A Password Manager that Doesn't Remember Passwords

Elizabeth Stobert
School of Computer Science
Carleton University
Ottawa, Canada
elizabeth.stobert@carleton.ca

Robert Biddle
School of Computer Science
Carleton University
Ottawa, Canada
robert.biddle@carleton.ca

ABSTRACT

The problems with passwords are well-known: secure passwords are difficult to remember, users have too many passwords, and users have difficulty matching their passwords to accounts. Password managers and cued graphical passwords are two password solutions that address the issues of memorability and keeping track of of passwords. We have developed *Versipass*, a password manager that incorporates key elements of password managers and cued graphical passwords to avoid existing problems of password memorability and associating passwords with accounts. Instead of remembering passwords, Versipass remembers image cues for graphical passwords. These cues help users to better remember their passwords and to more easily link passwords with accounts. Versipass also facilitates safe password reuse by allowing users to use the same image cue for multiple accounts.

Categories and Subject Descriptors

K.6.5 [**Computing Milieux: Security and Protection**]: Authentication.

Keywords

Authentication, password managers, graphical passwords

1. INTRODUCTION

The problems with passwords are well-known: secure passwords are difficult to remember, users have too many passwords, and users have difficulty matching their passwords to accounts. These problems lead users to insecure coping mechanisms such as picking passwords that are memorable but easy for attackers to guess, reusing passwords across multiple accounts, and writing passwords down.

Password managers and cued graphical passwords are two password solutions that address the issues of memorability and keeping track of of passwords. We have developed *Versipass*, a password manager that incorporates key elements of password managers and cued graphical passwords to avoid existing problems of password memorability and associating passwords with accounts. Instead of remembering passwords, Versipass remembers image cues for graphical passwords. These cues help users to better remember their passwords and to more easily link passwords with accounts. Versipass also facilitates safe password reuse by allowing users to use the same image cue for multiple accounts. When a user attaches the same password cue to multiple accounts, Versipass hashes the password differently for each account, leading to safer reuse.

Alone, neither cued graphical passwords nor password managers completely solves users' problems with passwords. However, these ideas together provide support for the problems that users commonly have with passwords. They provide better memorability, while better organizing users' passwords and the associations between passwords and accounts. They also support users' existing habits (such as password reuse), while improving the security of these habits.

2. PASSWORD COPING TECHNIQUES

Cued graphical passwords and password managers are two existing technologies that can address the problems user users cope with the difficulties of passwords.

Graphical passwords are passwords that use images for users to log in. They leverage the *picture superiority effect* [19], a psychological result that finds that humans are better at remembering images than textual information. Graphical passwords leverage this result for more memorable passwords. Research on graphical password schemes has been ongoing for approximately 15 years, but graphical passwords have yet to see wide deployment. However, there are indications that they might become more mainstream. The Android mobile operating system includes a popular pattern unlock option for unlocking the phone, and more recently, Windows 8 includes a picture password option to log into operating system user accounts. Research on graphical passwords has shown that graphical passwords are more memorable than text passwords, even when password space is equivalent [25].

User-chosen graphical passwords have been shown to be susceptible to *hotspots*, areas of the image that are significantly more likely to be chosen by users in their passwords, which make graphical passwords vulnerable to dictionary

NSPW'14, September 15–18, 2014, Victoria, BC, Canada.

ACM 978-1-4503-3062-6/14/09 ...$15.00.
http://dx.doi.org/10.1145/2683467.2683471.

attacks [29]. Randomly assigned graphical passwords avoid this problem by removing the user's choice, and this is the approach we advocate.

Our research on graphical passwords has shown that assigned graphical passwords are indeed more memorable than assigned text passwords [25], and that the image cues associated with some graphical passwords can help users distinguish multiple passwords from each other [5]. In particular, we developed Image PassTiles, a cued-recall graphical password system where the user is shown an image overlain by a grid. The users' password consists of a set of grid squares, and to log in, the user must click on the correct squares. In a study of memorability, we found that random assigned PassTiles passwords were more memorable than other assigned passwords. These passwords were also less susceptible to multiple password interference, and were quick to enter. This remained true, even as the security of the assigned passwords was increased. However, though PassTiles passwords are more memorable than comparable text passwords, users are still unable to remember as many of them as they have accounts.

Password managers are essentially the best existing technology to help the end-user handle knowledge based authentication. Password managers are programs that save or generate and enter passwords for the user, and have an advantage over technologies like single sign-on (SSO) because the user may choose to use them on any account (instead of waiting for the account administrators to implement SSO). Password managers address the issue of quantity because they are able to remember any number of passwords. Although there is variation in how specific managers work, there are two main models of password manager functionality [7]. Wallet-based managers store the user's passwords in a "password wallet" file that is protected by a master password. Hashing password managers compute a cryptographic hash of a master password at every login [22].

Password managers are not without problems. Both hashing and wallet password managers have a single point of failure in the master password. If this password is discovered, an attacker can gain access to all of a users' accounts. Some password managers are also incautious about how the passwords are stored: some of the browser-based password managers use a wallet model, but store users' passwords in the clear [14]. Another issue for password managers is roaming: password managers are typically difficult to transport to other computers and browsers, and can complicate life for users who use a variety of computers.

Dedicated password managers are not widely used [13], and users instead resort to less secure techniques to manage their passwords. Although standard password advice is not to reuse passwords across multiple accounts, the majority of users do [10]. Reusing passwords helps users to manage having passwords for large numbers of accounts. Additionally, with fewer passwords, it is easier for users to keep track of which password is associated with which account. While reusing passwords creates real security risks, these risks are poorly aligned with the incentives to create, remember, and keep track of unique passwords.

Another technique that users often develop to remember their passwords is to write them down. Recording passwords has obvious security risks, but many security experts actually advise writing passwords down if they can be kept in a physically secure location [3, 24]. The argument is that writing a password down allows the user to remember more elaborate and more secure passwords, and if the password is kept at home, having a written copy of it will likely never risk exposing it to anonymous online attackers. However, many users do not securely store their recorded passwords, and instead choose to keep them in online locations that are more easily accessible but more vulnerable to attack.

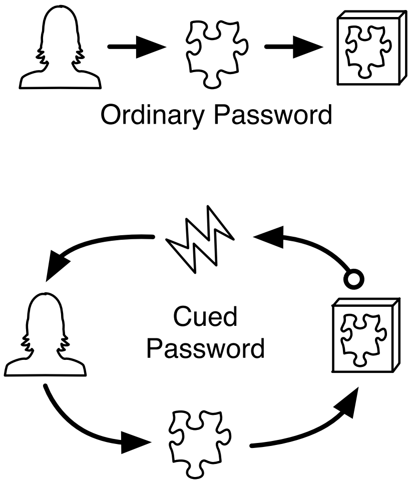


Figure 1: *The cue model.* The account provides a password cue to the user, who uses it to associate and remember their password.

3. A NEW MODEL FOR PASSWORDS

Versipass incorporates two key ideas to address the issues of cueing passwords and password reuse. Although these ideas are used elsewhere, Versipass is the first system to combine them, and we believe that the combination of these ideas gives more benefit to the user than either idea alone.

3.1 The Cue Model

Versipass is more rightly a password *cue* manager, since it does not actually store any passwords. Instead, it stores password cues that allow users to generate passwords and safely pass them to websites.

In the *cue model* (Figure 1), the password system sends a cue to the user, who uses it to recall their password. This model is used in cued graphical passwords and in other cued passwords, as well as by Versipass. More generally, the cue model describes all cued passwords. The cue may be visual, or audio, but its purpose is to cue memory and provide context for the user. Another example of cued passwords are challenge questions, often used for password resets (challenge questions are discussed in Section 7.3).

3.2 The Category Model

Figure 2 shows how category passwords are used to protect multiple accounts in Versipass. A single password is assigned to a category of accounts (we imagine that a user might categorize accounts with a similar theme, or that are used at a similar time). Although not portrayed in the figure, the password that is reused on multiple websites is salted and hashed differently for each of the websites where it is used.

