Persuasive Cued Click-Points: Design, implementation, and evaluation of a knowledge-based authentication mechanism

Sonia Chiasson, *Member, IEEE,* Elizabeth Stobert, Alain Forget, Robert Biddle, *Member, IEEE,* and P. C. van Oorschot, *Member, IEEE*

Abstract—This paper presents an integrated evaluation of the Persuasive Cued Click-Points graphical password scheme, including usability and security evaluations, and implementation considerations. An important usability goal for knowledge-based authentication systems is to support users in selecting passwords of higher security, in the sense of being from an expanded effective security space. We use persuasion to influence user choice in click-based graphical passwords, encouraging users to select more random, and hence more difficult to guess, click-points.

Index Terms—authentication, graphical passwords, usable security, empirical studies

1 Introduction

The problems of knowledge-based authentication, typically text-based passwords, are well known. Users often create memorable passwords that are easy for attackers to guess, but strong system-assigned passwords are difficult for users to remember [6].

A password authentication system should encourage strong passwords while maintaining memorability. We propose that authentication schemes allow user choice while influencing users towards stronger passwords. In our system, the task of selecting weak passwords (which are easy for attackers to predict) is more tedious, discouraging users from making such choices. In effect, this approach makes choosing a more secure password the path-of-least-resistance. Rather than increasing the burden on users, it is easier to follow the system's suggestions for a secure password — a feature lacking in most schemes.

We applied this approach to create the first persuasive click-based graphical password system, Persuasive Cued Click-Points (PCCP) [2], [3], and conducted user studies evaluating usability and security. This paper presents a consistent assimilation of earlier work [1]–[4] and two unpublished web studies, reinterprets and updates statistical analysis incorporating larger datasets, provides new evaluation of password distributions, extends security analysis including relevant recent attacks, and presents important implementation details. This systematic examination provides a comprehensive and integrated evaluation of PCCP covering both usability and security issues, to advance understanding as is prudent before practical deployment of new security mechanisms. Through eight user studies [1]–[4], [7], we compared PCCP to text passwords and two related graphical password systems. Results show that PCCP is effective at reducing hotspots (areas of the image where users are more likely to select click-points) and avoiding patterns formed by click-points within a password, while still maintaining usability.

The paper is structured as follows. Section 2 covers related authentication schemes and Persuasive Technology. Section 3 describes PCCP. Methodology and relevant details of the user studies are available in Section 4. Results of the usability evaluation are in Section 5. Section 6 examines the characteristics and skewed nature of the password distributions. Section 7 provides a security analysis against likely threats. Relevant implementation issues are addressed in Section 8. Section 9 offers concluding remarks.

2 Background

Text passwords are the most popular user authentication method, but have security and usability problems. Alternatives such as biometric systems and tokens have their own drawbacks [8]–[10]. Graphical passwords offer another alternative, and are the focus of this paper.

Click-based graphical passwords: Graphical password systems are a type of knowledge-based authentication that attempt to leverage the human memory for visual information [11]. A comprehensive review

• *All authors are from Carleton University, Ottawa, Canada. E-mail: chiasson@scs.carleton.ca*
Parts of this paper appeared earlier in publications [1]–[5].

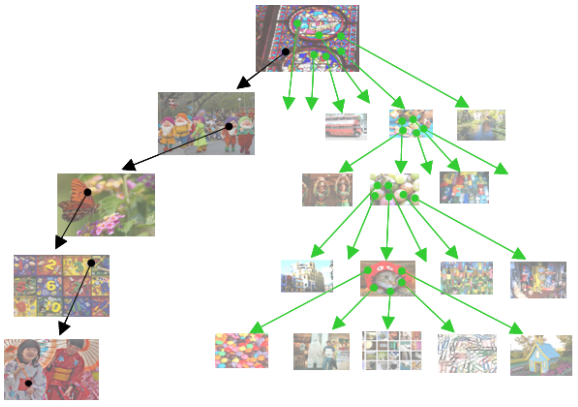


Fig. 1. A user navigates through images to form a CCP password. Each click determines the next image.

of graphical passwords is available elsewhere [12]. Of interest herein are *cued-recall click-based graphical passwords* (also known as locimetric [13]). In such systems, users identify and target previously selected locations within one or more images. The images act as memory cues [14] to aid recall. Example systems include PassPoints [15] and Cued Click-Points [7].

In PassPoints, passwords consist of a sequence of five click-points on a given image. Users may select any pixels in the image as click-points for their password. To log in, they repeat the sequence of clicks in the correct order, within a system-defined *tolerance square* of the original click-points. Although PassPoints is relatively usable [1], [15], [16], security weaknesses make passwords easier for attackers to predict. *Hotspots* [17]–[20] are areas of the image that have higher likelihood of being selected by users as password click-points. Attackers who gain knowledge of these hotspots through harvesting sample passwords can build attack dictionaries and more successfully guess PassPoints passwords [18], [19]. Users also tend to select their click-points in predictable *patterns* [5], [20] (e.g., straight lines), which can also be exploited by attackers even without knowledge of the background image; indeed, purely automated attacks against PassPoints based on image processing techniques and spatial patterns are a threat [21].

A precursor to PCCP, Cued Click-Points (CCP) [7] was designed to reduce patterns and to reduce the usefulness of hotspots for attackers. Rather than five click-points on one image, CCP uses one click-point on five different images shown in sequence. The next image displayed is based on the location of the previously entered click-point (Figure 1), creating a path through an image set. Users select their images only to the extent that their click-point determines the next image. Creating a new password with different click-points results in a different image sequence.

The claimed advantages are that password entry becomes a true cued-recall scenario, wherein each image triggers the memory of a corresponding click-point. Remembering the order of the click-points is no longer a requirement on users, as the system presents the images one at a time. CCP also provides *implicit feedback* claimed to be useful only to legitimate users. When logging on, seeing an image they do not recognise alerts users that their previous click-point was incorrect and users may restart password entry. Explicit indication of authentication failure is only provided after the final click-point, to protect against incremental guessing attacks.

User testing and analysis showed no evidence of patterns in CCP [5], so pattern-based attacks seem ineffective. Although attackers must perform proportionally more work to exploit hotspots, results showed that hotspots remained a problem [2].

Persuasive Technology: Persuasive Technology was first articulated by Fogg [22] as using technology to motivate and influence people to behave in a desired manner. An authentication system which applies Persuasive Technology should guide and encourage users to select stronger passwords, but not impose system-generated passwords. To be effective, the users must not ignore the persuasive elements and the resulting passwords must be memorable. As detailed below, PCCP accomplishes this by making the task of selecting a weak password more tedious and time-consuming. The path-of-least resistance for users is to select a stronger password (not comprised entirely of known hotspots or following a predictable pattern). The formation of hotspots across users is minimized since click-points are more randomly distributed. PCCP's design follows Fogg's Principle of Reduction by making the desired task of choosing a strong password easiest and the Principle of Suggestion by embedding suggestions for a strong password directly within the process of choosing a password.

3 Persuasive Cued Click-Points (PCCP)

Previous work (see above) showed that hotspots and patterns reduce the security of click-based graphical passwords, as attackers can use skewed password distributions to predict and prioritize higher probability passwords for more successful guessing attacks.

Visual attention research [23] shows that different people are attracted to the same predictable areas on an image. This suggests that if users select their own click-based graphical passwords without guidance, hotspots will remain an issue. Davis et al. [24] suggest that user choice in all types of graphical passwords is inadvisable due to predictability.

We investigated whether the system could influence users to select more random click-points while maintaining usability [2]–[5]. The goal was to encourage more secure behaviour by making less secure choices (i.e., choosing poor or weak passwords) more time-consuming and awkward. In effect, behaving securely became the *safe path-of-least-resistance* [2].

