. Since there are not many other password security education tools and PPCs do have a side feature of educating users about password security, we decided to compare our PSV prototype's performance with three different designs of existing PPCs. Our results suggested that our PSV prototype was considered the most informative tool for educating users about password security.

The rest of this paper is arranged as follows. In the next section we present related work on PPCs. Section 3 discuss design considerations of PSV, and Section 4 gives details on the web-based PSV prototype system. In Sec. 5 we explain how we conducted the semi-structured interview and analyzed the results. The final section concludes the paper with further discussions and future work.

2 Related Work

PPCs can be traced back to research work conducted in the early 1990s [11, 15]. Nowadays PPCs have become ubiquitous on computer systems and websites, as a standard component of the password creation and update processes. The basic functionality of a PPC is to give *immediate* feedback on the strength of the password the user is entering so that the user can make a more informed decision on if the current password is strong enough to be used.

It has been observed that PPCs could influence users to choose stronger passwords [4, 5, 26], but users can also be confused by inappropriate/inconsistent strength ratings given by different PPCs [3]. Much research [13, 16, 23, 30, 33] has therefore been done to develop more robust PSMs so that the estimated password strength matches the actual risk against password crackers better.

At the UI level, some studies [5, 28] have showed that the PPC UI design matters in terms of influencing users to create stronger passwords, and some designs could be more effective. The most common UI design is a (horizontal or vertical) 1-D bar (or segmented box) showing the estimated password strength score as a progress bar, a colored bar/box, and sometimes a very short textual description such as "weak" and "very strong" as well. Some PPCs also show a more detailed textual description (maybe visible only after a link/button being clicked), which can cover recommendations on how to improve the current password and password policies. Some PPCs choose to use different PSMs e.g. those based on peer pressure [23] and fear appeal [28], which also require the UI to be designed differently. Among all PPCs we are aware of, one PPC [24] is quite unique in displaying multiple 1-D bars, which show details about how the overall password strength score is calculated based on multiple sub-ratings. Although the multi-bar PPC is much more informative, Ciampa found out it is the hardest to understand compared with other simpler PPCs [5]. The general absence of *clear* feedback and *sufficient* information about the returned password strength scores in PPCs can leave users confused about why a password is given a specific

rating by a PPC thus let them choose to neglect PPCs and depend on their own subject judgments on passwords [4, 25].

Ciampa studied the effectiveness of four different UI designs on password feedback mechanisms in PPCs [5]. Besides a common 1-D bar PPC, he also examined (1) a dial reading based PPC [19], (2) a fear appeal based PPC [29], (3) the multi-bar PPC "The Password Meter" [24]. His results showed that the fear appeal based PPC is the most effective among all the four tested feedback mechanisms on influencing users towards stronger passwords. However, the majority of participants were observed preferring the multi-bar PPC, even though it was the hardest to understand. Ciampa also reported the need of supporting users with the required security level based on the used context.

Ur et al. conducted a comparative study on PPCs used by 14 popular websites in 2012 [26]. They found out that most PPCs studied have a simple 1-D bar based UI design. They also found out that different PPCs' appearances did not have major effect on either users' attitudes or password arrangement. Although using PPCs did motivate users to creating longer passwords, which were not observed to be less memorable, users often did not have a clue about the reason behind ratings given by PPCs, which might cause confusion and mislead them when improper PSMs are used. They also found out that participants had tended to select weaker passwords when they became frustrated, and lost trust in the PPC. Similar observations around the psychological phenomenon "frustration" and "discomfort" were also reported by Haque et al. in a 2014 study [8].

Two more recent studies [25, 27] suggested that many users do have prior knowledge on how to strengthen their passwords, but they do not always follow the knowledge to create strong passwords in real world. One study [25] further suggested that this knowledge-behavior gap may be the result of neglecting to educate users about different attacks to passwords.

Furnell's study [6] revealed great inconsistencies among PPCs on 10 popular websites, and the password composition recommendations given by those websites were largely unclear and insufficient to guide users. The same observations were reported by de Carné de Carnavalet and Mannan in their work [3], in which they examined 13 PPCs deployed at 11 widely-used web services.

Komanduri et al. proposed a system called Telepathwords [12], which predicts most likely weak passwords based on the current password as the prefix and show them to alert users about such choices (since guessable passwords are weak). Telepathwords is not a PPC per the standard definition, but it show the security of the current password in a different way to guide users. They reported that the quality of passwords created using Telepathwords were higher than a number of PPCs they used for comparison. However, although users found that the feedback given by Telepathwords was helpful, many of them also reported it being difficult and annoying to use. This again highlighted the difficulty of designing good password security tools.

Some recently-reported personalized attacks on passwords [13,31] imply that PSMs and PPCs need to be personalized and contextualized. This is also echoed by Loge et al.'s work on a PPC for Android unlock patterns [14], in which they

observed that the password strength could be influenced by individual features such as age and gender.

3 Password Security Visualizer (PSV): Design Considerations

Our overall aim for PSV is to help enhance users' overall understanding of password security, based on what we have learned from existing PPCs and other password security tools with an educational effect. This lets us to reflect about what users truly need if we want to educate them about password security, eventually leading us to design PSV as a system going beyond password checking. Our main design goals for PSV include: (1) to help users gain more knowledge and have less confusions on *all* aspects of password security, including but not limited to password strength, (2) to highlight the *complexity* of password security by *externalizing* inconsistencies between different PPCs and more advanced attacks on passwords; (3) to *engage* users actively so that the process of learning is enjoyable, (4) to produce an *open* system that can be easily executed and customized by users on different platforms.