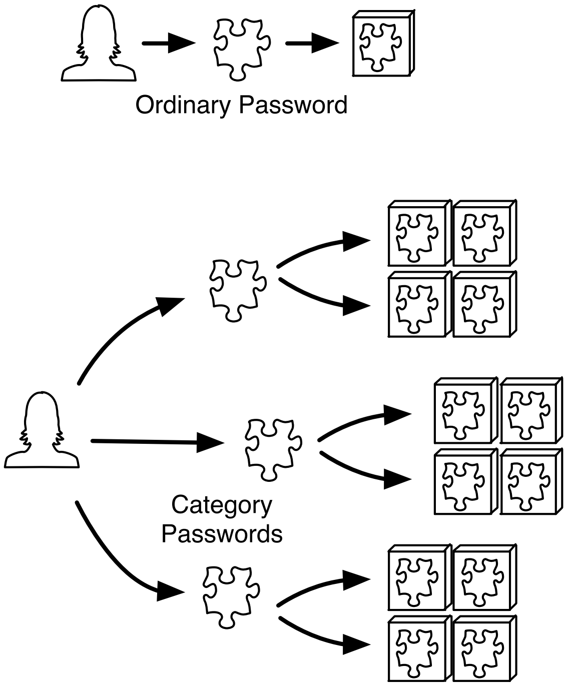


Figure 2: *The category model.* A user has a password for each category of accounts.

However, this step is invisible to users, much the way salting and hashing within websites is currently invisible to users.

Salted and hashed category passwords allow users to more safely reuse passwords across accounts. Password reuse is a key coping strategy for users, and one they are unlikely to abandon. If users are able to remember a small number of strong (randomly assigned) graphical passwords, then category passwords can allow them to apply those passwords to all of their accounts.

Through the use of the cue and category models, Versipass supports users' existing habits, while simultaneously protecting users against the risks of those habits. Since Versipass hashes the input responses differently for each website, it provides protection for users who reuse passwords. This allows passwords to be reused more safely while not increasing the burden on the user. Versipass also provides explicit cues for users, which help users to distinguish passwords and accounts. There is evidence that users already look for cues in the webpage environs, but Versipass provides a strong cue that is present at password creation to help users remember their passwords, and associate them with the correct accounts. Cues also minimize the possibility that users will systematically guess all of their passwords on every website, a coping strategy that can expose all of users' passwords to an attacker in a single session.

The category model builds on the work of existing hashing password managers. PwdHash is a password manager developed by Ross et al. [22], that hashes a user's master password together with the website domain name and an optional salt to create unique passwords for every website. However, PwdHash uses the same password for all accounts, and does not include the notion of grouped accounts under a single password category. Versipass introduces the notion of cues to support random passwords, and allows users to protect accounts in different categories, avoiding a single point of failure.

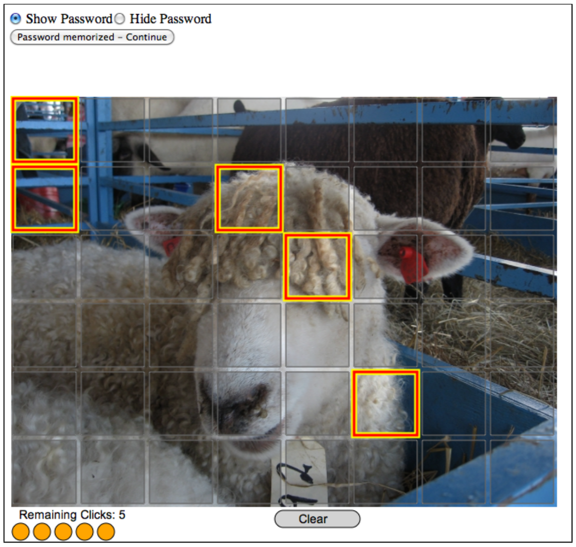

● Show Password ○ Hide Password
Password memorized – Continue
Remaining Clicks: 5
Clear

Figure 3: The password creation interface for Image PassTiles. The tiles highlighted in orange are the user's assigned password tiles.

Another advantage of Versipass is that it allows graphical passwords to be used on any website without requiring any server-side changes. Password schemes are implemented through Versipass, and require no changes to existing website infrastructure. This means that users can choose the type of password with which they are most comfortable.

4. VERSIPASS PROTOTYPE

In this section, we present the current Versipass prototype. This prototype makes use of the Image PassTiles graphical password scheme, and we present the details of that scheme, as well as walkthroughs of the Versipass prototype and a discussion of the issues facing the implementation of Versipass.

4.1 Image PassTiles

Image PassTiles is a variant of PassTiles, a graphical password scheme developed by us for use in research [25]. It is a locimetric scheme that leverages cued recall, and our studies have shown that it has good memorability, usability, and security, as well as some logistical advantages.

Image PassTiles presents the user with an image overlain by a grid (Figure 3), and the user's password consists of a set of grid squares, known as tiles (hence, Pass*Tiles*). To log in, the user must simply click on each of their password tiles, in any order. At its standard configuration, the PassTiles grid has 48 tiles (6×8), and passwords consist of 5 tiles, but the grid size and password length can both be adjusted to provide any level of security.

Image PassTiles passwords are randomly assigned: they are generated by the system, and the user is given no choice about which tiles are included in their password. Our studies have shown that these randomly generated passwords are more memorable than equivalently secure random text passwords [25]. The role of the background image is to provide a memory cue for the user to help them both remember their password tiles and distinguish different passwords from each other. At password creation, grid tiles are outlined, so that the user can remember which tiles form their password by identifying the parts of the image that lie behind the relevant tiles. Because the passwords are randomly generated, they are not related to the image cues, and no information can be inferred from the images by themselves. The background images can be any images, and could even be the user's personal images.

Image PassTiles passwords are encoded as text strings by labelling each password tile with a string, and then concatenating the string of clicked tiles. Since password tiles may be clicked by the user in any order, the text string is then given a standard ordering for comparison. The encoded text strings can be treated as any other text passwords, and can be salted and hashed for secure storage.

4.2 The Password Manager

The current prototype of Versipass is built to integrate with MVP [4], a framework we constructed for earlier research to allow us to conduct ecologically valid tests of password systems. The Versipass prototype draws its implementation of Image PassTiles from MVP, and saves user account details in the MVP database. The prototype is implemented as a web application that the user must sign into to set up their passwords and accounts. To use Versipass when logging in to websites, we have provided a *bookmarklet* for the user to save. The bookmarklet is a short piece of JavaScript code that relays the userID and URL of the site back to MVP to look up the saved information for that account. Figure 4 shows a sequence diagram of a login using Versipass.

We initially chose the bookmarklet approach for its flexibility, considering that users would be able to save the bookmarklet on any computer, even where they did not have access to system settings. However, we now think that a better approach would be a browser extension. Although the extension would have to be installed by the user, it would avoid issues with copying and pasting the password from one browser tab (or window) to another. The current bookmarklet approach has had to work around these problems. We plan to implement future prototypes of Versipass as browser extensions, rather than the current approach.

Figure 5 shows screenshots of the step-by-step process to add a new website account and password to Versipass, and Figure 6 shows the step-by-step process that a user would take to log into a website using Versipass. In both walkthroughs, the assumption is that the user is logged into the password manager. The user does not need to have the Versipass webpage open, but they do need to log in to Versipass at the beginning of every session (the login information is saved in a session cookie).

Ross et al. [22] identify a number of challenges affecting the implementation of a browser extension for their password manager, PwdHash. These issues include JavaScript attacks on the cleartext password, salting, encoding the password to fit website password policies, compliance with browser auto-complete functionality, helping the user with password resets, roaming across different computers, and dictionary attacks. We address some of these challenges here, and address issues relating to Versipass' security in Section 5.

Versipass lets users select part of their own salts. In our prototype, the user provides a string that is used together with the URL of the website as the salt for that account. We took this approach because situations arise where users need to change their passwords, and a simple way to do this is to change the salt. Simply using the URL would mean that there is no way to change the salt for an account. We have considered creating a mechanism to randomly generate salts, but this could be dangerous if a user changed the salt without understanding the effect on the generated password and effectively locked themselves out of their accounts (even if they knew the password input). One way to mitigate this problem might be to keep a list of previously used salts, allowing the user to revert to earlier salts.

