Confused Johnny: When Automatic Encryption Leads to Confusion and Mistakes

Scott Ruoti, Nathan Kim, Ben Burgon, Timothy van der Horst, Kent Seamons
Internet Security Research Lab
Computer Science Department
Brigham Young University
Provo, Utah, USA
{srouti, nkim, bburgon, timv}@isrl.cs.byu.edu, seamons@cs.byu.edu

ABSTRACT

A common approach to designing usable security is to hide as many security details as possible from the user to reduce the amount of information and actions a user must encounter. This paper gives an overview of Pwm (Private Webmail), our secure webmail system that uses security overlays to integrate tightly with existing webmail services like Gmail. Pwm's security is mostly transparent, including automatic key management and automatic encryption. We describe a series of Pwm user studies indicating that while nearly all users can use the system without any prior training, the security details are so transparent that a small percentage of users mistakenly sent out unencrypted messages and some users are unsure whether they should trust Pwm. We then conducted user studies with an alternative prototype to Pwm that uses manual encryption. Surprisingly users were accepting of the extra steps of cutting and pasting ciphertext themselves. They avoided mistakes and had more trust in the system with manual encryption. Our results suggest that designers may want to reconsider manual encryption as a way to reduce transparency and foster greater trust.

Categories and Subject Descriptors

H.5.2 [**Information Interfaces and Presentation (e.g. HCI)**]: User Interfaces—*user-centered design, evaluation*

General Terms

Design, Human Factors, Security

Keywords

Usable security, secure email, manual encryption, security overlays

Symposium on Usable Privacy and Security (SOUPS) 2013, July 24–26, 2013, Newcastle, UK.

1. INTRODUCTION

Secure email solutions exist but have not been widely adopted. Research indicates that this is due in part to usability issues, especially in the areas of key management and portability [19, 15]. These issues are a significant impediment to secure webmail, as users expect high levels of usability and portability from their webmail systems. We believe that users will adopt a secure webmail system only if it is tightly integrated with their existing webmail systems in order to maintain the usability and convenience they are accustomed to. If secure webmail becomes a burden to users, they will reject it and choose instead to focus on their primary goal to send and receive email.

This paper presents results and lessons learned from usability studies of Pwm (Private WebMail, pronounced "Poem"), our solution to extend existing webmail systems (Gmail, Hotmail, Yahoo! Mail) with end-to-end encryption and message integrity. Pwm's security is mostly transparent; key management details are hidden and users are never exposed to ciphertext. Pwm was designed to maximize usability so that users would be willing to adopt it. Pwm integrates tightly with webmail providers' interfaces using security overlays, reducing the burden a user feels towards learning a new system.

The first research question addressed in this paper is how usable is Pwm's tight integration with existing webmail systems using security overlays and its transparent encryption. Pwm was designed to help everyday users send and received encrypted email with little or no training. We conducted IRB-approved user studies of Pwm where nearly all participants were able to decrypt secure messages sent to them without any prior notice or training.

However, these user studies revealed two concerns: First, some users did not trust that the system was secure because the security details (key management and encryption) were so transparent that they did not have a clear idea about how the system actually worked. Second, a small but consistent number of users accidentally sent plaintext when they intended to communicate a sensitive message. Since the steps a user takes to send a message are quite similar for both encrypted and unencrypted data, a user's "click-whirr" response makes them susceptible to sending sensitive messages without first enabling Pwm [5]. Several users realized their mistake immediately after they sent the message, but the damage was already done.

These problems caused us to reconsider hiding some of the security details. The second research question addressed in

this paper is whether manual encryption in an application separate from the browser would prevent users from mistakenly sending out sensitive information without encryption, compete with Pwm in terms of usability, and reveal enough security details that users would have greater trust in the system. To test this hypothesis, we built Message Protector (MP), a mockup that included manual encryption in an application separate from the browser.