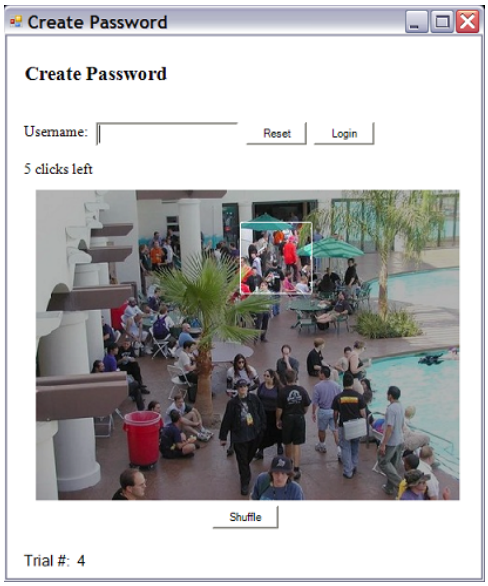


Fig. 2. PCCP Create Password interface. The viewport highlights part of the image. (Pool image from [25])

By adding a persuasive feature to CCP [7], PCCP [2] encourages users to select less predictable passwords, and makes it more difficult to select passwords where all five click-points are hotspots. Specifically, when users create a password, the images are slightly shaded except for a *viewport* (see Figure 2). The viewport is positioned randomly, rather than specifically to avoid known hotspots, since such information might allow attackers to improve guesses and could lead to the formation of new hotspots. The viewport's size is intended to offer a variety of distinct points but still cover only an acceptably small fraction of all possible points. Users must select a click-point within this highlighted viewport and cannot click outside of the viewport, unless they press the *shuffle* button to randomly reposition the viewport. While users may shuffle as often as desired, this significantly slows password creation. The viewport and shuffle button appear only during password creation. During later password entry, the images are displayed normally, without shading or the viewport, and users may click anywhere on the images. Like PassPoints and CCP, login click-points must be within the defined tolerance squares of the original points.

The *theoretical password space* for a password system is the total number of unique passwords that could be generated according to the system specifications. Ideally, a larger theoretical password space lowers the likelihood that any particular guess is correct for a given password. For PCCP, the theoretical password space is $((w \times h)/t^2)^c$, where the size of the image in pixels ($w \times h$) is divided by the size of a tolerance square (t^2, in our experiments, 19^2), to get the total number of tolerance squares per image, raised to the power of the number of click-points in a password (c, usually set to 5 in our experiments).

While it is beyond our present scope to establish an acceptable theoretical password space for authentication schemes, Florencio and Herley [26] suggest that theoretical password spaces of 2^{20} suffice to withstand online attacks. Whereas text passwords have very skewed distributions [27], resulting in an *effective password space* much smaller than the theoretical space, PCCP is specifically designed to significantly reduce such skews. Further design and implementation details of PCCP are discussed in Section 8.

4 Description of User Studies

We discuss eight different user studies (see Table 1), including three studies of PCCP [2], [4], two of PassPoints [5], [7], one of CCP [7], and two of text passwords [3]. We used the PassPoints, CCP, and text password studies as benchmarks where appropriate. The studies followed one of three methodologies intended to assess different aspects of the systems. Controlled lab studies collected baseline data, two-week recall studies stressed memorability, and web-based studies where participants logged in from home increased ecological validity. For example, in the PCCP Web study, 24 users had passwords for three accounts. They were asked to log in at 4 different times over the span of one week, resulting in 72 logins in total.

Most participants were university students from various fields. All were regular computer users comfortable with text passwords and a mouse. None took part in more than one study and none had previously used graphical passwords. Besides password tasks, participants completed a demographics questionnaire and a post-task questionnaire.

The lab and two week recall studies (Sections 4.1 and 4.2) used standalone J# applications for Windows. The 19-inch screen had a resolution of 1024×768 pixels. Consistent with earlier PassPoints studies [15], the images were 451×331 pixels, with tolerance squares 19×19 pixels, and passwords of 5 click-points, yielding a theoretical space of 2^{43} passwords, unless otherwise specified. No images were repeated between or within passwords for a given user.

The web studies (Section 4.3) were conducted with the MVP [28] web-based authentication framework. PCCP was again configured to use 451×331 pixel images, 19×19 tolerance squares, and 5 click-points. Since participants could log in from anywhere, screen size and resolution were not controlled.

In our studies we either asked users to pretend that these passwords were protecting important information or we gave users tasks on real websites. While we believe that this encouraged users to value their passwords, these were not high-value accounts and this may have affected user behaviour. We discouraged users from writing down passwords and did not allow them to write them down in our presence, but as with real-world systems, we had no way of

TABLE 1
Summary of eight studies. Numbers in parentheses are for the recall sessions.

Study Name	Duration	Number of Users	Pswds Per User	Trials
PCCP Lab	$1\times$	37	≤ 10	307
CCP Lab	$1\times$	57	≤ 12	505
PP Lab	$1\times$	41	≤ 17	581
PCCP 2wk	$2 \times 2wk$	82 (81)	6	462 (456)
PP 2wk	$2 \times 2wk$	32 (11)	6	192 (44)
Text 2wk	$2 \times 2wk$	34 (15)	6	204 (60)
PCCP Web	$4 \times 1wk$	24 (24)	3	184 (181)
Text Web	$4 \times 1wk$	21 (21)	3	138 (204)

TABLE 2
Parameters for six experimental conditions and number of users (N) in the PCCP 2-week recall study.

	w	h	Click-points	Condition Name	Password Space (in bits)	N
Small	451	331	5	S5	43	14
			6	S6	53	14
			7	S7	61	14
Large	800	600	5	L5	52	14
			6	L6	63	12
			7	L7	73	14

stopping them from doing so at home. Furthermore, we attempted to get a wide sample of users within the university setting and believe that the results apply to the broader population, but further studies would be needed to confirm generalizability.

4.1 Lab Studies

Lab studies consisting of one-hour sessions with individual participants were intended to evaluate usability and collect data on many images for initial security analysis. Participants were introduced to the system and instructed to pretend these passwords were protecting their bank information, and thus should select memorable passwords that were difficult for others to guess. Participants completed two practice trials (not included in the analysis) to ensure that they understood how the system worked. A trial consisted of creating, confirming, and logging on with a password, separated by a distraction task before login.

17 core images were used in all studies. Since PCCP and CCP required more images, 330 images (including the core 17) were compiled from personal collections and websites providing free-for-use images.

PCCP Lab [2]: This study had 37 participants who each completed up to 10 real (non-practice) trials, as time permitted. In total, data from 307 trials was collected. In addition to the general instructions, participants were told that the viewport was a tool to help them select more secure passwords, but that they could shuffle as many times as they wished to find a suitable click-point. The viewport was 75×75 pixels.

CCP Lab [7]: This study had 57 participants, who completed up to 12 trials for a total of 505 CCP trials.

PP Lab [1]: Here, 41 PassPoints Lab participants completed up to 17 trials, as time permitted. In total, 581 trials were included in this analysis.

4.2 Two Week Recall Studies

The main intention of the two week recall studies was to test long-term password memorability, look at the effects of multiple password interference, and collect information about the types of passwords created when users knew that they would need to recall them later. Each study was designed to strain memory by setting a difficult recall task so that differences between the schemes would be amplified.

Participants took part in two individual sessions, scheduled approximately two weeks apart. The sessions were 1 hour and 30 minutes long, respectively. In their first session, participants initially practiced creating and re-entering passwords for two fictitious accounts. The practice data was discarded and participants did not need to recall these passwords later. Next, participants created and re-entered passwords for six fictitious accounts (library, email, bank, online dating, instant messenger, and work). The accounts were identified by coloured banners at the top of the application window that included a unique icon and the account name. In the first session, the accounts were presented to all participants in the same order. In their second session, participants tried to re-enter these same six passwords in shuffled order.

PCCP used 465 images, including the 17 core images. Since participants only had 6 accounts and PassPoints has only one image per password, 6 of the 17 core images were used for the PassPoints study.

PCCP 2wk [4]: This study had 83 participants. Besides testing PCCP under its canonical configuration, we examined the effects increasing the theoretical password space by increasing image size and number of click-points per password. A between-subjects design was used, and participants were randomly assigned to one of six conditions (Table 2): *S5* (small image, 5 click-points); *S6* (small image, 6 click-points); *S7* (small image, 7 click-points); *L5* (large image, 5 click-points); *L6* (large image, 6 click-points); and *L7* (large image, 7 click-points). The small images were 451×331 pixels and the large, 800×600 pixels (standardizing to a 4:3 aspect ratio). Figure 3 shows the interface for the two image sizes. The small and large image conditions shared images resized to different dimensions. The viewport was 75×75 pixels.