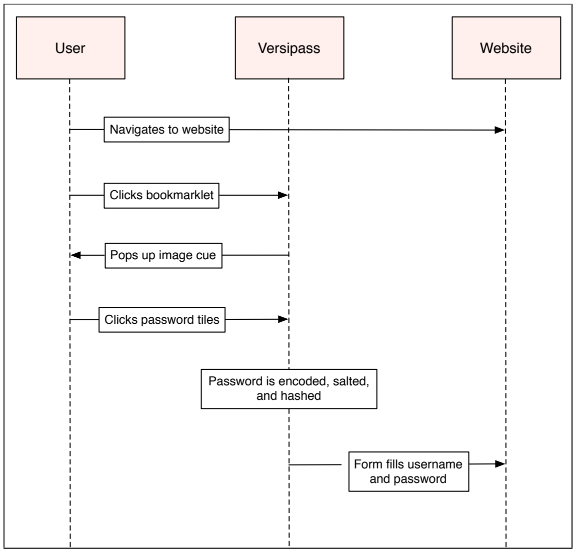
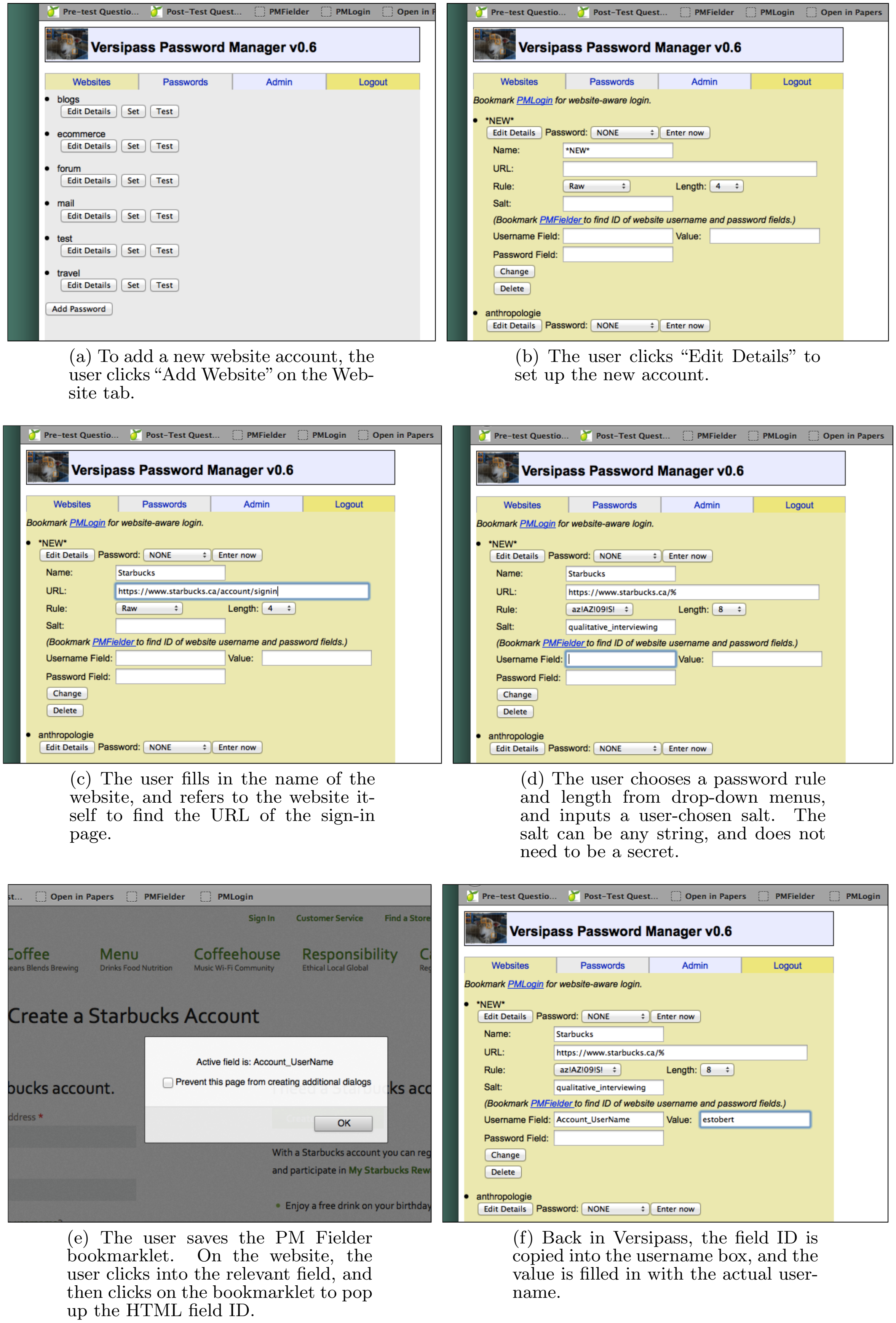


Figure 4: A sequence diagram of the login process using Versipass.

Versipass addresses the issue of encoding passwords to comply with password policies by having the user select the length and policy for the generated password. However, password policies are often not explicitly displayed, or only explained when they are not followed, so this might pose a problem for users. A possible improvement to Versipass would be the automation of this step – either through crowdsourcing (password policies for a website could be saved for all users of Versipass), or through some means where Versipass could automatically learn the website password policy.

In its current implementation, Versipass can be used on any computer without any installation. This has a definite advantage to a user who uses multiple computers and does not have installation privileges on those computers, but isn't such an advantage for most users and creates a clunky interface. Implementing Versipass as a browser extension will limit where Versipass can be used, but improve its usability when it is installed.



(a) To add a new website account, the user clicks "Add Website" on the Website tab.

Bookmark PMLogin for website-aware login.

- *NEW*
  Password: NONE
  Name: *NEW*
  URL:
  Rule: Raw Length: 4
  Salt:
  (Bookmark PMFielder to find ID of website username and password fields.)
  Username Field: Value:
  Password Field:
- anthropologie
  Password: NONE

(b) The user clicks "Edit Details" to set up the new account.

- *NEW*
  Password: NONE
  Name: Starbucks
  URL: https://www.starbucks.ca/account/signin
  Rule: Raw Length: 4
  Salt:
  Username Field: Value:
  Password Field:
- anthropologie
  Password: NONE

(c) The user fills in the name of the website, and refers to the website itself to find the URL of the sign-in page.

- *NEW*
  Password: NONE
  Name: Starbucks
  URL: https://www.starbucks.ca/%
  Rule: az!AZ09!S! Length: 8
  Salt: qualitative_interviewing
  Username Field: Value:
  Password Field:
- anthropologie
  Password: NONE

(d) The user chooses a password rule and length from drop-down menus, and inputs a user-chosen salt. The salt can be any string, and does not need to be a secret.