We conducted two more IRB-approved user studies using MP. We were surprised that users rated MP with manual encryption to be as usable as Pwm. Users also had more trust in MP with manual encryption and avoided the mistake of sending out sensitive messages unencrypted. However, more users preferred that security systems be tightly integrated with the browser. Thus, in the effort to balance security and usability, we argue that a combination of exposing some encryption details and tight integration will produce a system that users trust and help them to secure their data without making mistakes.

The remainder of this paper is organized as follows: Section 2 describes Pwm, and Section 3 presents the user studies of Pwm. Section 4 describes MP, and Section 5 presents the user studies of MP. Section 6 presents the limitations of our user studies and Section 7 discusses related work. Section 8 contains conclusions and future work.

2. PWM

Based on earlier research and our experience, we believe that there are three problems inhibiting the adoption of secure email by the masses:

1. Users are resistant to change. If secure email requires too much effort for the perceived benefits it will be rejected by users [11].

2. Users do not understand how to obtain, distribute, or use cryptographic keys [19, 15]. Additionally, PKI-based secure email has a chicken and egg problem, as most users will not perform key management until they have received an encrypted email, and users cannot receive an encrypted email until they perform key management.

3. Users are confused by the details of cryptography [19]. This leads users to omit or incorrectly use various cryptographic operations necessary for securing email.

We hypothesized that if these difficulties were overcome, users would be able to successfully use secure webmail and be willing to adopt it. Based on this hypothesis, we developed Pwm (Private WebMail, pronounced "Poem"). Pwm adds end-to-end encryption and message integrity to existing webmail systems (Gmail, Hotmail, Yahoo! Mail) and runs in all major browsers (Chrome, Firefox, Internet Explorer, Opera, and Safari). Pwm is designed to maximize usability and provide additional security to users who are already sending sensitive information over email. Pwm addresses the problems we identified as follows:

1. Pwm tightly integrates with existing webmail systems using *security overlays*, windows placed over the webmail providers interface that allow users to interact with secure content. Security overlays are functionally transparent to users helping avoid the frustration of learning a new system.

2. Key management is automatic and fully transparent to users. Keys are managed by a key escrow server that uses email-based identification and authentication (EBIA [8]) to authenticate users without their interaction.

3. All encryption is handled automatically by Pwm, and users are never directly presented with ciphertext or the details of encryption.

2.1 Security Overlays

Pwm uses security overlays to tightly integrate new security features into existing webmail interfaces. A security overlay is a window where users view and interact with secure content. It is positioned directly over the portions of the webmail provider's interface that need to be secured. The user interacts with the security interface in lieu of the overlaid portion of the webmail provider's interface. A security overlay is displayed using an iFrame and uses the browser's same domain policies to protect its contents from access by the honest-but-curious webmail provider.

Security overlays are designed to be functionally transparent to users, matching the functionality that exists in the overlaid portion of the webmail provider's interface. This functional transparency allows users to complete tasks in the way they are accustomed to, lowering the chance that users will disable the secure system to more readily complete their tasks. Security overlays are also designed to be visually distinctive from the webmail provider's interface. This distinction assists users in determining whether they are using a security overlay or the webmail provider's original interface and highlights features unique to the security overlay.

For example, Figure 1 is an encrypted Pwm email and Figure 2 is that same message after it has been decrypted. The security overlay has been positioned in the page where users expect to read email. Functionally, it is identical to reading any other message, but visually it is distinctive and allows users to quickly identify when they are reading encrypted emails. We avoid visual transparency as that would prevent users from determining when the system is in use and reduce trust in the system [7].

2.2 Key Management

A key escrow server handles key management. The key escrow server follows the principles of identity-based cryptography (IBC) introduced by Shamir [14] in that cryptographic keys are generated based on users' identities (i.e., email address). This model allows users to send encrypted email to recipients who are currently outside the system. Unlike IBC, the key escrow system uses symmetric key cryptography and key derivation [4, 12] instead of public key cryptography. The advantages of key escrow are (1) key management can be fully automated, (2) users can never lose their encryption keys, and (3) keys can be automatically ported to new devices. The disadvantage of key escrow is that the key escrow server has access to users' keys, which is a recognized trade-off to get the other usability benefits [1].