The data was used in two separate analysis. First, we compared the *S5* condition to the other schemes as its configuration directly matched that of the other studies. Secondly, we compared the 6 experimental conditions to each other to investigate the effects of increasing the theoretical password space.

PP 2wk [3]: This study had 32 participants who created 192 passwords in total; not everyone completed

Fig. 3. User interface for password creation for the small and large image sizes in PCCP [4].

the second session. Session 1 was completed by 32 participants, 11 of whom completed the two-week recall session. Session 2 was added to the methodology after examining the initial results for multiple password interference. Participants recruited after this methodology change completed Session 2.

Text 2wk [3]: 34 participants took part in this study and created 204 text passwords. 15 participants completed the two week recall session. As in the above study, Session 2 was added after initial analysis of password interference and was only available to participants recruited after this methodological change.

The text password system enforced an 8-character minimum, with no other restrictions, giving a theoretical space of 2^{52}. While this exceeds that for the compared graphical password schemes, we knew that the effective password space for text systems is often significantly reduced by predictable password choices [27]. We thus expected weak text password choices and potential reuse of passwords across accounts, resulting in a significantly reduced memory load, and chose this larger theoretical password space to avoid an unfair memorability comparison.

4.3 Web Studies

The web-based studies tested the schemes in a more ecologically valid setting (i.e., users completed tasks on real websites over the course of a week from their own computers). We evaluated usability of the schemes in everyday usage and examined whether this affected user choice of passwords.

We conducted a one week study evaluating PCCP and text passwords as the authentication mechanisms on three websites. Participants initially had a one hour session where they received training on using the websites and the password system, and created accounts on the three websites. The accounts were for a photo blog about a local university campus, a blog with a different look-and-feel offering advice to first year university students, and a phpBB forum to discuss the best locations on campus for various activities (e.g., the best place to buy coffee). The websites were populated with real content to engage users realistically. In each case, participants' main tasks included logging on to comment on a specific blog post or forum thread. In the week following the initial session, participants received email asking them to complete further tasks. Two tasks were assigned on each of Day 1, Day 3, and Day 6. These tasks were similar to those completed in the initial session and could be completed from any web-enabled computer.

PCCP Web: 24 participants collectively completed 72 at-home recall trials. The system parameters were set to 451×331 pixel images, 5 click-points per password, a tolerance region of 19×19 pixels, and a persuasive viewport of 100×100 pixels. Passwords were encoded using Centered Discretization [29].

Text Web: This study included 21 participants who completed 204 at-home recall trials. The system required text passwords of minimum length 6, including at least one digit and one letter, which gives a minimum theoretical space 2^{36} passwords (more if longer passwords were chosen), counting both uppercase and lowercase letters. We reduced the password length from earlier studies based on Florencio and Herley's recommendations [26] for online usage.

5 Usability Evaluation

We evaluated the usability of PCCP through several performance measures. To place the results in context, we compared PCCP to the other authentication schemes tested under similar conditions.

Statistical analysis was used to determine whether differences in the data reflected actual differences between conditions or might reasonably have occurred by chance. A value of $p < .05$ is regarded as indicating statistical significance, implying less than a 5% probability that results occurred by chance.

We consider the following performance measures for memorability and usability [12]: login and recall success rates, times for password creation, login, and recall, and the effect of shuffling on success rates. Logins occurred during the initial lab session and tested shorter-term memorability, while recalls occurred either at-home or during a second lab session and tested long-term memorability. Where appropriate, the same measures are included for the PassPoints, CCP, and Text studies. The studies were conducted over a few

TABLE 3
Login and recall success rates across the eight studies, as percentages. Recall represents either at-home tasks or a second lab session. Values that are not applicable are identified with dashes.

	PCCP Lab	CCP Lab	PP Lab	PCCP 2wk All	PCCP 2wk S5	PP 2wk	Text 2wk	PCCP Web	Text Web
Login: 1st	85	93	95	91	90	94	94	93	97
Login: 3rd	94	98	96	99	100	96	99	99	100
Recall: 1st	–	–	–	19	23	29	32	54	43
Recall: 3rd	–	–	–	31	34	34	32	67	56

TABLE 4
Create, login, and recall times in seconds. Recall represents either at-home tasks or a second lab session. Missing values are identified as *na* and values that are not applicable with dashes.

	PCCP Lab	CCP Lab	PP Lab	PCCP 2wk All	PCCP 2wk S5	PP 2wk	Text 2wk	PCCP Web	Text Web
Create	26	26	42	91	67	25	26	68	11
Login	15	na	na	18	15	12	10	13	6
Recall	–	–	–	27	25	12	10	20	6
Login Click	8	8	8	11	8	6	–	10	–
Recall Click	–	–	–	24	17	6	–	15	–

years and the analysis evolved as we gained more experience. In this paper, results have been re-calculated using the same process, to allow for more accurate comparison. As such, the numbers may vary from earlier publications [1]–[5], [7].

5.1 Success rates

Success rates are reported on the first attempt and within three attempts. Success on the first attempt occurs when the password is entered correctly on the first try, with no mistakes or restarts. Success rates within three attempts indicate that fewer than three mistakes or restarts occurred. Mistakes occur when the participant presses the Login button but the password is incorrect. Restarts occur when the participant presses the Reset button midway through password entry and restarts password entry. Restarts are analogous to pressing delete while entering text passwords, except that PCCP's implicit feedback helps users detect and correct mistakes during entry.

Table 3 summarizes login and recall success rates, aggregated on a per user basis to ensure independence of the data. In all studies, success rates are highest for login. We conducted statistical analysis using Kruskal-Wallis tests to compare success rates for studies conducted with the same methodology; these tests are non-parametric tests similar to ANOVAs, but intended for use with skewed sample distributions.

We first compared success rates for the three lab studies (*PCCP Lab*, *CCP Lab*, *PP Lab*). Kruskal-Wallis tests compared success rates for login on the first and third attempts respectively across the three studies. No statistically significant differences were found in either comparison. This suggests no evidence that logging in with PCCP is any different than with PP or CCP.

Participants had the most difficulty recalling passwords after two weeks for all schemes. A closer look at the different conditions within the *PCCP 2wk* study is provided in Section 5.3. Here, only the S5 condition from the *PCCP 2wk* study is compared to the *PP 2wk* and *Text 2wk* studies since they have similar theoretical password spaces. Four comparisons were made: login first and third attempts, and recall first and third attempts. Kruskal-Wallis tests show no statistically significant differences in any of the comparisons. This result suggests no evidence that PCCP passwords are any harder to recall after two weeks than PP or text passwords at comparable levels of security.

No statistical differences were found between web studies (*PCCP Web* and *Text Web*) for login and recall success rates. This is especially noteworthy because inspection of the text passwords revealed that 71% of participants [3] re-used identical or similar passwords across accounts, whereas PCCP passwords were different by design. This suggests that PCCP passwords offer additional security since reuse across systems is not possible, yet this did not affect success rates.

5.2 Password entry times

Times are reported in seconds for successful password entry on the first attempt. For login and recall, we also report the "entry time": the actual time taken from the first click-point to the fifth click-point. The analogous measure was not recorded for text passwords.

Table 4 presents password entry times for each study. PCCP times are similar to other schemes in the initial lab studies. However, the general trend across the two-week recall (*PCCP 2wk's S5* condition) and web studies is that PCCP passwords take longer to enter than the other schemes when comparing schemes with similar password spaces (i.e., *PCCP 2wk S5* and *PCCP Web*). During password creation, this can partially be explained by participants who used the shuffle mechanism repeatedly. During recall, this may be because PCCP participants had to recall different passwords (since by design it is impossible

TABLE 5
Number of shuffles per image for password creation.

	PCCP Lab	PCCP 2wk All	PCCP 2wk S5	PCCP Web
Mean	3	7	3	10
Median	1	3	1	6

to reuse PCCP passwords), whereas over half of Text participants reused passwords or had closely related passwords, suggesting a reduced memory load.

5.3 Varying system parameters: PCCP 2wk study

We summarize the effects of modifying the number of click-points and the image size on user performance. Detailed results are available in an earlier paper [4].