Active field is: Account_UserName

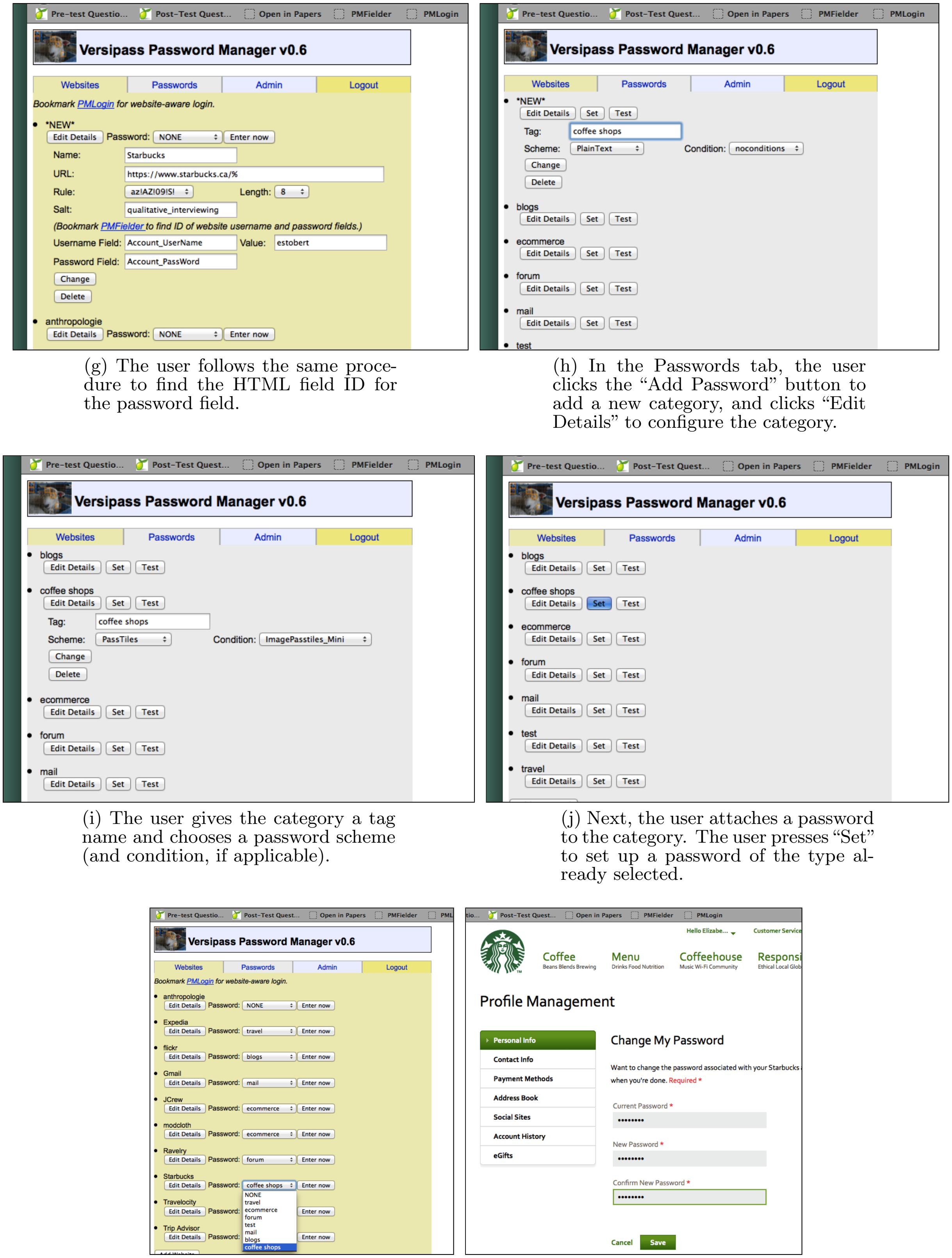
□ Prevent this page from creating additional dialogs

(e) The user saves the PM Fielder bookmarklet. On the website, the user clicks into the relevant field, and then clicks on the bookmarklet to pop up the HTML field ID.

- *NEW*
  Password: NONE
  Name: Starbucks
  URL: https://www.starbucks.ca/%
  Rule: az!AZ09!S! Length: 8
  Salt: qualitative_interviewing
  Username Field: Account_UserName Value: estobert
  Password Field:
- anthropologie
  Password: NONE

(f) Back in Versipass, the field ID is copied into the username box, and the value is filled in with the actual username.

Figure 5: Adding a new account and password category to Versipass (continued on next page).



Bookmark PMLogin for website-aware login.

- *NEW*
  Password: NONE
  Name: Starbucks
  URL: https://www.starbucks.ca/%
  Rule: az!AZ!09!S! Length: 8
  Salt: qualitative_interviewing
  (Bookmark PMFielder to find ID of website username and password fields.)
  Username Field: Account_UserName Value: estobert
  Password Field: Account_PassWord
- anthropologie
  Password: NONE

(g) The user follows the same procedure to find the HTML field ID for the password field.

- *NEW*
  Tag: coffee shops
  Scheme: PlainText Condition: noconditions
- blogs
- ecommerce
- forum
- mail
- test

(h) In the Passwords tab, the user clicks the "Add Password" button to add a new category, and clicks "Edit Details" to configure the category.

- blogs
- coffee shops
  Tag: coffee shops
  Scheme: PassTiles Condition: ImagePasstiles_Mini
- ecommerce
- forum
- mail

(i) The user gives the category a tag name and chooses a password scheme (and condition, if applicable).

- blogs
- coffee shops
- ecommerce
- forum
- mail
- test
- travel

(j) Next, the user attaches a password to the category. The user presses "Set" to set up a password of the type already selected.

Bookmark PMLogin for website-aware login.

- anthropologie Password: NONE
- Expedia Password: travel
- flickr Password: blogs
- Gmail Password: mail
- JCrew Password: ecommerce
- modcloth Password: ecommerce
- Ravelry Password: forum
- Starbucks Password: coffee shops
- Travelocity Password:
- Trip Advisor Password:

(k) In the Websites tab, the drop-down menu is used to link the "coffee shops" password to the Starbucks website account.

Profile Management

Change My Password

Want to change the password associated with your Starbucks when you're done. Required *

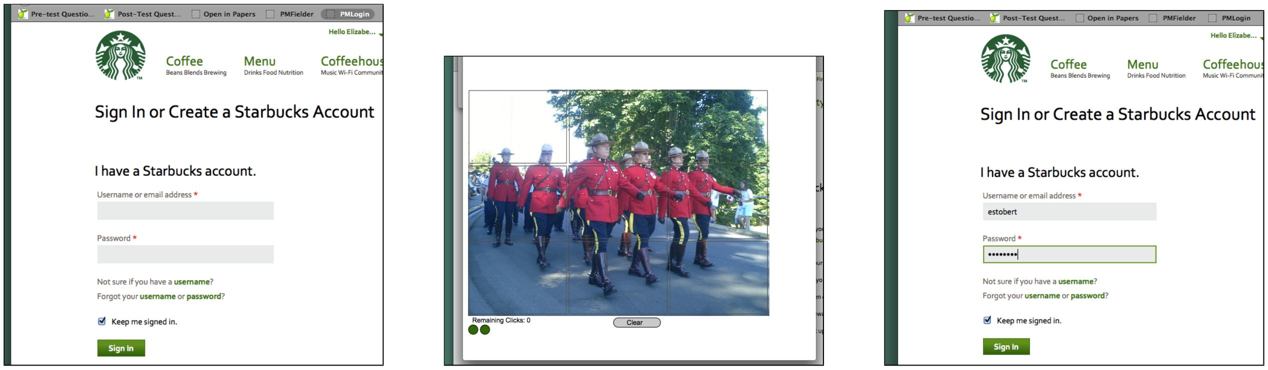
Current Password * ••••••••

New Password * ••••••••

Confirm New Password * ••••••••

(l) The user must change the password on the Starbucks website to match the password that is generated by Versipass.

Figure 5: (cont.) Adding a new account and password category to Versipass.



Username or email address * estobert

Password * ••••••••

☑ Keep me signed in.

(a) The user clicks the PMLogin bookmarklet from the website login page.

(b) The bookmarklet pops up the image cue, and the user enters their response.

(c) Versipass generates the password string, and pastes the username and password into the appropriate fields.

Figure 6: Logging in using Versipass.

5. VERSIPASS SECURITY

In our security analysis, we address two main types of attacks on passwords: guessing attacks and capture attacks [1].

5.1 Image PassTiles

5.1.1 Guessing Attacks

Image PassTiles passwords consist of 5 tiles on a grid of 48 tiles (6 × 8 tiles). These tiles are randomly chosen by the system, and the user has no choice about which password tiles they will be assigned. To log in, the user must click on the correct 5 tiles in any order. The theoretical entropy of PassTiles passwords is calculated as $log_2\binom{48}{5} = 21$ bits. Since these passwords are randomly assigned, their theoretical security is equal to their effective security against guessing attacks.

Although the theoretical security of Image PassTiles is 21 bits in the configuration described here, various elements of these passwords are configurable. The number of tiles may be increased by changing the grid dimensions, or passwords may be made to consist of more than 5 tiles. Of course, as the security is increased, passwords also become correspondingly more difficult for users to remember. In configuring Image PassTiles, we have followed Florencio, Herley, and van Oorschot's suggestion [11] that 20 bits of entropy should be sufficient to protect against online attack.

The strength of Image PassTiles passwords against offline attack is dependent on how the passwords are stored on the websites. If properly salted and hashed, Image PassTiles passwords should not be any more vulnerable to offline attack than 21-bit text passwords.

5.1.2 Capture Attacks

While strong enough against guessing attacks, Image PassTiles passwords are not as well-defended against shoulder-surfing attacks, where the attacker records the input and later enters it from the recording. PassTiles passwords are defended against casual shoulder-surfing by not displaying which tiles have been clicked on the screen at login, but a more sophisticated attack with a camera could record these clicks. Because of this, we suggest that Image PassTiles is probably best suited to physically secure entry locations, rather than publicly visible places.