To achieve those design goals, we decided to follow some well-established design principles to design and implement PSV. In the following, we will discuss those design principles, which will be followed by two example designs and a discussion on some key supporting algorithms running in the background.

3.1 Design Principles

For designing PSV, we followed a number of widely-recognized principles across different application domains [1, 21], including cyber security [34]. Here, we explain all these principles and discuss how we considered them for PSV.

Informative feedback: This principle aims to provide users with essential and sufficient information to make more informed decisions [21]. This has been observed for many simple PPCs where users only see a single rating of the given password without any further information on why the password is rated as such and what to do to improve. Therefore, supporting users with more informative feedback could help raise their awareness on password security and correct any misconceptions, which in turn will help them to make better security-related decisions such as choosing a stronger password.

The informative feedback PSV can provide include different aspects of password security such as the following (but not limited to these) categories: (1) basic password attributes such as length, types of characters used, and structural information e.g. repeated patterns or character transformation rules, (2) risks against simple and advanced dictionary-based attacks, and (3) an overall password strength like what is given by a typical PPCs or PSM. For the third category, it will be beneficial to show estimates from multiple PPCs and PSMs to inform users about the complexity and limitations around the overall password strength estimation, thus educating them that they should not blindly follow an

arbitrary PPC or PSM. Being able to understand the complexity and limitations will also help them become less confused when they enter such inconsistent ratings of different PPCs/PSMs. PSV will thus include a number of Password Information Units (PIUs), each showing one aspect of password security.

While offering more information is in general helpful, we must not lose sight of implications for overloading users with too much information, which can harm their learning performance. After a certain point, information overload will occur, which consequently may prevent users from processing the provided information. This requires controlling the amount of information shown to users by adapting some strategies (e.g., filtering and zooming). Some other principles discussed below can help in this regard as well.

Visualization: Information visualization can facilitate exploring and understanding rich information at a glance to attract users' interest and motivate them to learn, so it has been advocated by researcher over textual contents for superior learning outcome [34]. Most PPCs already support some level of visualization, however, this needs further strengthening in PSV as there will be more PIUs and more interactions with users. One focus will be to minimize possible distractions caused by too many PIUs and visualization itself.

Segmentation and contiguity: Both principles can help to manage information being presented by reducing its complexity, thus helping users to understand the information better. Segmentation is about breaking information into small chunks [34]. This may involve grouping related information into different units. Contiguity is about keeping related information near to each other to maintain a smoother information flow, which can help users to achieve a better comprehension of the information presented [34]. For PSV, the segmentation principle is naturally done by grouping information into PIUs. We will need to consider how to map each PIU to visual features such as shape, size, orientation, etc. Figure 1 shows two examples of how the combination of shapes and orientations can be used to map various PIUs. The contiguity principle can be applied by providing more detailed information about each PIU (e.g., using a pop-up tooltip) while the user is interacting with the PIU. The additional information should be placed close to the PIU of interest so the information contiguity is maintained.

Signaling: This principle is about drawing users' attention to significant information only if it is necessary, which can help enhance users' learning performance [34]. For PSV, different visual features can be used to signal important information in each PIU. Information can be signaled by many distinct visual features (e.g., more prominent color, unique shape, larger size, animation, change of styles, etc.). For example, different icons or shapes can be used to indicate different information categories of a PIU, and a PIU's location relative to a reference can signal a specific level of risk. Such signaling can help users to recognize important information more quickly. Another example is about using animation: a PIU can move smoothly from an old location to a new one once its risk level changes. This interactive visualization could help raise users' awareness on such risk changes w.r.t. any changes to the password being evaluated, thus achieving a better understanding of how password security risks are estimated and why.

Interactive and immediate feedback: It is known that providing interactive and immediate feedback to users can foster their learning performance [21, 34]. Immediate feedback can help to engage users via giving them a quick chance to reflect on what they just learned [34], and interactivity would allow users to absorb complex concepts and enjoy the learning process more. All PPCs have this principle built-in since the password strength estimate is always updated immediately when any change to the current password is made. For PSV, we can provide users with more interactive and immediate feedback by drawing users' attention to important security issues beyond the password strength estimate. For example, the interactive fear appeal idea proposed in [28] can be used to warn users about potential risk of password attacks immediately after a weak password is detected against some specific attacks. The risk level can be visualized by a number of "negative" icons such as skulls to achieve the fear appeal effect.

Reconfigurability and personalization: Further information enrichments are also obtainable from allowing the system to be easily reconfigured. Such reconfiguration can allow users to create a personalized space to enhance their learning experience and gain more relevant knowledge [1]. As far as we know, all existing PPCs are designed to offer the same information to all users, which cannot adapt to different users' needs. To support reconfigurability and personalization, special UI elements and lower-level programming interfaces should be introduced in PSV to allow easy addition and removal of PIUs and other supporting components (e.g. PPCs, PSMs, password dictionaries and personal information), and also easy modification of the behavior and look of each PIU and any supporting algorithm. Different levels of reconfigurability and personalization can be supported, ranging from simple information filtering to customization of how an individual PIU or component looks/works and to even completely change of the look or working mechanism of the whole system.

Portability: This principle is about the need to make a system more available when users are moving across devices and platforms. An installation-free system that can run cross different platforms will be ideal. For PSV, a natural choice is to implement it as a web-based system based on pure client-side technologies (HTML, CSS and JavaScript) so that any computing device and OS with a standard-compliant web browser can allow the user to use PSV.