Success rates: Success rates were very high for login; participants could successfully log in after a short time regardless of number of click-points or image size. Success rates after two weeks were much lower in all conditions, reflecting the artificial difficulty of the memory task — recalling 6 passwords created in a short time and not accessed for two weeks. The L7 condition had the lowest success rates, suggesting that passwords using large images and 7 click-points combined were most difficult.

Times: Mean times for each condition are generally elevated compared to times in the studies with smaller theoretical password spaces. No clear pattern emerges in the times taken to create passwords. A general increase in times can be seen in both the login and recall phases as more click-points or larger images are used. As should be expected, participants took much longer to re-enter their passwords after two weeks (recall), reflecting the difficulty of the task.

5.4 Shuffles

During password creation, PCCP users may press the shuffle button to randomly reposition the viewport. Fewer shuffles leads to more randomization of click-points across users. The shuffle button was used moderately. Table 5 shows the number of shuffles per image. For example, since *PCCP Lab* passwords involved 5 images, the mean number of shuffles per password would be $3 \times 5 = 15$. For the *PCCP 2wk* study, the mean and medians for all of this study's 6 conditions together (see the *All* column in Table 5) are higher than for S5 alone, indicating that for more difficult conditions, there was more shuffling.

The effect of shuffling on success rates are summarized in Table 6. Wilcoxon tests were used for statistical analysis; these are similar to independent sample t-tests, but make no assumptions about the sample distributions. The tests were conducted on login and recall success rates on the third attempt.

PCCP Lab study users who shuffled a lot had higher login success rates than those who shuffled little, and the result was statistically significant ($W = 91, p = 0.005$) (or $p = 0.015$ with Bonferroni correction). For the *PCCP 2wk* and *PCCP Web* studies, the same trend was apparent for login and recall, but the differences were not statistically significant.

Most participants used a common shuffling strategy throughout their session. They either consistently shuffled a lot at each trial or barely shuffled during the entire session. We interviewed participants to learn about their shuffling strategy. Those who barely shuffled selected their click-point by focusing on the section of the image displayed in the viewport, while those who shuffled a lot scanned the entire image, selected their click-point, and then proceeded to shuffle until the viewport reached that area. When questioned, participants who barely shuffled said they felt that the viewport made it easier to select a secure click-point. Those who shuffled a lot felt that the viewport hindered their ability to select the most obvious click-point on an image and that they had to shuffle repeatedly in order to reach this desired point.

5.5 Summary of Usability Results

We first summarize the studies with comparable theoretical password spaces (i.e., including *PCCP 2wk S5*). Overall, PCCP has similar success rates to the other authentication schemes evaluated (CCP, PassPoints, and text). PCCP password entry takes a similar time to the other schemes in the initial lab sessions, but the results indicate longer recall times for PCCP when recalling passwords beyond the initial session. Users who shuffled more had significantly higher success rates in the *PCCP Lab* study, but the difference in success rates between high and low shufflers was not statistically significant for the two-week or web studies. Furthermore, users reported favourable opinions of PCCP in post-task questionnaires [2].

Secondly, we compared conditions in the *PCCP 2wk* study. A general trend indicates that larger images or more click-points negatively impacts the password entry time. No clear pattern emerges between the 6 conditions for success rates, providing no evidence that either manipulation affects success rates in a consistent manner. However, the most difficult condition (L7) did have the lowest recall success rates.

6 Analysis of Password Distributions

6.1 Click-point clustering

To analyze the randomness and clustering of 2D spatial data across users, we turned to point pattern analysis [30] commonly used in biology and earth sciences. The analysis used *spatstat* [31], a spatial statistics package for the *R* programming language.

The J-statistic [32] from spatial analysis was used to measure clustering of click-points within datasets (the formation of hotspots). The J-statistic combines nearest-neighbour calculations and empty-space measures for a given radius r to measure the clustering

TABLE 6
Effect of shuffles on success rates (within 3 attempts). Success rates are percentages. "Users" represents the number of users who fell into each shuffling category. *n.s.* indicates that the statistical test was not significant. Values that are not applicable are identified with dashes.

	PCCP Lab		PCCP 2wk						PCCP Web		
	Users	Login	Users		Login		Recall		Users	Login	Recall
			All	S5	All	S5	All	S5			
Low (≤ 1 per image)	23	90	13	7	98	100	15	18	5	100	60
High (> 1 per image)	14	100	69	7	100	100	34	50	19	98	68
Wilcoxon Test	–	$W = 91, p = 0.005$	–		n.s.	n.s.	n.s.	n.s.	–	n.s.	n.s.

of points. A result of J closer to 0 indicates that all of the data points cluster at the exact same coordinates, $J = 1$ indicates that the dataset is randomly dispersed, and $J > 1$ shows that the points are increasingly regularly distributed. For passwords, results closer to $J(r) = 1$ are desirable since this would be least predictable by attackers. We examined clustering at $J(9)$ for the set of core images common across studies with at least 30 click-points per image for each study. A radius of 9 pixels approximates the 19×19 tolerance squares used by the system during password re-entry.

To compare sets of J-statistics to each other, we employed the following technique. Regarding the data as categorical, six categories stemming from the possible orderings are identified: (PCCP-CCP-PP), (PCCP-PP-CCP), (PP-CCP-PCCP), (PP-PCCP-CCP), (CCP-PP-PCCP), (CCP-PCCP-PP). Figure 4 shows the ordering for each of the 17 images. For example, the *bee* image falls in the PCCP-CCP-PP category because J(9) for PCCP exceeds J(9) for CCP, which exceeds J(9) for PassPoints. A Fisher's exact test between the observed results and the expected results (equal probability for each category) was applied to measure the significance of the association between the three categories. This test is similar to a chi-square test, but used when values in the associated contingency table are small.

Lab studies: We first compared the three lab studies [2]. Results show that *PCCP Lab* approaches complete spatial randomness for all 17 images (near $J = 1$) and is thus much more random than the *CCP Lab* and *PP Lab* datasets. Fisher's exact test shows that the difference is statistically significant ($p = 0.0005$).

All studies: For this paper, we also included data from the longer term studies. Figure 4 shows that the distribution of PCCP click-points is more random than PassPoints, but with differences smaller than in the lab studies. Fisher's Exact test shows that PCCP is more random than PassPoints and CCP ($p = 0.028$). A line graph was used for clarity, but these are discontinuous points.

Varying number of click-points: As detailed in an earlier paper [4], we examined the effects of the number of click-points on clustering on the *PCCP 2wk* data. Fisher's exact test shows no significant differences ($p = 0.358$), providing no evidence that increasing the number of click-points per password leads to more clustering across users.

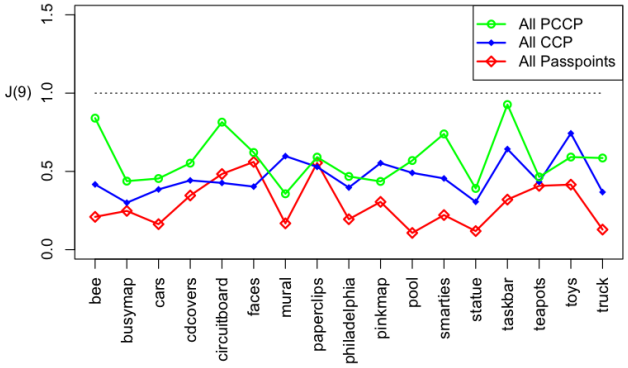


Fig. 4. $J(9)$ for the 17 core images, for all studies.

Varying image size: We also used the *PCCP 2wk* data to examine clustering due to image size [4]. Fisher's exact test shows a significant difference ($p = 0.002$), indicating significantly less clustering for larger images. This result suggests that PCCP's shuffle mechanism and viewport (if kept at the same pixel dimensions) are more effective in reducing clustering when used with larger images. We believe that this is due to the proportionally smaller area covered by the viewport in relation to the total size of the image making it less likely that known hotspots are available for selection.

6.2 Hotspot coverage

We summarize the hotspots per image using cumulative frequency distributions for the 17 core images. The distributions contain all user-chosen click-points for the given scheme for passwords that were, at minimum, successfully re-entered at least once during login. In other words, all click-points in the dataset are represented (including "hotspots" consisting of only one user-chosen click-point).