Image PassTiles passwords are the same at every login, and as such, are vulnerable to replay attacks. However, like other graphical password systems, Image PassTiles is moderately resistant to these accounts because the attacker must devise a methodology for recording the position of the image onscreen as well as recording the location of the clicks. This presents more difficulties than simply installing a keylogger to record text password entries.

Phishing attacks on Image PassTiles are complicated slightly because the attacker needs to present the user with the correct image cue in order to get their password. However, if the phisher is able to learn the username, they may be able to fetch the user's cue in real time. Once the attacker has the cue, they can present it to the user, and record the user's clicks to reveal their password.

Like text passwords, Image PassTiles passwords are vulnerable to man-in-the-middle attacks unless they are securely transmitted using SSL. SSL encrypts the password during transmission, preventing an attacker from intercepting the transmission, reading the password (or modifying it), and passing it along to the intended end point. Of course, SSL cannot protect against compromised end points, such as in a successful phishing attack.

5.2 Password Manager

Since Versipass does not store passwords, there is no risk that an attacker who gains access to Versipass might be able to access the user's accounts. However, such an attacker would be able to access personal information (the list of sites where the user has accounts, and the user's usernames), and cause significant disruption for the user (by changing their category passwords or by deleting information). Users should still be able to access their accounts via emailed password resets or challenge questions, but this kind of attack could be very frustrating for users. One possibility is that Versipass could provide a mechanism for users to create offline backups that could be later restored in the case of an attack.

Versipass protects against phishing attacks in the same way that other password managers do. If the URL of a login page is unknown to Versipass, the bookmarklet will not work, and Versipass will not pop up the image cue, or generate (and form-fill) a password for the site. A useful addition might be a warning message that alerts the user that they might be trying to log into a fraudulent website.

In the current form, Versipass stores all of the user's information on a server. This means that the user can access Versipass from any computer, and have access to their accounts no matter where they are. Even if the user doesn't have the ability to save bookmarks, they can still generate their passwords in the Versipass web page and copy them over to the websites manually. However, this also means that if the server is attacked (for example, in a denial-of-service (DoS) attack), the user might not be able to log into any of their accounts. If the Versipass database was stored locally (for example, with a browser extension), the user would be less vulnerable to DOS attacks, but would have difficulties when using multiple computers.

In a way, Versipass provides a second layer of authentication in addition to what is provided on the website. Because the user user must be signed in to Versipass to access their password cues and generation, they will have provided two layers of authentication any time they use Versipass to log into a website. It is also possible that the Versipass authentication could incorporate an additional authentication factor. For example, authentication to Versipass might happen via a one-time code that is provided over SMS.

Since Versipass only operates on the password entry portion of website logins, it should not interfere with existing two-factor authentication schemes. As long as the user still has access to the second factor, Versipass should not complicate the two-factor authentication.

6. EVALUATION

Versipass is a new paradigm for the end user, and as such, it is important that the password manager interface should be usable and comprehensible for users. Specific usability testing is needed to evaluate the new concepts within Versipass (password categories, no stored passwords, etc.), as well as the PassTiles passwords. Because Versipass's intended purpose is to help the end user behave more securely, we felt it was important to conduct early evaluations of Versipass' usability. These evaluations were intended to help us understand the users' impressions of the system and to inform future iterations of the design.

We conducted two evaluations of Versipass: a preliminary user study, and a cognitive walkthrough. Our user study was conducted with five participants from our lab (while familiar with our work, these participants were not security researchers or experts in security), and was intended to give an initial impression of the usability of the system. No personal data was collected, and the participants were fully informed about the purpose and intent of the study. We subsequently conducted a cognitive walkthrough to give more detail about the problems found in the user study, and to allow us to better understand the problems

Since the two evaluations yielded similar results, we present the combined results of the two evaluations in Section 6.2. We first present the separate methodologies, and then the combined results.

6.1 Method

6.1.1 User Study

In our user study, participants were given a scenario and a few tasks and asked to think aloud as they completed the tasks. We gave participants a brief explanation about the purpose of the study, and told participants that they should be honest about their impressions of the system since we were evaluating the system, not their performance. Participants began by completing a short pre-test questionnaire that collected demographic information as well as information about the participant's security habits and knowledge. They were given a written scenario describing a persona, Jo Kelly, who had begun a new job as a travel blogger and wanted to manage their multiple accounts using the password manager. They were also given index cards with Jo's account information. At the end of the scenario description, participants were asked to complete four tasks which included logging in to Versipass, adding a new account to Versipass, logging in to that account using Versipass, and configuring Versipass to use the same password on multiple websites.

Participants were asked to 'think aloud', describing their actions and thought processes as they worked through the task list. In the case that the participant became confused by a task unrelated to the password manager, the experimenter provided help in navigation. For instance, participants had to interact minimally with a few external websites, and one participant had difficulty finding a login link on a website.

Four participants reported that they use some kind of password manager in real life, and three of these reported using the password manager built into the Firefox web browser. The remaining participant reported using LastPass.

6.1.2 Cognitive Walkthrough

In our second evaluation, we conducted a cognitive walkthrough to follow up on the results of the user study, and to help us better understand where the problems that arose in the user study were rooted. A cognitive walkthrough is an inspection method evaluation that focuses on learnability for new users [30]. A group of evaluators steps through several tasks, and evaluates a persona's problem solving process at each step. At every step, three questions are asked: *Will the user know what to do?*, *Will the user see how to do it?*, and after they have completed the action, *Will the user know that they did the right thing?*

We conducted a pluralistic walkthrough, in which several evaluators came together to assess the usability of the prototype. The participants in our cognitive walkthrough were ourselves, who designed and constructed the prototype under evaluation, and the final participant was another graduate student in our department, who was familiar with usability and security, but who was not specifically knowledgeable about password managers or familiar with our prototype or design. We conducted the evaluation using the prototype described above in Section 4. We projected it onto a large screen in our lab to be equally visible to all participants, and each participant took on a role in the walkthrough. The roles were operator/navigator, notetaker, and persona.

For our cognitive walkthrough we used the same persona and tasks as in the user study. We assumed that Jo had already done some initial setup of the password manager.

She had an existing Versipass account with credentials, and had already added a few websites and passwords.

6.2 Results

In general, participants in the user study had difficulty with the password manager tasks, and we identified several ways in which our interface might be difficult to learn. Participants had differing levels of success, but none of the participants was able to successfully complete all of the tasks. The results of our evaluations identified three types of problems: mental model problems, security issues, and incorrect interface elements.

6.2.1 Mental Model Problems

Many of the errors that users made while completing the tasks seemed to be caused by a lack of understanding of how Versipass worked. In the cognitive walkthrough, although we had deep knowledge of the system, we still found ourselves misled based on a lack of awareness of where we were in the sequence of events.

In the user study, participants were not given any explanation of how Versipass worked, and although most participants eventually figured out how the manager worked, most of them missed the nuances of the functions, and misunderstood the advantages given by Versipass. Participants were initially confused by the separation of website accounts from password accounts. Most participants figured out this functionality, but they did not understand that using the same password cue on different website entries could mean sending different passwords to different accounts. The same result was echoed in the cognitive walkthrough, where we observed that it might be possible for a user to make their way through the set up process without really understanding the given advantages, or how Versipass differs from other password managers.

Since users did not necessarily understand that Versipass is designed to encode input passwords and hash them before passing them to the websites, several participants in our study were confused about why they would choose to use Versipass. In the words of one participant "if I enter the same password in the [password manager] [as I would enter on the website], why bother?". One of the major impediments to usability that we observed in the cognitive walkthrough was that the passwords on actual websites need to be changed to match the passwords generated by the password manager. This process is both tedious and confusing, since the user must login to the website with their existing credentials, use the password manager to generate a new password, change the website password to the generated password, and then configure the password manager for automated login.

A key problem that surfaced in the cognitive walkthrough was error recovery. We managed to make a serious error when we pasted the incorrect password into a password field on an actual website. Once this password was saved, we had no means of accessing the account except via the emailed reset. This incident emphasized the need for an error recovery strategy. Our mistake was caused by a few factors: a misunderstanding of which information was meant to be copied into which field, and the starred-out characters in the password field on the website prevented us from noticing the error until it was too late. Error recovery is a recurring issue for security products: it can be difficult to help users recover from an error without revealing key information, and it can also be difficult to know how much information was breached while the error was being made. The lack of mental model only worsens the problem, since the user may not even understand when they make errors.