Pwm interacts with the key escrow server using an invisible key management security overlay. This security overlay handles all key management operations (e.g., obtaining and storing keys, authentication). Authentication is handled by Simple Authentication for the Web (SAW [18]), a form of email-based identification and authentication (EBIA [8]).

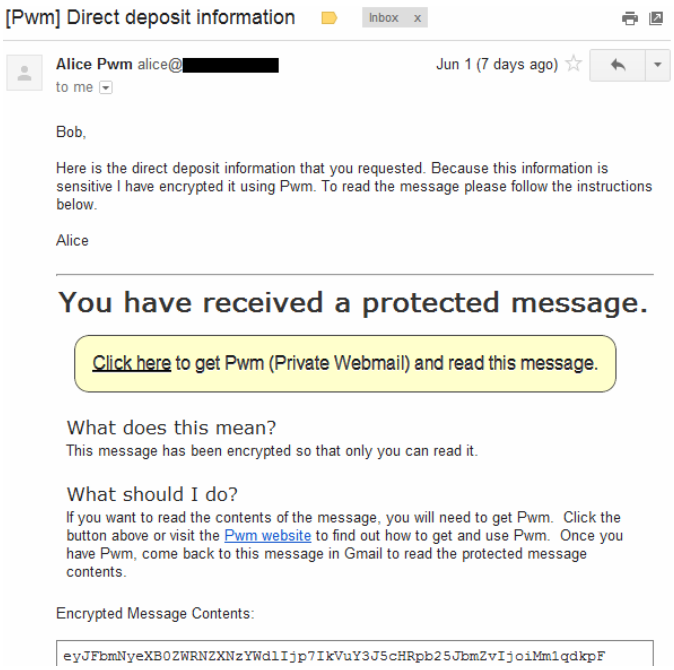

Figure 1: A sample email prior to decryption

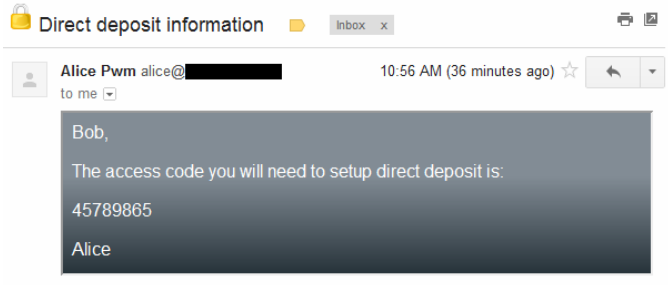

Figure 2: Decrypted email

SAW generates an authentication token for the key server and then splits it in half. One half is returned to the requester and the other half is sent to email account that is being authenticated. Pwm runs inside the webmail provider and has access to this email, allowing Pwm to obtain the full authentication token without reliance on user input. The combination of key escrow and SAW allows for transparent and automatic key management, removing many of the difficulties users faced with traditional secure email solutions.

2.3 Automatic Encryption

Pwm hides almost all security details from users. Encrypted Pwm emails include ciphertext, but it is positioned so that it will be largely ignored by users (Figure 1). The most prominent portion of an encrypted email is the instructions for setting up Pwm. These instructions are designed to help first time users obtain the software needed to decrypt the email. Optionally, the sender of an encrypted email can add a personalized message explaining the nature of the encrypted message and the need to obtain Pwm to access it. This message can provide important context so the recipient can trust that the message is legitimate.