3.2 Two Example Designs

Following all the design principles discussed above, we can have many different designs of PSV. To accommodate more information and enrich interactions with users, it is necessary to move from the simple 1-D bar based design of most PPCs to a large space for visualization. In other words, we need to use a 2-D or a 3-D space to show a number of visualized PIUs. The space should have a layout easy for reconfiguration and personalization. To balance informativeness and information overload, the information shown in PSV can be put into several layers and visual metaphors can be used to invite users to interact with each PIU to get more detailed information related to the PIU. Since working with a

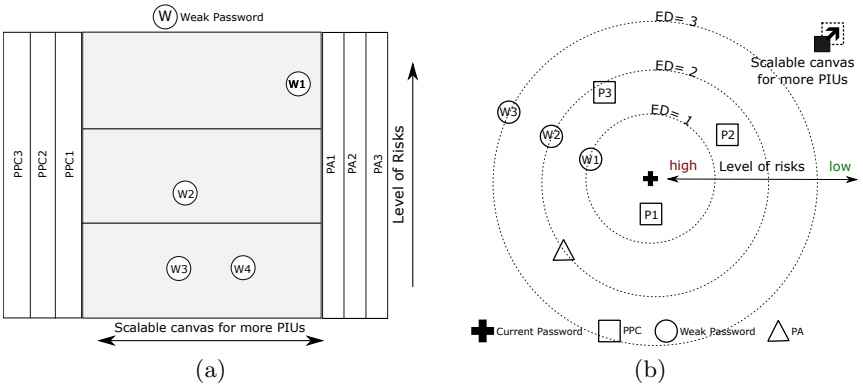


Fig. 1. Two example designs of PSV on a 2-D canvas, which is divided into (a) horizontal bars and (b) concentric circles representing different levels of security risk.

2-D space is easier and requires less computation, we decided to adhere to 2-D designs but will consider extensions to 3-D spaces in future.

Two example designs of PSV on a 2-D canvas are shown in Fig. 1. In both designs, the following groups of PIUs are included: 1) a number of PPCs showing the overall strength of the current password; 2) a number of PIUs showing basic password attributes (PA); 3) a number of weak passwords closer to the current password. Each weak password is visualized as an icon with a negative meaning following the fear appeal concept (e.g., a skull) and located at a position proportional to the edit distance (ED) [18] between the weak password and the current password. The number of weak passwords around the current password can be used as a proxy of the level of risk against dictionary-based attacks: the closer a dictionary entry to the current password and the more such entries are around, the more risky the current password is. The edit distance between a weak password and the current password can be related to a hybrid password attack which combines a dictionary-based attack with a simple brute force up to a number of character changes. The whole canvas can be extended to accommodate more PIUs easily, and removing or relocating existing PIUs is easy as well.

3.3 Supporting Algorithms

Different PIUs in PSV require a range of supporting algorithms which include at least the following groups.

PSMs: PSV can include a number of PPCs as PIUs and as mentioned before each PPC needs to work with one or more PSMs.

Password dictionary handling algorithms: PSV will support dictionary-based attacks so some algorithms will be needed to read and search in one or more dictionaries. A trie-based data structure can be used to efficiently store dictionaries and to accelerate the search process. A major subset of algorithms in this group

are for detecting weak passwords with a specific edit distance. Another subset of algorithms are for calculating the edit distance between two given strings.

Algorithms linking PSMs with PIUs: For some designs of PSV, the location of a PPC (as a PIU) is used to signal the password strength estimated (e.g., in the second design shown in Fig. 1). In this case, some algorithm will be needed to translate the password strength estimated to a location in the visualized space.

Algorithms for selecting and positioning of PIUs: Since multiple PIUs are displayed in a limited space, some algorithms are needed to decide what PIUs to show (how many) and where. Dynamic adjustment to some PIUs (e.g., reducing the size of a PIU or rotating it) may also be considered. These algorithms need to consider prioritization and randomization when not all PIUs can be shown due to limited space.

Parallelization and pre-computation algorithms: To ensure immediate feedback to users, the visualization of all PIUs needs to be fast enough to catch up with the typing speed of the user, even on relatively less powerful computing devices (e.g., smart phones). This requires most time-consuming computation to be done in an asynchronous manner (e.g., using HTML5 workers and AJAX), and be parallelized as much as possible. Pre-computation should be included, e.g., when a new character is added into the current password, each dictionary trie does not need to be searched from scratch, but from the last visited node.

4 Our PSV Prototype

We implemented a prototype of the second example design of PSV shown in Fig. 1(b). This prototype is developed using pure client-side web technologies including HTML5, CSS and JavaScript, which makes the prototype highly portable. We also made a simple interface for the PSV prototype to be incorporated into password creation/update pages of any HTML5-ready websites. The prototype can be found online at `http://passwords.sccs.surrey.ac.uk/PSV/`. In this section, we describe how we implemented the front-end and back-end parts of the prototype.

4.1 Front-end UI

The PSV prototype includes three groups of visual elements: a 2-D canvas, a configuration panel and a number of PIUs.