Figure 5 shows cumulative frequency distributions for each image. Grey lines represent the click-point distributions for the 17 images, for click-points collected across all studies for that particular scheme. One would expect half of the click-points to be contained in the most popular 50% of hotspots if click-points were completely randomly distributed. In the figures, this random distribution would appear as a straight diagonal line. In comparison, the PassPoints graph shows that in the worst case, half of all click-points are contained within the most popular 1.3%

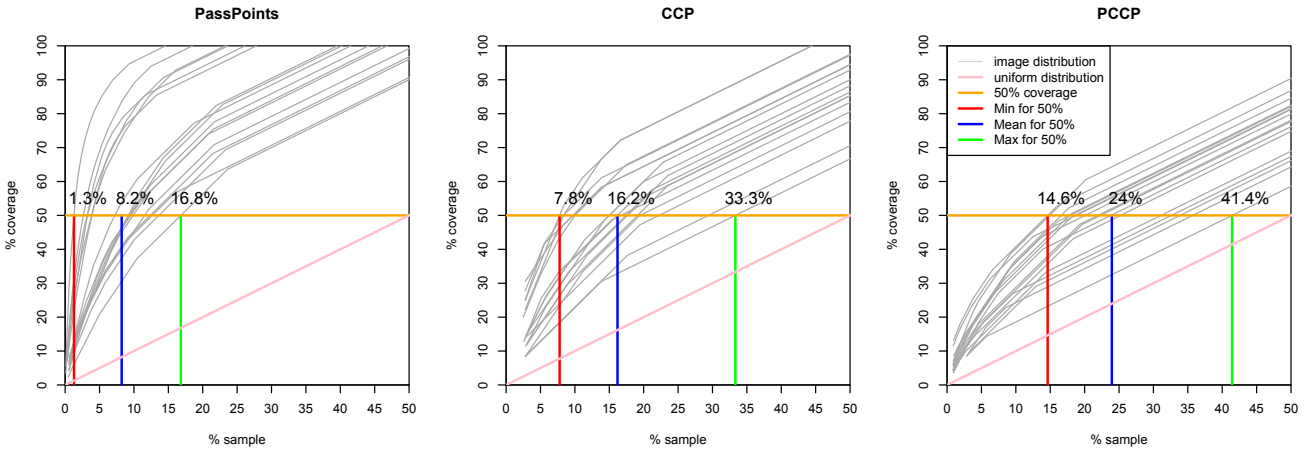


Fig. 5. Cumulative frequency distribution of hotspot coverage for PassPoints, CCP, and PCCP.

of hotspots within the distribution, while in the best case, half are contained within the most popular 17.3%. For PCCP, half of click-points fall within the within the top 14.6% hotspots on the worst case image. On the best image, half are contained within the top 41.4% for PCCP, approaching the ideal of 50%.

To test for significance in the differences between PP, CCP and PCCP, we looked at the dictionary data for the 17 images individually. Kruskal-Wallis 3-way tests show strong significant differences between the distributions ($p < 0.00001$) for each image. We further compared only CCP and PCCP, to look at the effect of the viewport and shuffling mechanism specifically. Kruskal-Wallis 2-way tests show strong significance for each image. This indicates that PCCP click-points have a flatter distribution and thus an attack dictionary based on hotspots should be less effective for PCCP than for the other schemes (see also Section 7.1).

This analysis focused on individual click-points, not entire passwords. However with the recommended implementation, attackers get no partial feedback on correctness partway through an offline guess, precluding divide-and-conquer (piecewise) attacks on PCCP.

6.3 Spatial Patterns

We looked at several password characteristics to find whether known patterns exist that could help attackers fine-tune an attack strategy. These patterns involve the spatial position of click-points relative to each other and do not consider the background image. In earlier work [5], we performed this analysis on a subset of the current data, focusing primarily on data from lab studies. We now perform similar analysis on all 5-click-point password data on 451×331 pixel images collected to date for each scheme. Details are included in a technical report [33], but the analysis reveals similar results to the original paper [5].

The click-point distributions of PCCP along the x- and y-axes fell within the range for random distributions with 95% probability, while those of PassPoints showed a clear progression from top-left to bottom-right based on the ordinal position of the click-points within the password. We believe that the difference is users' selection strategy is based on whether the click-points are selected on one image, as in PassPoints, or distributed across several images. With one image, as in PassPoints, users tend to start at one corner of the image and progress across the image with each subsequent click-point. However, with CCP and PCCP, users see a new image for each click-point and tend to select each click-point independently, with no regard to its ordinal position within the password.

Click-points within PassPoints were much closer together (i.e., shorter segments between successive click-points), while CCP's segments were the longest and within range of the random distributions. PCCP's segments were slightly shorter than CCP's. Given that no other spatial patterns are apparent for PCCP, we suspect that these shorter segments are an artifact of the viewport positioning algorithm, which slightly favoured more central areas of the image. For further discussion of viewport positioning, see Section 8.3.

With respect to angles and slopes formed between adjacent line segments within passwords, analysis shows that PCCP passwords have large angles and favour no particular direction. In contrast, PassPoints passwords often form straight horizontal or vertical lines. Similarly, the frequency distributions for the overall shapes formed by following the path from the first to last click-point for PCCP are within the range of the random datasets. PassPoints passwords were much more likely to form identifiable shapes.

6.4 Colour Patterns within PCCP Passwords

We also considered strategies of choosing click-points based on the content of the image. Specifically, we examined 859 PCCP passwords for colour consistency.

We examined the 11×11 pixel centre of the tolerance square for each click-point. We then calculated the mean of the perceptual distance between

the colour surrounding each click point, using the CIE76 definition of ΔE^*_{ab} ranging from 0 to 100, with a value of 2.3 regarded as a "just noticeable difference". The distribution of these mean colour differences ranged normally from 8.08 to 60.21 with a mean of 29, but even the minimum of 8.08 included easily distinguishable colours. This suggests that it is very unlikely that users chose passwords consisting of very similar colours. We next isolated the hues of click points within a password and calculated their differences, but found little evidence of overall consistencies within passwords. Visual inspection of the passwords revealed no other evident relationships.

6.5 Summary of Password Distributions

Analysis of click-point clustering showed that PCCP had the least clustering of click-points across different users. Similarly, hotspot analysis showed that PCCP had the flattest click-point distribution and was least likely to contain hotspots when compared to CCP and PassPoints. In tests of numerous spatial relationships and patterns, we found no significant differences between PCCP and what is expected to occur by chance. And finally, colour analysis showed that users did not choose click-points within passwords based on colour.

7 Security

We next discuss PCCP's resistance to standard security threats: guessing attacks and capture attacks.

7.1 Guessing Attacks

The most basic guessing attack against PCCP is a brute-force attack, with expected success after exploring half of the password space (i.e., with a theoretical password space of 2^{43}, success after 2^{42} guesses). However, skewed password distributions could allow attackers to improve on this attack model. Section 6 examined the password distributions based on several characteristics. We now consider how these could be leveraged in guessing attacks.

Pattern-based attack: One of the proposed attacks [21] on PassPoints is an automated pattern-based dictionary attack that prioritizes passwords consisting of click-points ordered in a consistent horizontal and vertical direction (including straight lines in any direction, arcs, and step patterns), but ignores any image-specific features such as hotspots. The attack guesses approximately half of passwords collected in a field study on the *Cars* and *Pool* images (two of the 17 core images) with a dictionary containing 2^{35} entries, relative to a theoretical space of 2^{43}.

Given that PCCP passwords are essentially indistinguishable from random for click-point distributions along the x- and y-axes, angles, slopes, and shapes (see technical report [33]), such pattern-based attacks would be ineffective against PCCP passwords.

Hotspot attack with all server-side information: PassPoints passwords from a small number of users can be used [34] to determine likely hotspots on an image, which can then be used to form an attack dictionary. Up to 36% of passwords on the *Pool* image were correctly guessed with a dictionary of 2^{31} entries.

The attacker's task is more difficult for PCCP because not only is the popularity of hotspots reduced, but the sequence of images must be determined and each relevant image collected, making a customized attack per user. An online attack could be thwarted by limiting the number of incorrect guesses per account.

