Pass: Online Authentication Made Simple

Michelle Lee
Cornell University
tl428@cornell.edu

Jim Li
Cornell University
zl238@cornell.edu

Andrew Semmes
Cornell University
als452@cornell.edu

Zhaoxing Wu
Cornell University
zw479@cornell.edu

Abstract

Passwords are currently the primary method for authenticating users online. While this method of authentication offers many advantages, the user experience of using passwords presents many opportunities for exploration and areas for improvement. In this paper, we sought to identify one such area by first exploring different alternatives to passwords or methods of using passwords, and then proposing a new password experience called *Pass*. *Pass* is designed using an iterative HCI design process that relies on user interviews and user testing. The main features include helping users create different password profiles for each type of passwords they use, and logging them into websites automatically by tracking a combination of their characteristics and browsing habits. After three iterations of the design, users were largely satisfied with the tool's user experience, though there is still room for improvement.

Author Keywords

Passwords; online security; user experience; usability; HCI.

ACM Classification Keywords

Human-centered computing; HCI; Interaction design; interaction design process and methods; user interface design.

Introduction

The use of passwords as a means of Internet authentication continues to multiply with the ever increasing online activities. Passwords have persisted to be the go-to authentication method for online accounts protection. They are affordable for service providers to integrate into their websites and are readily understood by users [1]. Instantaneous account setup, simple passwords reset, and convenient access have contributed to the persistent use of passwords.

However, social engineering, phishing attacks, and shoulder surfing are some of the common methods deployed by hackers to undermine the effectiveness of password-protected accounts [2]. Despite the availability of more secure authentication alternatives such as two-factor authentication [3], graphical schemes [4], and biometric authentication[5], they have failed to become widely adopted. This is because these methods either require additional user efforts and buy-in or incur additional costs for service providers, creating sufficient frictions for mainstream adoption [6]. Economics and usability are more likely than technological advancements to be the primary drivers of authentication changes.

Given the persistence of passwords and the call for more usable authentication, we aim to explore how we can improve the usability of passwords by suggesting a new password experience, *Pass*, and the implications for future research.

Formative Work

Our research began with 6 informal contextual inquiries with Cornell students. The interview subjects consisted of students from engineering and non-engineering backgrounds to balance a varying degree of understanding of passwords authentication.

Preliminary User Interviews

Interview subjects were selected at random in two locations: Duffield Hall and Mann Library. Each interview was intended to explore the subject's attitudes towards cybersecurity and passwords, experience with using passwords, and strategy for creating and managing passwords. Furthermore, we probed the interview subjects' "likes and wants" for an ideal passwords experience.

Below are the interview highlights:

1. Password memorization and reuse were common practices. Users did this out of convenience and some believed this to be safer than saving passwords to a third party tool like a browser keychain or a password manager.
2. Users understood the risk of having personal information stolen but were not concerned enough to protect their passwords with additional safeguards.
3. Many users varied their password complexity by perceived importance. For example, they create a more complex password for financial accounts than social media accounts.
4. Few users used password managers. They did not find enough values from password managers to warrant paying for them.

Affinity Diagram and Requirements

We created an affinity diagram (Figure 1) to organize the findings. The diagram was organized by "strategy", "attitudes", and "wants and likes". In general, users came up with a relatively complex password and would

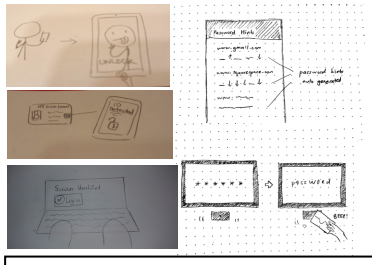

Figure 2: A collage of sketches from our brainstorming

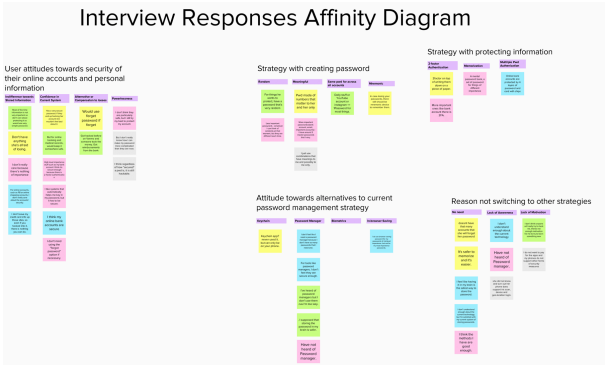

Figure 1: Affinity Diagram

reuse or vary a few characters in the password for other accounts. The lax attitudes towards passwords stemmed from the lack of perceived consequences for the loss of access to these accounts. That said, most users expressed if there was a more convenient way to authenticate accounts they would be willing to create more diverse and complex set of passwords.