Once Pwm is installed and running, it automatically decrypts the email and displays the plaintext contents of the message to the user in a security overlay (Figure 2). If they had opened the email with Pwm already running, they would only see the decrypted message and not the encrypted email. Users can detect that they are reading a decrypted message because of the visual distinctiveness of security overlays and the addition of a lock icon to the subject of the message (Figure 2 and Figure 3)

When a user replies to an encrypted email, their response is automatically encrypted for them. Unlike replies, new messages are not encrypted by default. Instead, users are presented with an open lock icon next to the email compose form that must be clicked in order to activate the security overlay for composing encrypted email (Figure 4). When users click the send button on the security overlay, the message is encrypted and sent automatically without ever showing them ciphertext. We take this user-centric approach to maximize the impact encrypted emails have on users by reserving encryption for when it is needed [10]. Since new users must install software in order to read Pwm messages, it limits the number of new users required to install Pwm to those that need to access sensitive messages. It also helps to minimize the impact on webmail's ad-based revenue model. Encrypted email limits the webmail provider's ability to scan user's messages in order to serve targeted ads. If default encryption produced a surge of encrypted emails, this might cause a webmail provider to block Pwm traffic or actively limit Pwm's ability to tightly integrate with the webmail interface. For convenience, a user can turn on encryption permanently for a given recipient that they always desire to communicate with securely.

2.4 Setup

The prototype can be installed and run using either a browser extension or a bookmarklet. A browser extension is a well-known method for adding functionality to the browser, but bookmarklets are a newer and increasingly popular method for doing the same thing. A bookmarklet is simply a browser bookmark that contains JavaScript instead of a URL. The bookmarklet and the browser extension both function by inserting the in-page services script tag onto the webmail provider's web page. The only difference between the two is that the browser extension is always running, while the bookmarklet must be clicked each time the user visits Gmail.

Bookmarklets have several advantages in comparison to browser extensions, the most important being ease of setup. On the prototype's website, the bookmarklet is represented by a large button with the text "Secure My Email". Installation is as simple as dragging this bookmarklet to the bookmarks bar. Bookmarklets are also quick and easy to use, whenever Bob wants to run the prototype he only needs to click the "Secure My Email" bookmarklet in his bookmarks bar.

As demonstrated by the success of Pinterest,[1] average users are able to set up and use bookmarklets with little difficulty. Since this does not qualify as installation in the traditional sense, Bob does not need administrative privileges to use it. Furthermore, the prototype can be set up and run on any computer where Bob accesses webmail.

[1]Pinterest is a website that makes heavy use of bookmarklets. `http://pinterest.com/`

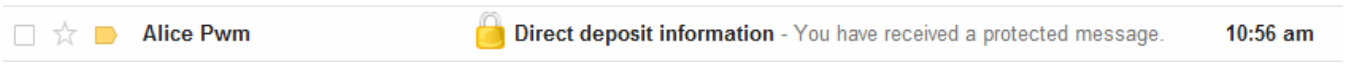

Figure 3: New Pwm email in inbox

Figure 4: Compose interface

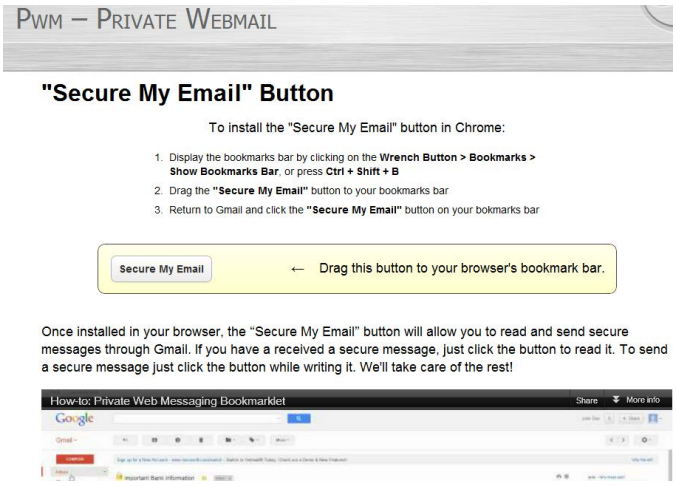

Figure 5: Pwm website

3. PWM USER STUDIES

We conducted two IRB-approved user studies to evaluate the usability of Pwm. The goal of these studies was to test whether Pwm's design would lead to secure email that was both usable and desirable for users. This included determining whether new, untrained users could set up and use Pwm relying only on the directions provided in the plaintext portion of the encrypted email (Figure 1) and Pwm's website (Figure 5). We also wanted to discover what, if anything, caused users to fail when sending and receiving encrypted email. Finally, we wanted to determine users' opinions toward the tight integration provided by security overlays.