2-D canvas and overall look. A 2-D canvas is used as the container of PIUs. Figure 2 shows a screenshot of the PSV prototype's canvas whose background is rendered as an active radar with a rotating beam "scanning" for security concerns constantly (indicating the working status of PSV). The radar canvas as a visual metaphor matches the cyber security context well, which was the main reason why we decided to go for the second example design of PSV. The center of the radar canvas represents the current password and a number of (three as a default value, which is reconfigured) concentric circles are drawn

to accommodate all PIUs. The three concentric circles allow us to locate weak passwords with a particular distance with the current password (following the segmentation principle). Other PIUs (mainly PPCs) are mapped to any point from the radar center to the largest circle linearly so that the distance to the center represents the level of risk. A PIU will disappear if the risk is considered lower than a threshold (the value corresponding to the largest circle) so that it is unnecessary to show it any longer. From a user's perspective, while he/she is entering a password the radar canvas is dynamically updated with immediate feedback (via relevant PIUs), and the task of defining a strong password is to remove as many (ideally all) PIUs out of the radar so that no risks are visible (i.e., high). When the task is not to define a password, the user can play with the system by entering different passwords to learn more about password security. The design allows easy reconfiguration and personalization as a PIU can be easily added to or removed from the 2-D canvas. Each PIU's look and settings can also be configured separately or as a group (e.g., one can refine how a PPC is located by introducing a new linear or nonlinear mapping between the password strength estimate and the distance to the radar center). For three example passwords, Figure 3 shows how the whole PSV's UI looks like.

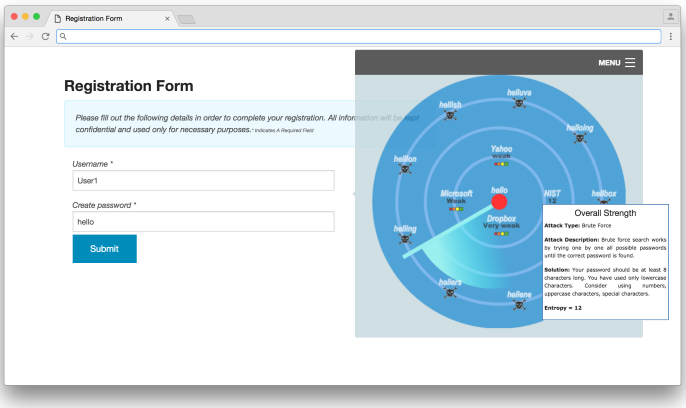
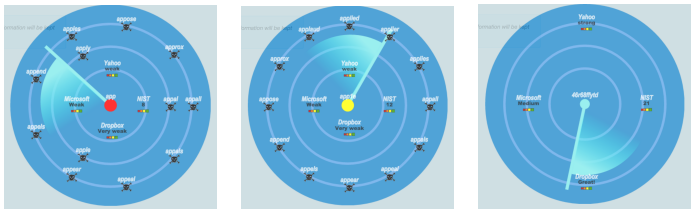


Fig. 2. The screenshot of an example user registration page of our PSV prototype.

PIUs. There are four different types of PIUs we include in our PSV prototype. Password attributes are not currently included because we found them least useful for user education purposes. We may add some in future versions.

The first type is the center of the 2-D canvas. As mentioned above, the center represents the current password. We use a small circle filled with a specific color to visualize three different states: light blue (normal), red (the password itself is a weak password), yellow (the password contains at least one weak password



(a) `app` (b) `app1e` (c) `46r68ffytd`

Fig. 3. Screenshots when three passwords were being entered into our PSV prototype.

segment). On top of the small circle the current password is shown in clear. We do not hide the password since PSV is designed as an educational tool. If the PSV is used as a PPC, the password can be simply removed or asterisks are shown as usual.

The second type covers PPCs. The current version incorporates four PPCs based on the common 1-D bar design: a PPC we developed based on the NIST password entropy [2] as the underlying PSM, the open-source password checker zxcvbn (which has been deployed by Dropbox) [33][5], the PPCs used by Microsoft and Yahoo! (for which we implemented our own versions). Note that the four are just used as examples and more PPCs can be added easily.

The third type covers weak passwords signaling the risks against dictionary-based attacks. We use a skull icon by following the fear appeal concept used by some PPCs such as those in [28]. Our prototype considers three different types of dictionary attacks to detect weak passwords related to the current password: 1) *naive* dictionary attack where each entry is checked as is, 2) *smart* dictionary attack where some common character transformation rules are considered, and 3) *targeted* dictionary attack where the user's personal information is used to build a small personalized dictionary. As a demonstrator, the targeted dictionary attack currently gets the user's first and last names by asking them to log into his/her Facebook account via the Facebook API. This can be extended to cover more personal information such as what was used in [13, 31].

The last type covers tool-tips that are shown when the user moves mouse over any PPC or weak password. Such tool-tips provide more detailed explanation to the corresponding PIU in order to provide more information about the risks of concern and guidance on how to reduce such risks. A unique part of the information shown on each tool-tip is about weak password segments, which are highlighted using different colors so that users are encouraged not to include any dictionary entries in their passwords. This can educate users about attacks combining multiple dictionaries. In addition, when a character transformation is applied to match a dictionary entry, the tool-tip will highlight the transformation to inform users about the risks of smart dictionary attacks.

[5] We incorporated an older version of the PPC zxcvbn downloaded from `https://github.com/dropbox/zxcvbn`.

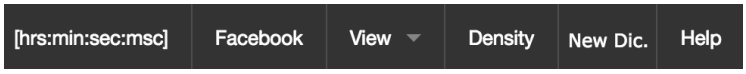


Fig. 4. The menu bar of our PSV prototype.

Configuration panel. To support reconfigurability and personalization, we also created a configuration panel as part of our PSV prototype on the top of the 2-D canvas. The configuration panel has two versions, one is shown in Fig. 4 for a typical layout on a PC, and a more mobile-friendly version as shown in Fig. 2 which breaks down the menu items into smaller items. The configuration panel empowers the user to make the following changes to the behavior and look of the PSV prototype.