6.2.2 Security Issues

The second task (adding an email account to Versipass) in the user study was designed to highlight how participants would protect an account with high security value. This goal was not explicitly stated, and in general, we did not emphasize the realism of the study, but it was interesting that only one participant commented at all on the security of Versipass. This person stated that they would have liked to protect their email account, but did not understand how to go about doing so.

Other participants made no mention of security at all, which was surprising in a study that primed users to pay particular attention to a tool for managing computer security. Participants clearly did not consider how Versipass might help them to increase the security of their accounts, meaning that the manager failed in one of its primary purposes.

6.2.3 Incorrect Interface Elements

Several aspects of Versipass's user interface were found to be misleading to users in the study. While these kinds of errors can be relatively easily addressed, they were a major impediment to the usability of the system, especially on the scale that they occurred. In the cognitive walkthrough, we flagged similar issues, and discussed specific fixes for many problems.

There were several places where the names of buttons and text entry fields confused participants, causing them to incorrectly enter information, enter incorrect information, or lose information. In the accounts tab, the button that participants expected to be called "Save" was actually labelled "Change". Multiple participants mentioned looking for a save button, and more than once, participants neglected to click the "Change" button, leading to information loss. Also in the account management tab, text fields were provided where the user was meant to input the HTML labels of the username and password on the actual website login page. Participants consistently misunderstood what information was supposed to be put in these boxes, and several participants put their own usernames and passwords in these boxes. This is a dangerous error, since they were typing actual passwords in cleartext. Another example of poor labelling was a set of drop-down menus that changed the parameters of the hashed password. Participants did not understand the meaning of the labels "Rule", "Length", or "Salt", and were unable to divine their meaning from the information in the menus. As such, they were not able to use these features of the password manager.

As participants explored Versipass, there were a few key elements that participants consistently missed or ignored. The major example of this was the link to the PM Fielder bookmark. The link to the bookmarklet was contained in an explanatory sentence above the relevant fields. Strangely, although it was close to fields where participants were having trouble, participants consistently failed to read this sentence. One participant commented that they had disregarded the sentence because it began with the word "bookmark", and they associated that instruction with non-critical information. Of participants who eventually noticed the instruction, one participant was able to successfully follow the directions after clicking the link, but another participant appeared bewildered by the short instructions and made no attempt to follow them or even read them closely.

Setting up an account with Versipass necessarily involves a certain amount of interaction with a website because the website password must be changed to be the same as the password generated by Versipass. During our cognitive walkthrough, we realized that the interaction with the website needs to be carefully guided, and that the order in which actions should be completed also needs to be carefully indicated. For instance, the new account tab in Versipass should tell the user where the next step should happen, and what they should be trying to accomplish on the website.

Another place in which Versipass could better guide the user is the order in which the user should fill fields in the password manager. Although creating a website account in Versipass requires the same information as it would without Versipass (for instance, choosing the character sets used in the password and the length of the password), these steps are made more explicit in Versipass, and appear at a point in the process where the user may not have yet considered them. In our walkthrough, we realized that the fields in the new account tab had to be filled in out of order, causing confusion.

Many of the usability problems in the prototype are related to the nature of an early prototype. This prototype was constructed as a proof of concept and as an early exploration into the feasibility of the password manager idea, and we did not pay particular attention to style or user interaction. The usability issues with our prototype are a sobering reminder that these kinds of problems cannot be glossed over. We need to devote time and effort to producing an interface that is consistent, attractive, and user-friendly.

7. RELATED WORK

In this section, we outline related work in several areas related to the Versipass project.

7.1 Password Use

A number of studies have investigated the number of passwords and accounts that users possess. A 2006 study by Gaw and Felten [13] found that undergraduates at their university had an average of about 12 accounts, but that users had fewer unique passwords and frequently reused passwords. Additionally, they found that users with more accounts reused passwords more frequently. They also asked participants about their justifications for password reuse, and most cited easier memorability. Participants also mentioned classifying their accounts by the desired level of privacy and security.

Florencio and Herley [10] conducted a large scale study of password use through the six-month deployment of a Microsoft toolbar, collecting usage data from more than $250,000$ users. During the six month study, the average user accessed a total 25 accounts, and logged into eight accounts per day. The average user had 6.5 passwords, each of which was shared across 3.9 websites. The study also examined the type of passwords that users entered, and the study confirmed anecdotal evidence that most users select weak passwords and mostly choose passwords consisting of lower case letters.

A two-week diary study of password use by Hayashi and Hong [15] collected detailed records of the number of password entries. In their study, users accessed a mean of 8.6 accounts, and they estimated that most participants had about 11 accounts in total. Although the study did not directly investigate password reuse, all participants reported reusing passwords.

Reusing passwords across multiple accounts is a common coping strategy for having multiple passwords. Although this strategy is widely used [13, 10, 15], it carries non-trivial security risks. If a reused password is discovered (e.g., through a leaked password set), an attacker may be able to gain access to several accounts. In an empirical study of multiple leaked datasets, Das et al. [8] found that 43% of passwords in their total data set were reused across multiple accounts. They also demonstrated this information on reused passwords can be leveraged for more efficient password attacks.

While "strict" security advice tells users not to reuse passwords, anecdotal evidence suggests that password reuse is the technique used by many security experts to manage passwords. Norman describes his experience asking security professionals about their personal password strategies:

> "What do security professionals do? I asked attendees at the security conferences. Many of the security experts said they do 'what everybody does: have two passwords.' " (p. 63, [17])

However, little work has investigated how to make password reuse safer.

7.2 Graphical Passwords

Graphical passwords are image-based passwords. A variety of graphical password schemes have been proposed, many of these with good usability and security properties. A survey of graphical passwords is available from Biddle, Chiasson, and van Oorschot [1]. We mention notable examples of graphical password systems here, but more detail may be found in the survey.

Graphical passwords leverage the *picture superiority effect* [19], a psychological result that says that humans have better memory for images than words. The picture superiority effect is reasoned to be due to *dual coding.* Dual coding theory [18] speculates that the human brain encodes visual information in two forms: once in a visual form corresponding to the image, and once in a verbal descriptive form. This dual encoding makes the information more available upon retrieval, and thus leads to better remembering.

De Angeli et al. [9] propose a classification system for graphical passwords based on the type of input requested by the password system. *Cognometric* password schemes ask the user to correctly recognize images among a larger array to log in, *locimetric* schemes ask the user to identify target points on a single image, and *drawmetric* schemes ask the user to draw a particular image to log in. Although these classifications are based on input, they also loosely correspond to the different types of memory retrieval that may be leveraged by graphical password schemes. Cognometric systems leverage recognition memory, locimetric systems leverage cued recall memory, and drawmetric systems leverage pure recall memory. We conducted a series of studies comparing the types of memory retrieval in graphical passwords [25], and found that although recognition-based passwords were most memorable, they were prohibitively slow to enter. Cued-recall passwords were comparably memorable, but were significantly faster for users to log in.

An important feature of some graphical password schemes is that they support cueing. The images in graphical passwords help users to better remember their passwords, and to distinguish their passwords from each other. They can also help minimize multiple password interference [5], where a user mixes up two passwords, or matches a correct password to an incorrect account. Encoding specificity theory [28] finds that cues are only useful if they are present at the time that the memory is encoded. Thus, for passwords, the image cue must be present at password creation for it to be most useful at password entry. Cued graphical password schemes include PassPoints [31], Persuasive Cued Click-points [6], and Image PassTiles [25].

7.3 Challenge Questions

Another way of allowing access to accounts is through the use of challenge questions (also referred to as "personal validation questions" or "secret questions"). Most familiarly seen as part of the password reset process, challenge questions are asked by the system and the user must provide the correct response [16]. The idea is that a user will be able to remember the correct answers without rehearsal, but that another person would not be able to easily guess a user's responses.

Challenge questions have been shown to have a number of problems: they typically have small password spaces, making them susceptible to statistical guessing attacks [2], and they have been shown to be vulnerable to personal attacks, where an acquaintance has enough knowledge to guess the responses [23]. Users also sometimes find it difficult to remember their responses to these questions. Questions such as "What is your favourite colour?" have a limited answer space, and users may not always have a strong and continuous preference.