To explore an offline version of this attack, assume in the worst case that attackers gain access to all server-side information: the username, user-specific seed, image identifiers, images (see Section 8.2), hashed user password and corresponding grid identifiers (see Section 8.1). The attacker determines the first image I_1 from the available information. Hotspot analysis identifies the center of the largest hotspot on I_1. The next image I_2 is predicted based on I_1's hotspot and the user-specific seed which determines the image mapping. In this way, a password guess contains the largest hotspot on each predicted image. The same process could be used to determine passwords using 5-subsets of popular hotspots. The resulting dictionary would grow combinatorially based on the number of hotspots followed at each stage.

Because each user password in PCCP involves different images, it is difficult to collect enough statistical information in an experimental setting for meaningful hotspot analysis. Our best analysis in this direction involved using data on the 17 core images. For each of the 95 user passwords involving solely these images, used as target passwords to find, we built a list of the 10 largest hotspots for each of the 17 images, using all *PCCP Lab* and *PCCP 2wk - S5* data. These hotspot lists were combined to form a guessing dictionary containing 2^{37} entries for the 17 images. None of the 95 passwords appeared in the dictionary, indicating that no password in our collected data consisted entirely of top-10 hotspots. We expect that this attack would be similarly unfruitful for other images of similar complexity. We also note that this attack is infeasible unless an attacker has previous knowledge of which images belong to a user's password.

We next consider a second hotspot attack strategy under the same assumption of all server-side information being known, and in this case consider the level of effort required for a 3% chance of guessing a target password. With the basic configuration of 19×19 pixel tolerance squares, and 451×331 pixel images, there are approximately 400 tolerance squares per image. If no hotspots exist and there are no patterns (i.e., if random and independent click-points are chosen), each tolerance square has an equal $1/400$ chance of being part of the user's password. However, from Figure 5 we know that for the PassPoints datasets

explored, on average the largest 8.2% of hotspots cover 50% of user-chosen click-points. This means that for approximately a 3% ($(50/100)^5$) chance of guessing a password, a dictionary constructed of all ordered sequences of 5 click-points, each click-point being among the corresponding set of these hotspots from the appropriate (assumed known) image, would contain 2^{26} entries. In comparison, PCCP requires the top 24% of hotspots to achieve the same coverage, giving a dictionary of 2^{33} entries for a 3% chance of guessing a password comprised solely of hotspots.

Hotspot attack with only hashed password: Suppose attackers gain access only to the hashed passwords, for example, if the passwords and other information are stored in separate databases. Offline dictionary attacks become even less tractable. The best attack would seem to involve building a guessing dictionary whose entries are constructed from the largest hotspots on random combinations of images.

7.2 Capture Attacks

Password capture attacks occur when attackers directly obtain passwords (or parts thereof) by intercepting user-entered data, or by tricking users into revealing their passwords. For systems like PCCP, CCP, and PassPoints (and many other knowledge-based authentication schemes), capturing one login instance allows fraudulent access by a simple replay attack. We summarize the main issues below; detailed discussion is available elsewhere [12].

Shoulder-surfing:

All three cued-recall schemes discussed (PCCP, CCP, PassPoints) are susceptible to shoulder-surfing although no published empirical study to-date has examined the extent of the threat. Observing the approximate location of click-points may reduce the number of guesses necessary to determine the user's password. User interface manipulations, such as reducing the size of the mouse cursor or dimming the image may offer some protection, but have not been tested. A considerably more complicated alternative is to make user input invisible to cameras, for example by using eye-tracking as an input mechanism [35].

Malware: Malware is a major concern for text and graphical passwords, since keylogger, mouse-logger, and screen scraper malware could send captured data remotely or otherwise make it available to an attacker.

Social Engineering: For social engineering attacks against cued-recall graphical passwords, a frame of reference must be established between parties to convey the password in sufficient detail. One preliminary study [36] suggests that password sharing through verbal description may be possible for PassPoints. For PCCP, more effort may be required to describe each image and the exact location of each click-point. Graphical passwords may also potentially be shared by taking photos, capturing screen shots, or drawing, albeit requiring more effort than for text passwords.

PCCP and CCP have a security advantage over PassPoints: an attacker launching a phishing attack would need to retrieve many images from the server instead of only one. With a man-in-the-middle (MITM) attack, only one image per click-point would need to be retrieved, since the correct image would be identified by the legitimate website when the user's click-point is entered. However, attackers who collect the images beforehand would need to gather all of them in order to display the correct next image when the user enters a click-point (see Section 8.2 for discussion of the image selection algorithm). Attackers who make assumptions about likely hotspots and only collect the corresponding images risk missing images if the user clicks elsewhere. Although social engineering remains a threat with PCCP, attacks require significantly more effort and have a lower probability of success than for text passwords or PassPoints.

In light of these potential guessing and capture attacks, PCCP is best deployed in systems where offline attacks are not possible, and where any attack must involve an online system that can limit the number of guesses per account per time period; this limit should include password restarts. Even with account-locking after t failed login attempts, defences must throttle such online guessing attacks sufficiently to guard against system-wide attacks across W accounts since an attacker gets $t * W$ guesses per time window [37]. All client-server communication should be made securely (e.g., through SSL) to maintain the secrecy of user click-points and images.

7.3 Summary of Security Analysis

Given that hotspots and click-point clustering are significantly less prominent for PCCP than for CCP and PassPoints, guessing attacks based on these characteristics are less likely to succeed. Taking into account PCCPs sequence of images rather than a single image offers further reduction in the efficiency of guessing attacks. For capturing attacks, PCCP is susceptible to shoulder-surfing and malware capturing user input during password entry. However, we expect social engineering and phishing to be more difficult than for other cued-recall graphical password schemes due to PCCPs multiple images.

8 Relevant Implementation Issues

The following discusses two prototype implementations of PCCP and highlights issues relevant for a best-practice implementation. The first prototype, intended for experiments only, included design decisions which facilitated data gathering but would not be advisable in actual deployment. The lab and two week recall studies (Sections 4.1 and 4.2) used a standalone J# application custom-designed to guide participants through the experimental process. This provided a controlled environment to gather initial

data about the usability and security of the schemes. Image selection was done in such a way that all users saw a particular core set of images and all password information (e.g., click-point coordinates and images) was stored in the clear, allowing evaluation of characteristics like the effect of password choice.

The second prototype moved towards an ecologically valid system taking into account implementation details necessary for a real web-based authentication system. The *PCCP Web* study (Section 4.3) was conducted with a web-based authentication framework (*MVP* [28]) especially designed to be deployed and accessed by users in their regular environments. The system is intended to allow authentication to become a secondary task, by supporting primary tasks on real websites that require users to log in as part of the process. The *PCCP Web* study used modified versions of Wordpress blogs and phpBB forums. The modifications were made to locally-installed packages, altering the authentication process. A button was included rather than a textbox for password entry; pressing the button opened the authentication window and loaded the PCCP authentication module, which takes the userid from the website, collects the user's PCCP password, and returns an encoded password string (see Section 8.1). The original websites remained responsible for authentication, using the encoded string as they would use an entered text password.

The following sections describe several practical design and implementation choices made in building the second prototype, and the reasoning behind them.

8.1 Discretization

Discretization of click-points allows for approximately correct click-points to be accepted by the system without storing exact click-point coordinates in the clear. Our second prototype implemented *Centered Discretization* [29], wherein an invisible discretization grid is overlaid onto the image, dividing the image into square tolerance areas, to determine whether a login click-point falls within the same tolerance area as the initial click-point. For each click-point, the grid's position is set during password creation by placing it such that there is a uniform tolerance area centered around the original click-point, by calculating the appropriate (x, y) grid offset (Gx, Gy) (in pixels) from a (0,0) origin at the top-left corner of the image. On subsequent user login, the system uses the originally recorded offsets to position the grid and determine the acceptability of the each login click-point.