The participants for both studies were recruited at Brigham Young University using posters that invited students to participate in a Gmail usability study, but did not alert them that it was related to security. To minimize unfamiliarity with Gmail impacting our results, we stipulated that volunteers for the study should be active Gmail users. We also indicated that the study would take approximately thirty minutes and provide compensation of $10.

During this study, all participants used the same computer[2]. We provided Gmail accounts to participants to use in the study. This allowed us to preserve the privacy of the participants' personal accounts and furthered our ability to provide a uniform environment. Participants were required to complete the study using the Google Chrome Web browser.To match a fresh install of Google Chrome, we ensured that the bookmarks bar was not displayed initially. Before beginning, participants completed a demographic questionnaire (Appendix A.1). Users were not required to identify themselves, and we did not record the identity of any participant.

[2]3.0 GHz Intel Core 2 Quad CPU with 8 GB of RAM

3.1 Bookmarklet Study

3.1.1 Setup

This study used the bookmarklet version of Pwm and was comprised of 25 students, representing 19 different majors, and with low to medium technical experience. Of the 25 participants, 19 (76%) had been Gmail users for over a year and only 3 (12%) had been Gmail users for less than 6 months. Twenty-three (92%) of the participants used Gmail on a daily basis.

We remotely monitored each user's actions in real-time using RealVNC and recorded their actions locally using CamStudio. Participants were presented with simple tasks to complete using Pwm (Appendix A.2). After completing the tasks, participants were presented with a short survey about their experience using Pwm (Appendix A.3). We then augmented this survey with a brief interview in which we asked each participant about difficulties or failures we had observed.

3.1.2 Tasks

Each participant was given three tasks to complete using Pwm. These tasks were designed to simulate what an individual would experience if they received an unsolicited Pwm email and began using Pwm.

In the first task, participants were told to check their inbox for an email containing instructions on how to proceed with the study. Unknown to them, this email had been encrypted using Pwm. Participants were given no explanation or help from the study conductor and were required to rely only on the directions provided by the encrypted Pwm email. Once decrypted, the email instructed participants to send an encrypted reply and return to the study instructions. The primary goal of this task was to observe whether untrained users could successfully set up and use Pwm with no outside assistance. Because we were in a lab environment where participants knew they would not be exposed to any real risks, we refrained from drawing any conclusions about participants' trust in bookmarklets.

For the second task, participants were asked to open a new Gmail session, send an encrypted email to the study conductor, and then wait for a reply with further instructions. If participants did not encrypt their email, they would then receive an unencrypted reply informing them that their email had not been encrypted and instructing them to try again. Once the participant successfully sent an encrypted email, they received an encrypted reply instructing them to close Gmail and return to the study instructions.

The third task required a new Gmail session be started. Since Pwm was no longer running, the participant would need to restart Pwm by clicking on the bookmarklet. The primary goal of this task was to determine whether participants would be able to correctly restart Pwm in order to compose an encrypted email.

3.1.3 Results

Overall, participants were highly successful in using Pwm. All but one of the 25 participants (96%, Confidence Interval (CI) at 95%, ±7.68) successfully set up Pwm and decrypted the email received in the first task. The only participant who failed to decrypt the email had correctly set up Pwm but then moved on to the second task without trying to read the decrypted email. When asked why she did this, she said that it was because she assumed the task was complete once she had added the bookmarklet. This was a flaw in the task setup because we should have had information contained in the encrypted message that participants needed to report in order to continue on with the study. This would also have more closely resembled real world use cases.