We therefore decided on a preliminary list of requirements to guide our initial designs:

1. Focus on the "convenience factor" of authenticating accounts using passwords.
2. Address users' tendency to have different passwords for accounts with varying degrees of security.
3. Provide multiple options for authentication.
4. Reduce the cognitive load of passwords memorization for different accounts.

Low Fidelity Prototypes: *SmartLogin* & *AutoRegister*

We generated many preliminary ideas through our brainstorming session (Figure 2). By considering the requirements we set in the previous section, we boiled down our ideas down to two that most closely aligned prototypes:

SmartLogin

SmartLogin (Figure 3.1) is an in-built browser capability that uses machine learning to automatically logs users habits. The user will be authenticated or denied based on a hidden score that is calculated from a combination of different factors. From the brainstorming session, we identified 6 potential characteristics that would go into this score: facial tracking, keystrokes, browsing habits, trusted devices, location, and voice recognition. While each factor alone would be insufficient for determining a person's authenticity, a combination of some or all of them would allow the score to be high enough to authenticate the user.

The prototype was a simple interface that showed users a terms and conditions page asking them to consent to the browser tracking their browser usage data in order to verify their identity and automatically log them into their websites. There is also a simple screen that showed the user that they would be automatically logged in on whatever site they use.

AutoRegister

AutoRegister (Figure 3.2) is a browser feature that helps users create different password profiles for each type of password that they would usually use. For example, the user could create one password profile for "Low Importance Sites" and a different one for "High Importance Sites". This prototype was designed in response to users having different groups of passwords for different types of sites.

Unlike a password manager that generates a random password for each site, *AutoRegister* uses passwords that are already familiar to the user. When the user encounters an account creation form on a website, *AutoRegister* automatically prompts the user to select an existing profile to fill the form with. This eliminates

the user's dependency on the tool to remember passwords, since each password is defined beforehand by the user.

User Testing Session #1

We conducted 6 user testing sessions to evaluate the pros and cons of the low fidelity prototypes. Users found both prototypes to be an improvement to existing solutions, especially the experience with account logins. Some users, however, expressed concerns of being tracked by the underlying machine learning based system. Users were also confused by the design of *AutoRegister*, whose levels of importance selection did not fulfill users' needs for having different passwords for accounts with the same level of importance.

High Fidelity Prototype: *Pass*

The two low-fidelity prototypes addressed different users' needs and were not mutually exclusive. Instead, they complemented each other. *SmartLogin* addressed the convenience factor of entering passwords, while *AutoRegister* addressed the convenience factor of creating and generating passwords. In addition, since *SmartLogin* still required users to create a password for websites in the first place, *AutoRegister* filled that gap perfectly. Hence, we combined the 2 prototypes into a comprehensive single solution, *Pass*.

Easy Accessibility

Pass is implemented as a browser extension, inspired by the password-saving feature native to most browsers, as opposed to a desktop program. This design decision was drawn from findings in our initial round of user interviews, in which many users did not bother to use a password management tool but used the password-saving feature of browsers. We posit that this is because the password-saving feature of a browser is readily understood, highly accessible, and easy to use.

Aiding the User's Mental Models

Our interviews and user tests revealed several users who were concerned about how *SmartLogin* worked. To increase the user's confidence in the system, we incorporated short descriptions below each tool to explain not only what they do, but how they work. Many users were also concerned about the quantity of private data being tracked. To alleviate this concern, we placed a message in bold above the permissions settings saying that all data collected would be encrypted and never divulged.

Sense of Control

When accessed, the user is brought to the main settings page. Following Jakob Nielsen's user interaction heuristic of offering the user an control and freedom with an "emergency exit", we added a simple switch to easily turn *SmartLogin* and *AutoRegister* on and off. The user is also able to easily uncheck *SmartLogin's* permissions options to disable the tracking of specific data. The permissions option are presented front-and-center on the *SmartLogin* settings screen, and requires no drilling down into the settings hierarchy to change.

System Status Visibility

Since this is a tool that runs in the background, system status visibility is concealed from the user unless it is deliberately shown through the interface. We incorporated popup elements that provided both feedback and action controls for the user as s/he browses the web. The settings page for *SmartLogin* also reflects its current calculated effectiveness based on which boxes have checked by the user. If too few boxes have been checked, the interface prompts the user for action with red error messages while recommending appropriate actions to resolve the error.