Information filtering: The panel provides two ways to filter information shown on the 2-D canvas: a slider enabling dynamic control of the number (i.e., density) of weak passwords shown on the canvas, and a number of menu items to switch some types of PIUs on or off which includes indirect control via enabling or disabling existing password dictionaries and password attacks.

Adding new dictionaries: The PSV prototype allows users to add their own dictionaries into the system. This include personalized and normal dictionaries through "Facebook" and "New Dic" menu items, respectively. Normal dictionaries added will be stored in the system and can be enabled/disabled as built-in dictionaries, while the personalized dictionary is only accessible in the memory after the user logs into his/her Facebook account and will be released once he/she logs out.

4.2 Supporting Algorithms

Our PSV prototype is supported by some underlying algorithms for different purposes, which can be categorized based on five steps of the whole information processing chain: data storage, creation of candidate PIUs, positioning of PIUs, selection of PIUs, and visual presentation of selected PIUs. These steps are explained briefly below.

Data storage. Since multiple dictionaries are used in PSV, we need an efficient data structure and corresponding algorithms for creating and modifying dictionaries in the selected data structure. For our PSV prototype, we decided to use the succinct trie data structure implemented by Hanov [7]. A segment of such a trie can be seen in Fig. 5(a), where red nodes represent dictionary entries (concatenating all letters from the root node sequentially).

For personalized dictionaries, our PSV prototype currently extracts the user's first and last names from his/her Facebook account (after login), which are stored in the volatile memory and deleted permanently once the user logs out.

Creation of candidate PIUs. To create candidate PIUs that can be further selected for visualization, some algorithms are needed to produce information

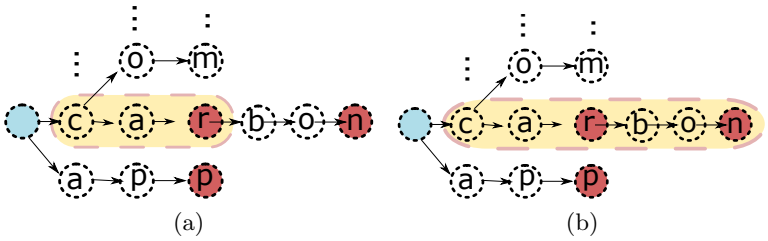


Fig. 5. The process of searching for weak passwords in a dictionary, where the password given is "car" and two detected weak passwords are "car" and "carbon".

needed by all candidate PIUs. Information needed for the current password PIU is straightforward, so we ignore it here and focus on other three types of PIUs.

Detection of weak passwords: An algorithm was developed to search through all enabled dictionaries to detect weak passwords whose edit distance from the current password is not greater than 3. In our prototype, we used Levenshtein distance as the edit distance since it is the most common metric used [18]. An example of the searching process is shown in Fig. 5. The results are stored as an array in which each element represent a weak password. We implemented multi-threading capability using HTML5 Web workers to improve performance of the searching process and to avoid blocking the main user interface.

Password strength metering (PSM): To visualize any PPC, the underlying PSM has to be executed on the current password. For our PSV prototype, there are four PSMs each serving one PPC. A PSM produces either a numeric value such as an entropy value or an ordinal value (among three or four different levels) to represent the strength of a given password.

Tool-tip generation: For each weak password and PPC PIU, a tool-tip object is also created to contain more detailed information and guidance to users.

Positioning of PIUs. One algorithm is needed to map each PIU type to a specific position on the 2-D canvas. For weak passwords, they can be naturally mapped to one of the three circles based on their edit distance from the current password. For PPCs, this will depend on the format of the password strength value: 1) if the underlying PSM returns an ordinal value then the PPC can be naturally mapped to one of the three circles as well (outer circles correspond to stronger passwords); 2) if the underlying PSM returns a numeric value like an entropy then the PPC is linearly or nonlinearly mapped to a position on a radial line starting from the center of the 2-D canvas, where the most outer circle will be set to correspond to a specific value considered as "very strong".

Selection of PIUs for visualization. Not all candidate PIUs are actually visualized since the 2-D canvas has a limited space and when a specific risk drops below a threshold we do not need to show it. For PPCs and weak passwords, they will disappear if their positions go beyond the most outer circle. Tool-tips

are always hidden since showing them will make the canvas too crowded, instead, one such tool-tip is shown dynamically when the user moves mouse over a specific PIU. The maximum number of PIUs shown is automatically calculated based on the size of the canvas. The configuration panel also allows the user to tailor the number of weak passwords which will also influence what PIUs are selected.

Visual presentation of selected PIUs. Each PIU type needs an algorithm to do the actual visualization. This may involve re-positioning selected PIUs, e.g., re-distributing all weak passwords with the same edit distance uniformly on the corresponding circle to make them look better, and moving some PIUs around to avoid conflicts with one or more neighboring PIUs.

5 Semi-structured Interviews

Semi-structured interviews were conducted (by the first co-author of the paper, referred to "the researcher" hereinafter) to investigate the efficacy of our PSV prototype on educating users about password security, compared with three traditional PPCs. Our main goal is to demonstrate PSV as a superior tool for password security education. This user study was reviewed by the University of Surrey's University Ethics Committee (UEC) and a favorable ethics opinion (FEO) was secured.