Of the 24 that decrypted the email in the first task, 23 (96%, CI ±7.84) successfully sent an encrypted reply. The only participant who failed to send the encrypted reply had correctly used Pwm but then clicked Gmail's "Compose" button rather than the "Send" button. He did not repeat this error on the second task. When asked about this, he said that he was accustomed to using Gmail on his iPod Touch where the send button is in the upper left-hand corner of the screen where the "Compose" button was in our test.

On the third task, 22 participants (88%, CI ±12.74) successfully sent an encrypted email on their first try. Of the three who failed, one immediately recognized his mistake and correctly sent an encrypted email before receiving a reply. When asked about this, he reported that he knew it wasn't encrypted when he didn't see the security overlay's black background. The remaining two participants successfully sent an encrypted email after receiving the reply asking them to try again. One of the two stated that they had misread the instructions and didn't realize they were supposed to encrypt the email. The other reported that he didn't realize he needed to click the bookmarklet again and said that he wouldn't repeat that mistake again.

3.1.4 System Usability Scale

We used the System Usability Scale (SUS) [3], a usability evaluation metric developed at Digital Equipment Corp., to rate the usability of Pwm. SUS works by asking participants to respond to ten statements on a Likert scale. We included these statements as part of the survey we administered to participants. Based on the participants' responses we calculated a SUS score of 75.70 out of 100 (standard deviation (SD) of 13.61, CI ±5.33) for Pwm.

As part of an empirical evaluation of SUS, Bangor et al. [2] reviewed SUS evaluations of 206 different systems and compared these against objective measurements of the various systems' success to derive adjective-based ratings for SUS scores (Appendix C, Figure 1). When compared against Bangor's findings on 273 SUS studies, our score of 75.70 falls in the third quartile (70.5–77.8) and above the mean score of 69.5. Pwm's score qualifies for an adjective rating of "Excellent" and is considered "acceptable" in Bangor's acceptability range.

3.1.5 Lessons Learned

Overall, this study was a success. Pwm succeeded in helping first-time users set up and use secure email. When asked in the survey what they liked about Pwm, 23 out of 25 (92%) stated that it was simple and easy to use. No participant indicated that they felt Pwm was difficult to use. Most participants stated that they would use Pwm if they needed to send secure email. Five of the participants (20%) even asked if Pwm was available for download because they wanted to begin using it immediately.

Participants were able to clearly tell the difference between Pwm's secure interface and the underlying interface. Some liked the distinct black background of the security overlay while others wished it looked more like Gmail's native interface. When asked, all participants indicated that it was easy to determine when email had been encrypted using Pwm. The three participants (12%) that initially failed the second task indicated that in the future they would not make the same mistake as they would ensure they could see the distinctive look of the security overlay before sending a sensitive message. While it is hard to know if this is correct, it is still encouraging that users were able to recognize the importance of the visual distinctiveness.

Bookmarklets also proved to be highly usable. Only five (20%) of the participants had used a bookmarklet before; nevertheless, all participants were able to set up and use Pwm. Many participants noted that they liked the fact that the Pwm bookmarklet did not require installation. The one complaint was that the instructions for how to install the bookmarklet should have been more prominent. No participant demonstrated pre-existing knowledge of how to enable the bookmarks bar and the instructions were crucial in helping them set up Pwm. The participants who read the instructions before attempting to add the bookmarklet set up Pwm far faster (average of 30 seconds) than those that tried to add the bookmarklet without first reading the instructions (average of 1.5 minutes).

We asked about half of the participants, including the three who struggled with the second task, how they would react to having email encrypted by default. We explained that this would ensure that they would not accidentally send email without encryption. The participants disliked this idea. In their minds, they saw encryption as something that they would only turn on for sensitive messages and thought it would be annoying to need to frequently turn off encryption. They recognized that decrypting messages adds work for the recipient, and wished to avoid this unless the message was important. Some participants rejected automatic encryption because they believed that only Gmail users could install Pwm and read messages they had sent. Also, it is possible that the short-term nature of the study unfairly biased participants against the ease of decrypting messages as a disproportionate amount of their time (in comparison to real use scenarios) was spent installing Pwm.