Challenge questions are similar to cued passwords because the question provides a cue to the user to help them remember a response. This cue is usually text-based, but Renaud and Just [21] proposed using random image cues for challenge questions. They found that the image-based challenges were more memorable, while maintaining equivalent security.

7.4 Password Managers

One approach to handling passwords and accounts is to use a password manager. Password managers are programs that store and enter users' passwords, thus saving the user from the difficulty of remembering their passwords. Common password managers include the password managers built into web browsers (such as Mozilla Firefox), and specific password management software (such as LastPass). Browser-based password managers save passwords when they are typed into the appropriate fields, and then automatically input them when the page is visited again (often without authentication). Dedicated password managers typically work in one of two ways [7]: they either generate a password at login by hashing the user's master password together with information from the website, or they store the user's passwords in a password "wallet" which is protected by a master password (which may be required at every login or session). In either case, all accounts are compromised if the master password is lost, making the password manager a single point of failure.

Existing research on password managers has shown that they can have usability problems that affect their ability to securely manage users' passwords. Chiasson, Biddle and van Oorschot [7] conducted a study of two password managers and found that both managers had significant usability issues. Worse, participants had poor mental models for how the software worked, and these poor mental models led them to make dangerous and unrecoverable security errors.

An alternative technique for managing passwords is to reuse passwords across multiple accounts. This strategy is widely employed by users [13, 10, 15], but has security risks similar to those of a password manager. If a reused password is discovered (e.g., through a leaked password set), an attacker may be able to gain access to several accounts. However, reusing a few passwords is safer than reusing one password across all accounts. For most end users, one of the risks of reusing passwords is that they often pick weak passwords [10]. Not only are these passwords vulnerable to attack, but they weaken their password reuse strategy. If users were able to reuse a few more secure passwords, they would likely be less vulnerable to attack.

7.5 Single Sign-On

Single sign-on (SSO) systems are services that provide authentication for multiple websites. Users authenticate to an SSO provider, who checks their credentials and relays the result back to the website. A variety of architectures for SSO exist [20]. Two SSO standards are OAuth and OpenID, and well-known SSO providers include Facebook and Google.

SSO has not seen wide adoption, either by users or websites [26, 27]. Sun, Boshmaf, Hawkey and Beznosov identified this lack of adoption as being due to misaligned incentives between users, identity providers, and relying parties [26]. A study of OpenID [27] showed that users had a variety of concerns with SSO, including privacy and trust. They also found that users misunderstood how SSO worked, and that it did not fit their existing habits.

8. DISCUSSION

Versipass presents a new paradigm for password management. By integrating features of graphical passwords and password managers into the cue and category models, Versipass is a system capable of helping users better manage their passwords by taking advantage of their innate capabilities and existing habits. Our evaluation reveals a number of challenges, relevant not only to Versipass, but also to password management.

8.1 Mental Models of Versipass

One of our observations in our evaluations was that a user might be able to configure and use Versipass without properly understanding the details of how Versipass manages passwords. This raised questions about how much the user needs to understand in order to use the password manager safely and successfully. More generally, how much do users need to understand about passwords and attacks to secure themselves online?

In our previous work on graphical passwords, participants have rarely seemed bothered by the idea of an image cue. Perhaps surprisingly, they seemed not to question the presence of the image on the site. Even in online studies where the experimenter was not present, we saw little evidence that participants were not adapting to the graphical password model. However, in the evaluations of Versipass, it seemed that users were more confused by the presence of a cue that did not come from the website itself. We speculate that users misunderstand the cue model in the context of a password manager because of their preconceived idea that all password managers will remember passwords for you.

We intended that the concept of category passwords would allow users to strategically reuse passwords, much in the way that they currently reuse passwords. Our study did not ask users how they would apply this functionality to their own passwords, but participants seemed to understand that Versipass allowed the same password cue to be used for multiple accounts. However, participants in our study did not understand that those reused passwords were encoded differently for each account. Although users do not need to understand the details of how the encoding differs, we do think that they need to be more aware that they are protecting each account.

A result of our cognitive walkthrough was that category passwords might be difficult for new users to set up from scratch. We thought that an example category (with a random password) could be added to new Versipass accounts to give that users a starting point for configuring the manager.

Although we call Versipass a password manager, it actually works in a way that is fairly unlike the most common existing password managers (of course, users don't understand existing managers very well either). We want users to understand how Versipass can support their existing habits, and keep them more secure online. To do this, users need to understand Versipass well enough to use the features as intended, and to avoid making dangerous mistakes.

To address some of the misconceptions surrounding Versipass, we propose changing its title, and positioning it as a *Login Manager* rather than a password manager. Versipass effectively manages the login process: while using Versipass, a user would rely on it to manage usernames and passwords, and as a means of accessing websites. Hopefully, removing the "password manager" title would stop users from making incorrect assumptions about its functionality, and help them better focus on what Versipass can do to aid them.

8.2 Memory and Passwords

Although the difficulty of password policies is often acknowledged, the utter impossibility of these policies is too rarely acknowledged (though Florencio, Herley, and van Oorschot discuss the topic in a recent paper [12]). Humans do not have infinite memory, but there does not appear to be any limit to the number of passwords that they are asked to create and remember. No organization seems willing to say "this is too much".

Although SSO could address the problems with password policies, it is unrealistic to expect that any SSO provider will be used widely enough to address the issue for end users. Accounts are inevitably held with a number of service providers, and we cannot expect that all of those providers will be willing and able to use a single SSO provider. Instead, we must see password managers as the way of the future. Password managers can be controlled by individual users, who can make decisions about their own security. Instead of considering password managers as a flawed solution, we need to consider improvements to them.

There are a number of opportunities for password managers to better support users and take on a larger role in the password ecosystem. Password managers provide an excellent lens for users to view their accounts and better understand their web presence and overall security. Existing dedicated password managers (for example, LastPass) do provide some of this functionality, but the browser based managers do not. The internet could also better accommodate password managers. Websites could be constructed with better hooks for password managers, such as XML files providing the password policy directly to the password manager.

Another opportunity for improvement is in the nature of passwords themselves. We integrated graphical passwords into Versipass because of the benefits of cueing, which will allow users to better remember and distinguish their passwords. However, text passwords can also be cued and there are opportunities to improve the security and usability of cued text passwords.

8.3 What is a Password?

The central concept in the password manager that we think is most likely to be misunderstood is the distinction between the user input and the encoded hashed string that is sent to the website. Which of these is the "password"?

Confusion is likely to occur if we call the user's input clicks the password. A series of clicks is not obviously a password, in the way that most users think of it (as a word). However, we are reluctant to refer to the generated hashed string as the password, because users do not really see that string, and may object to the lack of knowledge about this string.

In our cognitive walkthrough, a discussion arose about locus of control. We were all able to offer anecdotal evidence that users want to have control over their passwords. Users want to pick their passwords, and they want to be able to remember them at will. Using a password manager removes a layer of control over passwords. Users must trust the password manager to consistently generate the same passwords, to save required information, but not save other information.

We speculate that users will feel unhappy about the fact that their input (clicks on a graphical password screen) is not obviously translated into the website password. Both the encoding of the click points and the hashing renders the user's password unpredictable to them. Additionally, having different salts for the same input passwords will mean that the same input will produce different strings for different websites. We suspect that users will feel uncomfortable about having not having control over what their passwords will be, or being able to remember them at will. This is a complex issue: users do not want to have to remember their passwords, but they also don't want to trust someone else to remember their passwords!

We think that one way to handle this issue is to make the salting and hashing less obvious to the user. Websites that follow best practices already salt and hash users' passwords instead of saving them in the clear. However, because this happens invisibly, users aren't bothered by it. Perhaps Versipass could achieve a similar level of comfort by hiding the salting and hashing from the user. The design decision is complicated though: we want the user to understand that their input is being transformed differently for each website they use, but we don't want them to feel upset by it, or controlled by the password manager. Automatizing and hiding security from users can have awkward consequences because processes are rarely fully hidden, and difficult cases are returned to the users who do not have sufficient information to handle the problem (for example, browsers often ask users to make a decisions about invalid SSL certificates).

Versipass needs to walk a fine line in the information it exposes to the user. The user needs to understand what the password manager is doing, but does not need to be bogged down in understanding the details.

9. CONCLUSION

We have developed Versipass, a new approach to password management. Versipass involves the novel combination of two models: the cue model and the category model. Together, these models support the user by providing cues for strong passwords and supporting the safe reuse of each of the passwords across multiple websites. In this paper, we presented the models and our prototype implementation together with an evaluation.