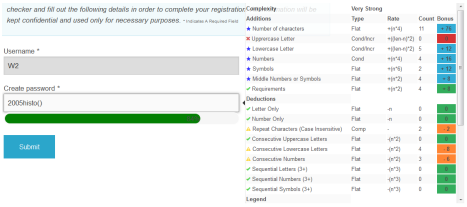
To align the UI of our PSV prototype and the three traditional PPCs so that any differences we observed should be only about the PSV and PPCs themselves, we designed a uniform login page with four different variants each of which uses a different password security/checking system. The three traditional PPCs we used include: 1) zxcvbn – a PPC based on the most common 1-D color bar design and the widely-used zxcvbn as the underlying PSM [33], 2) PM – the multi-bar based PPC called "The Password Meter" [24], 3) IFA – the interactive fear appeal based PPC proposed in [28]. We implemented our own versions of the three PPCs to ensure the consistent look of the overall login page. Figure 6 shows UIs of the three PPCs we implemented.

5.1 Interview Design

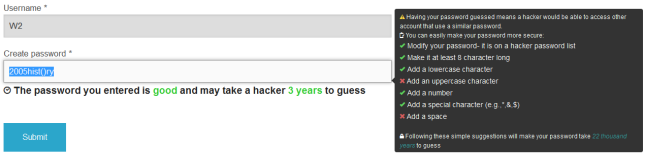
Although being an interview type user study, participants needed to give subjective opinions on password security/checking tools they might not have any prior knowledge so the user study also involves a short testing session for each tool. We also collected some basic demographic information about participants at the beginning using a questionnaire: age, gender and educational background. The whole session was conducted on a one-to-one basis to avoid interference between participants. The interviews were audio-recorded for further analysis, which later was deleted after being transcribed. Each participant spent around an hour to complete the whole session and was compensated £10 for their time.

In the testing sessions, each participant was asked to play the role of an imaginary security consultant to examine each of the four tools by doing the

(a) zxcvbn



(b) PM



(c) IFA

Fig. 6. The UI screenshots of the three PPCs used in our user study.

following for no less than 5 minutes: 1) trying a number of passwords given by the researcher and of their own choice; 2) paying attention on distinct information shown by each tool; 3) trying to understand the information shown; 4) making notes on different information shown to prepare for the interview with the researcher. Participants were encouraged to interact actively with the researcher during the assessment tests to simulate real-world scenarios where a security consultant will normally interact with the vendor of a candidate tool to get more information about it. Participants were offered to have a break between testing sessions, but none opted to have one. To minimize the bias caused by participants' own prior experience with any of the tested tools and to give participants a big picture of what the study is about, the four tools were introduced to the participants beforehand by the researcher.

The actual interview took place after each participant finished all the four testing sessions. The researcher asked each participant a number of questions around the four password security/checking tools to gather his/her subjective opinions on different aspects of those tools. When a participant asked for clarification on any tool, the researcher also provided needed information. Participants were not told that the PSV tool was developed by us, although at the end of the interview some asked the researcher if we developed some of the tools.

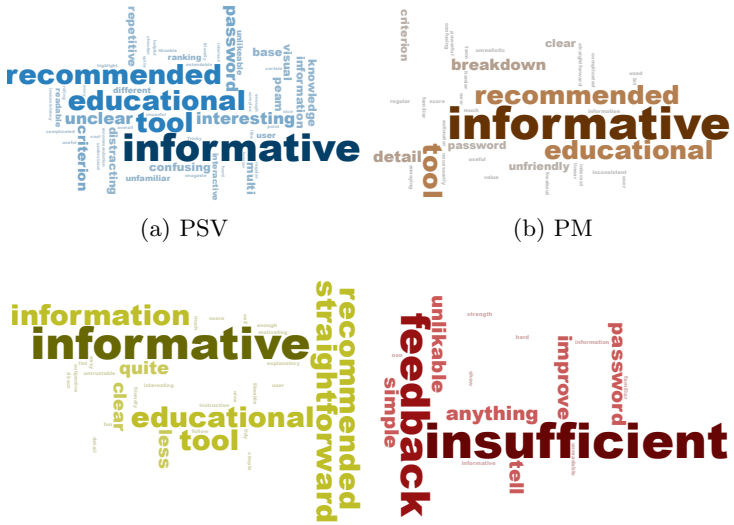
5.2 Participants

We recruited 20 participants using posters and the online research participation system (SONA) of School of Psychology, University of Surrey. The gender ratio

was not controlled: we got 14 participants and 7 male. The participants were in the age range of 19 to 45, with a median age of 22. Most participants were students from different subjects: psychology (25%), business (30%), engineering (25%), and others (15%). None of them had a strong knowledge on computer science or computer security. 70% of them are undergraduate students and 25% of them are post-graduate students. One participant worked in the University of Surrey as an administrative assistant.

5.3 Results

PSV as an educational tool: In our interview, we collected information about the most educative password checkers perceived by participants. We asked them questions about their newly acquired knowledge after testing the four tools. Figure 7 shows what all participants collectively said about each tool as a word cloud. Many participants found the zxcvbn PPC is the least educative, while the PSV and the PM PPC are the most informative ones. All participants reported that they had gained some new knowledge from PSV and the PM PPC. Many of them found that the PSV directly highlights distinct strategies used for guessing passwords and possible inconsistencies among different PPCs, which they found interesting due to the richer information presented in a visual manner. As a comparison, many felt that they had learned about more concrete new rules to improve password strength from the PM PPC.



(a) PSV (b) PM

(c) IFA (d) zxcvbn

Fig. 7. Mostly highlighted words for the PSV prototype and the three PPCs, each shown as a word cloud (generated by the online tool WordSift [20]).

At the beginning of the interview, most (18, 90%) participants failed to identify that the PSV as the most informative password checkers according to their understanding of informativity (see Table 1). Although they agreed that the PSV could provide a lot of information but they did not believe such information is all useful, and the majority felt that the PM PPC is the most informative tool. However, after explaining different components of the PSV and the PM PPC with greater details to participants, almost all participants were converted to articulate that the PSV is the most informative tool. A few participants remained their original opinion that the PM PPC is the best, based on the argument that their subjective judgments match the outcomes from the PM PPC better. Note that no participants asked for more explanation on the zxcvbn and IFA PPCs since they are simpler and more straightforward.