For each password P_W, the system hashes the username W, as a unique salt intended to force user-specific attack dictionaries, and the following details for each click-point ($i = 1 \ldots 5$): its grid offset (Gx_i, Gy_i), a tolerance area identifier Tx_i, Ty_i (indicating the exact square containing the click-point), and its image identifier I_i. The system also stores the following additional information A_W in the clear: Gx, Gy for each click-point and a random seed S_W used to determine the pool of images for a given user (see Section 8.2). These components are described as:

$$C_i = (I_i, Tx_i, Ty_i, Gx_i, Gy_i)$$
$$P_W = h([C_1 \ldots C_i], W)$$
$$A_W = ([Gx_1, Gy_1 \ldots Gx_i, Gy_i], S_W)$$

The discretization grids and offsets are transparent and unknown to users. An attacker who gained access to this information would not know the user's password, but might try to use it to guess higher probability click-points, e.g., by overlaying corresponding grids onto images looking for popular target points centered within grid squares. Whether this provides any attack advantage over trying to exploit hotspots without grid information remains an open question.

8.2 Deterministic Image Sequencing

Each image is displayed using a deterministic function $I_{i+1} = f(S_W, C_i)$, based on the user-specific random seed S_W and the previous user-entered click-point C_i; $I_1 = f(S_W, 0)$. S_W is set during password creation and used to randomly select images from the system-wide pool of images, numbered from 0 to N. It is stored in the clear as part of A_W, described above. During login, the sequence of images is re-generated using f. This approach allows a different sequence of images per each user while still guaranteeing a consistent mapping of click-points to images for each user. If a password is changed, a new S_W is generated.

Using this implementation, there is a possibility that images are reused for a given user. For example, a user clicking on an incorrect location during login might, by chance, see an image belonging somewhere else within their password. While this poses a potential usability concern, the likelihood of this happening is correspondingly low with enough images. There is no evidence this occurred in any of our studies.

The image selection algorithm could be modified to disallow all image reuse for a given user, albeit possibly providing enough verifiable information to determine the entire password to an attacker who learns only the last image: if each possible traversal of images is unique, knowing the last image means that with effort, an attacker could find the unique password that ends with that particular image.

For usability, the minimum total number of images should be the number of tolerance squares in one grid (i.e., 432 in the basic PCCP configuration). This avoids the situation where multiple locations lead to the same next image, breaking the *implicit feedback* property of PCCP and likely confusing users. All images could be reused at each stage in the password and for every user. This strategy has the highest probability of collision where a user clicks on an incorrect click-point and unfortunately sees an image belonging elsewhere in their password. This probability can be

reduced or nearly eliminated if the overlap of images is reduced between password stages, increasing the number of images in a user's set. The trade-off is between usability problems of potential collisions during incorrect logins and reducing the ease of password reconstruction should an attacker learn some of the images in a user's password. A related question to explore is the possibility of collisions across systems if different deployments use the same image sets.

An alternative to increasing the number of images is to use larger images but crop them differently for each user. Hotspot analysis would be more difficult for attackers because the coordinates of hotspots could not be directly applied across accounts. If furthermore, each user receives a different pool of images (perhaps as an overlapping subset of the overall set of images in the system, as determined by S_W and f), an attacker would need to collect this data on a per-user basis when launching an attack.

8.3 Viewport Details

The viewport visible during password creation must be large enough to allow some degree of user choice, but small enough to have its intended effect of distributing click-points across the image. Physiologically, the human eye can observe only a small part of an image at a time. Selecting a click-point requires high acuity vision using the fovea, the area of the retina with a high density of photoreceptor cells [38]. The size of the fovea limits foveal vision to an angle of approximately 1° within the direct line to the target of interest. At a normal viewing distance for a computer screen, say $60cm$, this results in sharp vision over an area of approximately $4cm^2$. We chose the size of the viewport to fall within this area of sharp vision. For the lab studies, where we had control over the size of the screen and the screen resolution, we chose a viewport of 75×75 pixels. However, for the web-based system we used a slightly larger 100×100 pixel viewport since participants may be using a wide variety of system configurations. While the web-based prototype was designed primarily for standard size screens, it could be modified to accommodate smart phones or smaller screens. The system could determine the type of device (e.g., through browser settings data) and alter the size of the viewport dynamically.

The viewport positioning algorithm randomly placed the viewport on the image, ensuring that the entire viewport was always visible and that users had the entire viewport area from which to select a click-point. This design decision had the effect of de-emphasizing the edges of the image, slightly favouring the central area. A potential improvement would be to allow the viewport to *wrap* around the edges of the image, resulting in situations were the viewport is split on opposite edges of the image.

8.4 Variable number of click-points

A possible strategy for increasing security is to enforce a minimum number of click-points, but allow users to choose the length of their password, similar to minimum text password lengths. The system would continue to show next images with each click, and users would determine at which point to stop clicking and press the login button. Although most users would likely choose the minimum number of click-points, those concerned with security and confident about memorability could select a longer password.

9 Concluding Remarks

A common security goal in password-based authentication systems is to maximize the effective password space. This impacts usability when user choice is involved. We have shown that it is possible to allow user choice while still increasing the effective password space. Furthermore, tools such as PCCP's viewport (used during password creation) cannot be exploited during an attack. Users could be further deterred (at some cost in usability) from selecting obvious click-points by limiting the number of shuffles allowed during password creation or by progressively slowing system response in repositioning the viewport with every shuffle past a certain threshold. The approaches discussed in this paper present a middle-ground between insecure but memorable user-chosen passwords and secure system-generated random passwords that are difficult to remember.

Providing instructions on creating secure passwords, using password managers, or providing tools such as strength-meters for passwords have had only limited success [39]. The problem with such tools is that they require additional effort on the part of users creating passwords and often provide little useful feedback to guide users' actions. In PCCP, creating a less guessable password (by selecting a click-point within the first few system-suggested viewport positions) is the easiest course of action. Users still make a choice but are constrained in their selection.

Another often cited goal of usable security is helping users form accurate mental models of security. Through our questionnaires and conversations with participants in authentication usability studies, it is apparent that in general, users have little understanding of what makes a good password and how to best protect themselves online. Furthermore, even those who are more knowledgeable usually admit to behaving insecurely (such as re-using passwords or providing personal information online even when unsure about the security of a website) because it is more convenient and because they do not fully understand the possible consequences of their actions.

Guiding users in making more secure choices, such as using the viewport during password creation, can help foster more accurate mental models of security

rather than vague instructions such as "pick a password that is hard for others to guess". This persuasive strategy has also been used with some success to increase the randomness of text passwords [40].

Better user interface design can influence users to select stronger passwords. A key feature in PCCP is that creating a harder to guess password is the *path-of-least-resistance*, likely making it more effective than schemes where secure behaviour adds an extra burden on users. The approach has proven effective at reducing the formation of hotspots and patterns, thus increasing the effective password space.

ACKNOWLEDGMENT

We thank Chris Deschamps for his help in implementing the framework used in the web-based studies. The fifth author is Canada Research Chair in Authentication and Software Security, and acknowledges NSERC for funding the chair and a Discovery Grant. Funding from NSERC ISSNet and the fourth author's NSERC Discovery Grant is also acknowledged.

REFERENCES

[1] S. Chiasson, R. Biddle, and P. van Oorschot, "A second look at the usability of click-based graphical passwords," in *ACM Symposium on Usable Privacy and Security (SOUPS)*, July 2007.

[2] S. Chiasson, A. Forget, R. Biddle, and P. van Oorschot, "Influencing users towards better passwords: Persuasive Cued Click-Points," in *Human Computer Interaction (HCI), The British Computer Society*, September 2008.

[3] S. Chiasson, A. Forget, E. Stobert, P. van Oorschot, and R. Biddle, "Multiple password interference in text and click-based graphical passwords." in *ACM Computer and Communications Security (CCS)*, November 2009.

[4] E. Stobert, A. Forget, S. Chiasson, P. van Oorschot, and R. Biddle, "Exploring usability effects of increasing security in click-based graphical passwords," in *Annual Computer Security Applications Conference (ACSAC)*, 2010.

[5] S. Chiasson, A. Forget, R. Biddle, and P. C. van Oorschot, "User interface design affects security: Patterns in click-based graphical passwords," *International Journal of Information Security, Springer*, vol. 8, no. 6, pp. 387–398, 2009.

[6] J. Yan, A. Blackwell, R. Anderson, and A. Grant, "The memorability and security of passwords," in *Security and Usability: Designing Secure Systems That People Can Use*, L. Cranor and S. Garfinkel, Eds. O'Reilly Media, 2005, ch. 7, pp. 129–142.