We suggest that Versipass constitutes a new paradigm for password management by allowing users to create stronger passwords, better remember those passwords, and use a password manager without creating a single location where passwords are stored.

Our prototype implementation and our evaluations have numerous limitations. The current prototype reused code from an earlier project, which impacted both usability and security. However, the prototype allowed us to explore the ideas of the cue and category models, and our evaluations showed where improvements are necessary.

Future work in this area includes the development of a higher fidelity prototype in the form of a browser extension, or as an integrated browser feature. To better understand users' real world behaviour, we would like to conduct more ecologically valid tests of this prototype. In particular, we would like to explore how password managers can fit into the larger picture of users' password management strategies and coping techniques. This kind of testing will require great care because of the sensitivity and privacy of users' password and account data.

Versipass has brought up several issues involving passwords and password managers. One issue is that the concept of a "password" is becoming more fluid. On the one hand, graphical passwords largely hide what the actual passwords are, but on the other hand, even normal passwords are in fact saved in salted hashed form by any responsible system.

Any knowledge-based authentication must take into account human capabilities. In recent years, it has become the norm for people to have dozens of accounts, and we need to acknowledge the impossibility of humans remembering strong passwords for each and every one. Some technology is necessary to bridge this divide. Two approaches are single sign-on and traditional password managers. We suggest that our approach offers new advantages and harnesses users' abilities and current coping practices.

10. ACKNOWLEDGMENTS

We would like to thank the Government of Canada for support from NSERC ISSNet and the GRAND NCE, as well as support from an NSERC Canada Graduate Scholarship, and an NSERC Discovery Grant. We would also like to thank the conference attendees for their thoughtful discussion and feedback, as well as the anonymous reviewers and our shepherd for their help. Additionally, we would like to thank Furkan Alaca and Paul van Oorschot for their helpful comments that improved this paper.

11. REFERENCES

[1] R. Biddle, S. Chiasson, and P. C. van Oorschot. Graphical Passwords: Learning from the First Twelve Years. *ACM Computing Surveys*, 44(4), 2012.

[2] J. Bonneau, M. Just, and G. Matthews. What's in a Name? In *Financial Cryptography and Data Security*, pages 98–113. Springer, 2010.

[3] W. Cheswick. Rethinking Passwords. *Queue*, 10(12), Dec. 2012.

[4] S. Chiasson, C. Deschamps, E. Stobert, M. Hlywa, B. Freitas Machado, A. Forget, N. Wright, G. Chan, and R. Biddle. The MVP Web-based Authentication Framework. In *Financial Cryptography and Data Security*, pages 1–8, Bonaire, Feb. 2012.

[5] S. Chiasson, A. Forget, E. Stobert, P. C. van Oorschot, and R. Biddle. Multiple password interference in text passwords and click-based graphical passwords. In *CCS '09: Proceedings of the 16th ACM Conference on Computer and Communications Security*, Chicago, USA, Nov. 2009. ACM.

[6] S. Chiasson, E. Stobert, A. Forget, R. Biddle, and P. C. van Oorschot. Persuasive Cued Click-Points: Design, Implementation, and Evaluation of a Knowledge-Based Authentication Mechanism. *IEEE Transactions on Dependable and Secure Computing*, 9(2):222–235, Oct. 2011.

[7] S. Chiasson, P. van Oorschot, and R. Biddle. A Usability Study and Critique of Two Password Managers. In *15th USENIX Security Symposium*, pages 1–16, Vancouver, Canada, 2006.

[8] A. Das, J. Bonneau, M. Caesar, N. Borisov, and X. F. Wang. The Tangled Web of Password Reuse. In *NDSS '14: The Network and Distributed System Symposium*, San Diego, USA, 2014.

[9] 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*, 63(1-2):128–152, July 2005.

[10] D. Florêncio and C. Herley. A Large-Scale Study of Web Password Habits. In *International World Wide Web Conference Committee (IW3C2)*, pages 1–9, Banff, Canada, May 2007.

[11] D. Florêncio, C. Herley, and P. C. van Oorschot. An Administrator's Guide to Internet Password Research. In *USENIX LISA*, 2014.

[12] D. Florêncio, C. Herley, and P. C. van Oorschot. Password Portfolios and the Finite-Effort User: Sustainably Managing Large Numbers of Accounts. In *23rd USENIX Security Symposium*, San Diego, USA, Aug. 2014.

[13] S. Gaw and E. W. Felten. Password Management Strategies for Online Accounts. In *SOUPS '06: Proceedings of the 2nd Symposium on Usable Privacy and Security*, Pittsburgh, USA, July 2006. ACM.

[14] Hacker News. I'm the Chrome browser security tech lead, so it might help if I explain our rea... | Hacker

News, 2013.
`https://news.ycombinator.com/item?id=6166731`.

[15] E. Hayashi and J. Hong. A diary study of password usage in daily life. In *CHI '11: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, Vancouver, Canada, May 2011. ACM.

[16] M. Just. Designing and evaluating challenge-question systems. *IEEE Security & Privacy*, 2(5):32–39, 2004.

[17] D. A. Norman. When security gets in the way. *ACM SIGCSE Bulletin*, 16(6):60, Nov. 2009.

[18] A. Paivio. *Imagery and Verbal Processes.* Holt, Rinehart, and Winston, 1971.

[19] A. Paivio, T. B. Rogers, and P. C. Smythe. Why are pictures easier to recall than words? *Psychonomic Science*, 11(4):137–138, 1968.

[20] A. Pashalidis and C. J. Mitchell. A Taxonomy of Single Sign-On Systems. In *Information Security and Privacy*, pages 249–264, Berlin, Heidelberg, June 2003. Springer.

[21] K. Renaud and M. Just. Pictures or questions?: examining user responses to association-based authentication. In *BCS '10: Proceedings of the 24th BCS Interaction Specialist Group Conference*, pages 98–107, Dundee, UK, 2010. BCS.

[22] B. Ross, C. Jackson, N. Miyake, D. Boneh, and J. Mitchell. Stronger Password Authentication Using Browser Extensions. In *14th USENIX Security Symposium*, Baltimore, USA, Aug. 2005.

[23] S. Schechter, A. J. B. Brush, and S. Egelman. It's No Secret. Measuring the Security and Reliability of Authentication via "Secret" Questions. *IEEE Symposium on Security and Privacy*, pages 375–390, 2009.

[24] B. Schneier. Write Down Your Password, June 2005. `http://www.schneier.com/blog/archives/2005/06/write_down_your.html`.

[25] E. Stobert and R. Biddle. Memory retrieval and graphical passwords. In *SOUPS '13: Proceedings of the 9th Symposium on Usable Privacy and Security*, Newcastle, UK, 2013. ACM.

[26] S.-T. Sun, Y. Boshmaf, K. Hawkey, and K. Beznosov. A billion keys, but few locks. In *NSPW '10: The New Security Paradigms Workshop*, pages 61–72, USA, 2010. ACM.

[27] S.-T. Sun, E. Pospisil, I. Muslukhov, N. Dindar, K. Hawkey, and K. Beznosov. What Makes Users Refuse Web Single Sign-On?: An Empirical Investigation of OpenID. In *SOUPS '11: Proceedings of the 7th Symposium on Usable Privacy and Security*, Washington DC, USA, 2011. ACM.

[28] E. Tulving and D. Thomson. Encoding Specificity and Retrieval Processes in Episodic Memory. *Psychological Review*, 80(5):352–373, Dec. 1973.

[29] P. C. van Oorschot and J. Thorpe. Exploiting Predictability in Click-based Graphical Passwords. *Journal of Computer Security*, 19(4):669–702, 2011.

[30] C. Wharton, J. Rieman, C. Lewis, and P. Polson. The cognitive walkthrough method: A practitioner's guide. In J. Nielsen and R. L. Mack, editors, *Usability Inspection Methods*, pages 105–140. John Wiley & Sons, Inc., New York, NY, USA, 1994.

[31] S. Wiedenbeck, J. Waters, J.-C. Birget, A. Brodskiy, and N. Memon. PassPoints: Design and longitudinal evaluation of a graphical password system. *International Journal of Human-Computer Studies*, 63(1-2):102–127, July 2005.