Table 1. Participants' votes on the most informative tool for password security education, before and after more details on the PSV and the PM PPC were given.

Password Tool	Before	After	Converted
zxcvbn	0	0	0
IFA	5	0	-5
PM	13	1	-12
PSV	2	19	17

The results on the PM PPC are not totally unexpected since it is indeed the most informative PPC among the three tested. The results should not be interpreted negatively against PSV because as a container of PPCs the PM PPC can also be added to the PSV canvas (which we plan to do in future versions).

We also asked participants which tool (if only one can be selected) they would recommend to their "customers" (average users, normally not security professionals) for self-learning password security. 11 participants (55%) preferred the PSV over the three PPCs, 6 selected the PM PPC, and the remaining 3 selected the IFA PC. None of the participants recommended the zxcvbn PPC as it does not provide enough feedback to users. Some participants explained that they did not recommend the PSV mainly because they felt the PM PPC is easier for average users to understand.

PSV as a PPC: Although our main aim is to measure effectiveness of the PSV as a password education tool, we also gathered information on to what extent PSV can be used as a PPC. However, none of the participants considered the PSV a good PPC. The majority reported that the PSV does not give an overall estimation of the password strength nor direct instructions for improving the current password. Yet, they reported the same problems for the zxcvbn PPC. This suggests that the PSV is probably not worse than the common 1-D bar based PPCs. Note that our PSV prototype has four such PPCs embedded.

When being asked which PPC is the best, six participants chosen the PM PPC, arguing that it provides more details about the single password strength

estimate which can help users to trust the PPC more. Three participants chose the IFA PPC, based on the argument that it provides straightforward instructions where users will be able to construct passwords faster. Participants who preferred the IFA PPC also mentioned that the PM PPC would be their second preferable PPC whose user-friendliness is considered worse than the IFA PPC.

Participants' trust on PPCs: We also asked participants to what extent they trust and rely on PPCs. 12 participants (60%) responded to this question. All except one responded that they have some level of trust on PPCs. One participant argued that such trust can be established only with familiar PPCs. One another participant mentioned he/she always trusts PPCs. Other participants mentioned that they would ignore a PPC if the PPC's password strength estimate is higher than their own subjective judgment. On the other hand, most participants said that they would make serious efforts to improve their passwords if a PPC gives a rating lower than their subjective judgment.

6 Conclusions and Future Work

This paper presents Password Security Visualizer (PSV), a new password security educational system and a prototype developed based on proactive password checkers. We conducted a semi-structured interview based on a number of testing sessions with 20 participants to validate the usefulness of the PSV prototype. The results of the user study showed that the majority of participants agreed that PSV is the most educative tools comparing to three traditional types of PPCs and would recommend it to average users as a self-learning tool on password security. Participants however were not convinced the PSV is a good alternative PPC considering its lacking an password strength estimate and direct instructions for improving passwords. More conversations with participants also revealed that most participants found PPCs useful but their perceptions vary on what PPC is preferred and when they will follow the ratings of a PPC.

Participants' responses revealed that the rich information provided by our PSV prototype was perceived somewhat negatively especially at the very beginning. Some participants seemed confused about what to do with so much information since the PSV does not give them a single piece of information (like what traditional PPCs do) which they can simply focus on. This negative feeling was significantly reduced after we provided clearer instructions on how the PSV should be used and highlighted its conceptual differences from traditional PPCs, thus suggesting that the tool may be better used with instructors. For self-learning purposes, the PSV can be reconfigured to adapt the system's features and its UI to each user's individual preferences and needs.

In our future work, we will study how to improve the current designs and implementation of PSV to make it more useful as both a user education tool and an alternative PPC. For instance, we may redesign the PSV so that fewer PIUs are shown to make the UI less crowded and complicated.

Acknowledgments. Nouf Aljaffan was funded by a PhD scholarship from the King Saud University, Kingdom of Saudi Arabia. Haiyue Yuan and Shujun Li were supported by the UK part of a joint Singapore-UK research project COMMANDO-HUMANS, funded by the Engineering and Physical Sciences Research Council (EPSRC) under grant number EP/N020111/1.