We were interested to discover that approximately one third of participants were interested in how their email was being encrypted. Although these participants lacked the technical background to fully understand the cryptography being used, they would still like to see these details published on Pwm's website. They indicated that this would make them feel more confident using Pwm. Even though Pwm users do not want to be concerned with cryptographic details (e.g., key management, signing) while operating Pwm, they

still want this information available so that they can feel more confident that Pwm is securing their messages.

3.2 Voltage Comparison Study

In order to establish that Pwm provided usability benefits in comparison to existing depot-based secure email solutions, we conducted a user study comparing Pwm with Voltage SecureMail Cloud[3] (hereafter referred to as Voltage). Like Pwm, Voltage was designed to allow messages to be encrypted and sent to recipients who had not taken any preparatory action. In addition to comparing usability of similar features, comparing Pwm against Voltage also allowed us to compare users' reactions to secure email systems requiring software installation (Pwm) against systems requiring account creation and verification (Voltage).

In this study, Pwm was run using a browser extension. In the first user study, some participants suggested they would prefer to use an extension to a bookmarklet, and we wanted to see if using the extension would make any difference in the user's experience.

3.2.1 Setup

The second study was comprised of 32 students. Like the Pwm studies, participants were aware they were taking part in a usability study, but were unaware of its focus on security. All but one (97%) participant had been using Gmail for over six months, and 27 (84%) reported that they used Gmail on a daily basis. All participants had experience using Google Chrome. Two of the participants (8%) had encountered PGP in the past but had never used it for any significant period of time.

This study was a within-subjects study, where participants were given simple tasks to complete using both Pwm and Voltage. The order in which they used the systems was randomized so that half used Pwm first, and the other half Voltage. After completing the tasks for one system, they were given a survey rating their experiences (Pwm – Appendix A.4, Voltage – same questions as Pwm, but with "Voltage" replacing "Pwm"). Participants would then complete the tasks and associated survey for the other system. Participants were given a survey with questions about their online security behavior and preferences (Appendix A.5). Finally, participants were interviewed to gather more information about their experiences.

3.2.2 Pwm Tasks

The tasks for Pwm remained the same as the first Pwm study. The only difference was the instructions on the Pwm website were replaced with instructions for setting up and running the browser extension instead of the bookmarklet.

3.2.3 Voltage Tasks

To begin, participants opened an email that had been sent to the provided Gmail account using Voltage. This email, which was generated by Voltage, contained an HTML attachment that included a link to the Voltage website where they could read their encrypted email. At the Voltage website, participants were instructed to create a free Voltage account in order to view their message. Participants had been provided with fake credentials that they could use to fill in the account creation form. After submitting this information, participants were directed to return to their email to retrieve an account verification email from Voltage. After verifying their new Voltage account, participants were able to return to the Voltage website and read their encrypted message. This message instructed them to send a secure reply, which in Voltage only requires clicking "reply."

Unlike the Pwm tasks, participants were not required to send a new encrypted email through Voltage. The participants were using free Voltage accounts, which do not allowing sending new email (only replying), and licensing fees made it impractical to give each participant a commercial account. This step is trivial in Voltage, as it is no different than sending an email in any depot system, and so we do not believe this omission affects the usability results.

3.2.4 Results

Pwm Results.

As with the first study, all participants successfully set up Pwm, but this time they did so without any mistakes or delays. This is likely due to the ease of installing browser extensions in Chrome (only requires two mouse clicks) as well as greater user familiarity with browser extensions (in the first study 5 participants (20%) has used bookmarklets before, where as in the second study 28 (87.5%) had used extensions previously).

Users did experience confusion about being required to refresh the Gmail page before the extension became active (a limitation of Chrome extensions). Several users needed to return to the Pwm website and re-read the instructions before they refreshed the Gmail webpage.