[7] S. Chiasson, P. van Oorschot, and R. Biddle, "Graphical password authentication using Cued Click Points," in *European Symposium On Research In Computer Security (ESORICS), LNCS 4734*, September 2007, pp. 359–374.

[8] L. Jones, A. Anton, and J. Earp, "Towards understanding user perceptions of authentication technologies," in *ACM Workshop on Privacy in Electronic Society*, 2007.

[9] L. O'Gorman, "Comparing passwords, tokens, and biometrics for user authentication," *Proceedings of the IEEE*, vol. 91, no. 12, December 2003.

[10] A. Jain, A. Ross, and S. Pankanti, "Biometrics: a tool for information security," *Transactions on Information Forensics and Security (TIFS)*, vol. 1, no. 2, pp. 125–143, 2006.

[11] D. Nelson, V. Reed, and J. Walling, "Pictorial Superiority Effect," *Journal of Experimental Psychology: Human Learning and Memory*, vol. 2, no. 5, pp. 523–528, 1976.

[12] R. Biddle, S. Chiasson, and P. van Oorschot, "Graphical passwords: Learning from the first twelve years," *ACM Computing Surveys (to appear)*, vol. 44, no. 4, 2012.

[13] A. De Angeli, L. Coventry, G. Johnson, and K. Renaud, "Is a picture really worth a thousand words? Exploring the feasibility of graphical authentication systems," *International Journal of Human-Computer Studies*, vol. 63, no. 1-2, pp. 128–152, 2005.

[14] E. Tulving and Z. Pearlstone, "Availability versus accessibility of information in memory for words," *Journal of Verbal Learning and Verbal Behavior*, vol. 5, pp. 381–391, 1966.

[15] S. Wiedenbeck, J. Waters, J. Birget, A. Brodskiy, and N. Memon, "PassPoints: Design and longitudinal evaluation of a graphical password system," *International Journal of Human-Computer Studies*, vol. 63, no. 1-2, pp. 102–127, 2005.

[16] ——, "Authentication using graphical passwords: Effects of tolerance and image choice," in *1st Symposium on Usable Privacy and Security (SOUPS)*, July 2005.

[17] K. Golofit, "Click passwords under investigation," in *12th European Symposium On Research In Computer Security (ESORICS), LNCS 4734*, September 2007.

[18] A. Dirik, N. Menon, and J. Birget, "Modeling user choice in the Passpoints graphical password scheme," in *3rd ACM Symposium on Usable Privacy and Security (SOUPS)*, July 2007.

[19] J. Thorpe and P. C. van Oorschot, "Human-seeded attacks and exploiting hot-spots in graphical passwords," in *16th USENIX Security Symposium*, August 2007.

[20] A. Salehi-Abari, J. Thorpe, and P. van Oorschot, "On purely automated attacks and click-based graphical passwords," in *Annual Computer Security Applications Conf. (ACSAC)*, 2008.

[21] P. C. van Oorschot, A. Salehi-Abari, and J. Thorpe, "Purely automated attacks on PassPoints-Style graphical passwords," *IEEE Trans. Info. Forensics and Security*, vol. 5, no. 3, pp. 393–405, 2010.

[22] B. Fogg, *Persuasive Technologies: Using Computers to Change What We Think and Do*. Morgan Kaufmann Publishers, San Francisco, CA, 2003.

[23] J. Wolf, "Visual Attention," in *Seeing*, K. De Valois, Ed. Academic Press, 2000, pp. 335–386.

[24] D. Davis, F. Monrose, and M. Reiter, "On user choice in graphical password schemes," in *13th USENIX Security Symposium*, 2004.

[25] PD Photo, "PD Photo website," http://pdphoto.org, accessed February 2007.

[26] D. Florencio and C. Herley, "Where do security policies come from?" in *Symposium on Usable Privacy and Security*, 2010.

[27] M. Weir, S. Aggarwal, M. Collins, and H. Stern, "Testing metrics for password creation policies by attacking large sets of revealed passwords," in *Computer and Communications Security (CCS)*, 2010.

[28] S. Chiasson, C. Deschamps, M. Hlywa, G. Chan, E. Stobert, and R. Biddle, "MVP: A web-based framework for user studies in authentication (poster)," in *Symposium on Usable Privacy and Security (SOUPS)*, 2010.

[29] S. Chiasson, J. Srinivasan, R. Biddle, and P. C. van Oorschot, "Centered discretization with application to graphical passwords," in *USENIX Workshop on Usability, Psychology, and Security (UPSEC), San Franscisco, USA*, April 2008.

[30] P. Diggle, *Statistical Analysis of Spatial Point Patterns*. Academic Press: New York, NY, 1983.

[31] A. Baddeley and R. Turner, "Spatstat: An R package for analyzing spatial point patterns," *Journal of Statistical Software*, vol. 12, no. 6, pp. 1–42, 2005.

[32] M. van Lieshout and A. Baddeley, "A nonparametric measure of spatial interaction in point patterns," *Statistica Neerlandica*, vol. 50, no. 3, pp. 344–361, 1996.

[33] S. Chiasson, E. Stobert, A. Forget, R. Biddle, and P. van Oorschot, "Persuasive cued click-points: Design, implementation, and evaluation of a knowledge-based authentication mechanism," School of Computer Science, Carleton University, Tech. Rep. TR-11-03, February 2011.

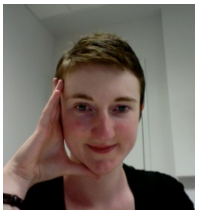
[34] P. C. van Oorschot and J. Thorpe, "Exploiting predictability in click-based graphical passwords," *Journal of Computer Security*, vol. 19, no. 4, pp. 669–702, 2011.

[35] A. Forget, S. Chiasson, and R. Biddle, "Shoulder-surfing resistance with eye-gaze entry in click-based graphical passwords." in *ACM SIGCHI Conference on Human Factors in Computing Systems: Note (CHI)*, 2010.

[36] P. Dunphy, J. Nicholson, and P. Olivier, "Securing Passfaces for description," in *4th ACM Symposium on Usable Privacy and Security (SOUPS)*, July 2008.

[37] B. Pinkas and T. Sander, "Securing passwords against dictionary attacks," in *9th ACM Conference on Computer and Communications Security (CCS)*, November 2002.
[38] A. Duchowski, *Eye Tracking Methodology: Theory and Practice*, 2nd ed. Springer, 2007.
[39] D. Florencio and C. Herley, "A large-scale study of WWW password habits," in *16th ACM International World Wide Web Conference (WWW)*, May 2007.
[40] A. Forget, S. Chiasson, P. van Oorschot, and R. Biddle, "Improving text passwords through persuasion," in *4th Symposium on Usable Privacy and Security (SOUPS)*, July 2008.

Sonia Chiasson is an Assistant Professor in the School of Computer Science at Carleton University in Ottawa, Canada. Her main research interests are in usable security: the intersection between human-computer interaction (HCI) and computer security. Current projects are on user authentication, usable security for mobile devices, and computer games for teaching about computer security.



Elizabeth Stobert is a PhD student in Computer Science at Carleton University. She has an MA in Psychology (2011) as well as a BA (2009) and B.Math (2008) from Carleton University. Her research interests are in the areas of HCI, security, and cognition.

Alain Forget is currently a Ph.D. Candidate of Computer Science. His thesis research is focusing on various aspects of usable authentication, including users' mental models of passwords, using Persuasive Technology to improve users' mental models of authentication and computer security, and exploring various solutions to the challenges users have with contemporary text passwords.

Robert Biddle is a Professor in the School of Computer Science and Institute of Cognitive Science at Carleton University in Ottawa, Canada. His research is in Human-Computer Interaction and Software Design. His current research projects are on usable security, especially authentication and security decision-making, and on large-scale multi-touch devices, especially environments for collaborative design and visualization.

Paul C. van Oorschot is a Professor of Computer Science at Carleton University in Ottawa, where he is Canada Research Chair in Authentication and Computer Security. He was Program Chair of USENIX Security 2008, Program co-Chair of NDSS 2001 and 2002, and co-author of the Handbook of Applied Cryptography (1996). He is on the editorial board of IEEE TIFS and IEEE TDSC. His current research interests include authentication and identity management, security and usability, software security, and computer security.