References

1. Beetham, H., Sharpe, R. (eds.): Rethinking Pedagogy for a Digital Age: Designing and Delivering E-Learning. Routledge (2007)
2. Burr, W.E., Dodson, D.F., Newton, E.M., Perlner, R.A., Polk, W.T., Gupta, S., Nabbus, E.A.: Electronic authentication guideline. NIST Special Publication 800-63-2 (2013)
3. de Carné de Carnavalet, X., Mannan, M.: From very weak to very strong: Analyzing password-strength meters. In: Proc. NDSS 2014. Internet Society (2014)
4. Carnavalet, X.D.C.D., Mannan, M.: A large-scale evaluation of high-impact password strength meters. ACM Transactions on Information and System Security 18(1), 1:1–1:32 (2015)
5. Ciampa, M.: A comparison of password feedback mechanisms and their impact on password entropy. Information Management & Computer Security 21(5), 344–359 (2013)
6. Furnell, S.: Assessing password guidance and enforcement on leading websites. Computer Fraud & Security 2011(12), 10–18 (2011)
7. Hanov, S.: Succinct data structures: Cramming 80,000 words into a Javascript file. Online document, `http://stevehanov.ca/blog/index.php?id=120` (2012), accessed on 10th February 2017
8. Haque, S.M.T., Scielzo, S., Wright, M.: Applying psychometrics to measure user comfort when constructing a strong password. In: Proc. SOUPS 2014. pp. 231–242. USENIX Association (2014)
9. Hunt, T.: `';--have i been pwned?` check if you have an account that has been compromised in a data breach. `https://haveibeenpwned.com/` (Last accessed on 11 February 2017)
10. Kafas, K., Aljaffan, N., Li, S.: Poster: Visual Password Checker. Presented at SOUPS 2013, 2-page summary available online at `https://cups.cs.cmu.edu/soups/2013/posters/soups13_posters-final19.pdf` (2013)
11. Klein, D.V.: Foiling the cracker: A survey of, and improvements to, password security. In: Proc. USENIX Security '90. pp. 5–14 (1990)
12. Komanduri, S., Shay, R., Cranor, L.F., Herley, C., Schechter, S.: Telepathwords: Preventing weak passwords by reading users' minds. In: Proc. USENIX Security 2014. pp. 591–606. USENIX Association (2014)
13. Li, Y., Wang, H., Sun, K.: A study of personal information in human-chosen passwords and its security implications. In: Proc. IEEE INFOCOM 2016. pp. 1242–1254. ACM (2016)
14. Loge, M., Duermuth, M., Rostad, L.: On user choice for Android unlock patterns. In: Proc. EuroUSEC 2016. Internet Society (2016)
15. Ma, J., Yang, W., Luo, M., Li, N.: Anatomy of a proactive password changer. In: Proc. USENIX Security '92. p. 171184. USENIX Association (1992)
16. Melicher, W., Ur, B., Segreti, S.M., Komanduri, S., Bauer, L., Christin, N., Cranor, L.F.: Fast, lean and accurate: Modeling password guessability using neural networks. In: Proc. USENIX Security 2016 (2016)

17. Narayanan, A., Shmatikov, V.: Fast dictionary attacks on passwords using time-space tradeoff. In: Proc. CCS 2005. pp. 364–372. ACM (2005)
18. Navarro, G.: A guided tour to approximate string matching. ACM Computing Surveys 33(1), 31–88 (2001)
19. Neil's Toolbox: Password security tester. http://www.neilstoolbox.com/password-tester/ (Last accessed on 11 February 2017)
20. Roman, D., Thompson, K., Ernst, L., Hakuta, K.: WordSift: A free web-based vocabulary tool designed to help science teachers in integrating interactive literacy activities. Science Activities: Classroom Projects and Curriculum Ideas 53(1), 13–23 (2016), https://wordsift.org/
21. Shute, V.J.: Focus on formative feedback. Review of Educational Research 78(1), 153–189 (2008)
22. Solutions, V.E.: Verizon's 2016 data breach investigations report. http://www.verizonenterprise.com/resources/reports/rp_DBIR_2016_Report_en_xg.pdf (2016)
23. Sotirakopoulos, A.: Influencing User Password Choice Through Peer Pressure. Master's thesis, University of British Columbia, Canada (2011), http://lersse-dl.ece.ubc.ca/record/270
24. Todnem, J.: The password meter. http://www.passwordmeter.com/ (Last accessed on 11 February 2017)
25. Ur, B., Bees, J., Segreti, S.M., Bauer, L., Christin, N., Cranor, L.F., Deepak, A.: Do users' perceptions of password security match reality? In: Proc. CHI 2016. pp. 3748–3760. ACM (2016)
26. Ur, B., Kelley, P.G., Komanduri, S., Lee, J., Maass, M., Mazurek, M.L., Passaro, T., Shay, R., Vidas, T., Bauer, L., Christin, N., Cranor, L.F.: How does your password measure up? the effect of strength meters on password creation. In: Proc. USENIX Security 2012. pp. 65–80. USENIX Association (2012)
27. Ur, B., Noma, F., Bees, J., Segreti, S.M., Shay, R., Bauer, L., Christin, N., Cranor, L.F.: "I added '!' at the end to make it secure": Observing password creation in the lab. In: Proc. SOUPS 2015. pp. 123–140. USENIX Association (2015)
28. Vance, A., Eargle, D., Ouimet, K., Straub, D.: Enhancing password security through interactive fear appeals: A web-based field experiment. In: Proc. HICSS 2013. pp. 2988–2997. IEEE (2013)
29. Wales, M.: How secure is my password? https://howsecureismypassword.net/ (Last accessed on 11 February 2017)
30. Wang, D., He, D., Cheng, H., Wang, P.: fuzzyPSM: A new password strength meter using fuzzy probabilistic context-free grammars. Proc. DSN 2016 pp. 595–606 (2016)
31. Wang, D., Zhang, Z., Wang, P., Yan, J., Huang, X.: Targeted online password guessing: An underestimated threat. In: Proc. CCS 2016. pp. 1242–1254. ACM (2016)
32. Weir, M., Aggarwal, S., de Medeiros, B., Glodek, B.: Password cracking using probabilistic context-free grammars. In: Proc. IEEE S&P 2009. pp. 391–405 (2009)
33. Wheeler, D.L.: zxcvbn: Low-budget password strength estimation. In: Proc. USENIX Security 2016. pp. 157–173. USENIX Security (2016), https://github.com/dropbox/zxcvbn
34. Zhang-Kennedy, L., Chiasson, S., Biddle, R.: The role of instructional design in persuasion: A comics approach for improving cybersecurity. International Journal of Human-Computer Interaction 32(3), 215–257 (2016)