Familiar & Consistent UI Elements

We heavily borrowed on design elements and concepts that would be familiar to most users of the internet. For example, the style of the pop ups in the prototype was

kept consistent with that of other extensions. *AutoRegister's* account creation autofill feature highlights textboxes after filling them, similar to many login autofill features we have today. *AutoRegister*'s password profiles uses grabbing icons similar to what is found on phone interfaces to afford drag-moving.

Aesthetic Design
To invoke the user's perception of *Pass* as a technologically advanced and "cool" tool, we came up with a design that was futuristic, but also clean and familiar. The color palette is simple, primarily using a cerulean and turquoise gradient along with black, white and grey for most other UI elements. The popups are non-invasive and minimalistic, featuring simple text and information

User Testing Session #2
We conducted another five user testing sessions with our mid-fidelity prototype. Again, we recruited Cornell undergraduates to be our users. We presented them with *Pass* and asked them to explore different features of the prototype.

We found that users felt more confident about *Pass* when it was transparent in terms of what information it collects. Although *Pass* included a security level indicator, users felt uneasy when they did not pick the features combination that could offer the highest level of security. Users also brought up security concerns regarding unsuccessful login, as the mid-fidelity prototype only required users to click on a reCaptcha box for identity verification after unsuccessful login. Even though machines could not log into the users account, someone who can access the user's computer would still be able to imitate the user's mouse movement, click on the box and logged in.

Iteration 2 of *Pass*
According to our second user test, our users were satisfied with the general functionality of *Pass* but were concerned about failed login and whether they could take full advantage of the tool. Thus, we decided to add an extra layer of authentication when the users failed to login automatically. We added a checkbox for users to click on. When they move their mouse, *Pass* would be able to gather additional information on the users' mouse movement habit and could use those data for additional machine learning analysis and offer a second attempt to automatically log user in. Moreover, instead of just indicating the strength of the feature combinations for secured login, we updated the security bar such that users would not be allowed to be logged in automatically if a weak feature combination is selected. This could ensure *Pass* provide a more secured login experience as well as to ease the users' anxiety of not using a strong enough combination.

User Testing Session #3 - The Final Round
From our final round of user testing, our four Cornell undergraduate student users gave us primarily positive feedback from our high fidelity prototype. Concerns from the previous round of interviews were largely resolved as users could see and understand how the system worked and tracked their data from the settings page. Our interviewees said that our high fidelity prototype was easy to use and intuitive, and when pressed for design problems or issues they could not give us any. When further questions on how we could improve the system, interviewees only brought up that some users may be uncomfortable with the system tracking their information, however they themselves were did not mind.

More specifically about the prototype, users liked that the automatic password login was more convenient to use than their current system of saving passwords on their phones or memorizing them. They all said that they would use this feature if it was included in their browser. One user in particular remarked that he liked that if the system was unsure of his verification at all, then the system would revert to the regular sign in page. He said that this implementation would allow for all the added convenience of the machine learning

verification and automatic login, without compromising security. Another one of our interviewees particularly like the auto-registration feature. She said that she uses the same two or three passwords on all her sites, so she liked that the system automated that habit for her.

Final Iteration of *Pass*

Due to the largely positive feedback from our users, we did not change our final prototype much from its previous iteration. Our interviewees felt comfortable with our design, and enjoyed the ease of use for logging into the prototype's artificial website.

Conclusion

Pass aims to provide users with a seamless login and registration experience. It is our hope that *Pass* can provide users with the benefit of high security without the sacrifice of convenience. The physical and behavioral biometrics that *Pass* tracks are not set in stone. Future work will concern deeper analysis of what biometric combinations will provide the most accurate and reliable authentication.

Acknowledgements

We would like to thank Professor Gilly Leshed and our Teaching Assistant Jiajing Guo for their help on this project.

References

1. Cormac Herley and Paul van Oorschot. 2012. A research agenda acknowledging the persistence of passwords. *IEEE Computer and Reliability Societies*: 28-36.
2. Steven Furnell and Leith Zekri. 2006. Replacing passwords: In search of the secret remedy. *Network Security*: 4-8.
3. Bruce Scheiner. 2005. Two-factor authentication: Too little, too late. *Communications of the ACM*: 136.
4. Susan Wiedenbeck, Jim Waters, Jean-Camille Birget, Alex Brodskiy, and Nasir Memon. 2005. PassPoints: Design and longitudinal evaluation of a graphical password system. *International Journal of Human-Computer Studies*: 102-127.
5. Lawrence O'Gorman. 2003. Comparing passwords, tokens, and biometrics for user authentication. *Proceedings of the IEEE*: VOL. 91, NO. 12.
6. Cormac Herley, P.C. van Oorschot, and Andrew S. Patrick. 2009. Passwords: If we're so smart, why are we still using them? *Proceedings of Financial Cryptography and Data Security.*