All of the participants (100%) successfully decrypted the email they received. They also all successfully sent an encrypted reply. In the third task, three participants (9%, CI ±11.22) sent their message without encryption. Two of the three recognized their mistake immediately after clicking "Send;" one recognized the mistake when he saw the lock icon on the compose form just after he clicked "Send" and the other when he realized he had not seen the dark background of a security overlay.

Voltage Results.

As with Pwm, all participants (100%) successfully read their encrypted message and replied to it. However, 14 of the 32 participants (44%, CI ±17.32) complained that the process for reading the initial email was extremely cumbersome. Two participants (6%, CI ±8.23) expressly stated that they did not want to leave Gmail to access encrypted email. Overall, only six participants (19%, CI ±13.59) indicated that they preferred Voltage. Of these participants, four preferred the look and feel of Voltage's website, one disliked installing any browser extensions, and one liked that there was a separate site for handling secure messages.

3.2.5 System Usability Scale

Pwm's extension implementation had a calculated SUS score of 70.70 (SD 12.28, CI ±4.26). Voltage had a calculated SUS score of 62.66 (SD 17.53, CI ±6.07). This is a statistically significant difference (paired two tailed t-test, $p = 0.0073$). This result matches opinions expressed by participants during the interview at the end of the study. According to Bangor's adjective ratings, both systems qualify for an adjective rating of "Good." Pwm was in the third quartile and above the mean of 69.5 while Voltage was in the

[3] `http://www.voltage.com/products/vsn.htm`

second quartile and below the mean. According to Bangor's acceptability ranges, Pwm qualifies as "acceptable" while Voltage ranks as "low marginal."

The SUS score for extension-based Pwm is lower than bookmarklet-based Pwm (75.70) but this difference is not statistically significant (unequal variance two tailed t-test, $p = 0.1576$). We believe that this difference in perceived usability was due to a bug that caused a delay between when the page loaded and Pwm became fully functional. This delay caused visible confusion in 7 of the 32 participants (22%), four of whom later commented that the delay bothered them. This annoyance could have contributed to the lower scores Pwm received in this study.

3.2.6 *Lessons Learned*

In addition to the SUS results, the user surveys also showed that users largely preferred Pwm to Voltage. When asked why, they stated that it was because Pwm was integrated directly with Gmail. This supports our original hypothesis that users are resistant to systems that require changes to existing behavior, and that tight integration is able to overcome this concern. Also, like the first study, many participants expressed a desire to use Pwm if they needed to encrypt their personal email.

Several participants suggested ways to improve Pwm. First, while they liked the simplicity of the browser extension, they were also interested in having the option of using a bookmarklet to run Pwm, preferring the flexibility that having both options would provide. Second, participants suggested Pwm's website should look more professional and provide additional details on how Pwm functions. While this is not directly useful to most participants, they indicated that their confidence would be bolstered by the knowledge that Pwm's claims are open to scrutiny by security experts. These suggestions closely parallel suggestions from the first user study. Also, of the six users who preferred Voltage, only one preferred it for reasons inherent to depot-based secure email, while the other five would have preferred Pwm if these two issues were addressed.

4. MP – MANUAL ENCRYPTION

The results from our Pwm user studies along with results from a user study of Private Facebook Chat[13] (a companion system built using security overlays for Facebook Chat) were very positive, especially when compared to some earlier usability studies for secure email [19, 15]. However, several aspects of Pwm and PFC were concerning. First, each study had a small number of users (approximately 10%) that forgot to enable encryption before they sent secure messages. Second, follow-up interviews with participants revealed that some of them did not understand how Pwm works, believing that anyone with Pwm installed could decrypt their email if they stole it.

Initially we thought these issues would be simple to resolve. We considered modifying Pwm to better train users, including displaying video explaining how to encrypt message the first time they ran Pwm, but ultimately felt that this would make users think Pwm was either spam or a virus. We also considered turning on encryption for all messages, but decided against doing so because it places an undue burden on each recipient and potentially interferes with the revenue model of webmail providers; when asked about this option, participants in the user studies also rejected it as undesirable.

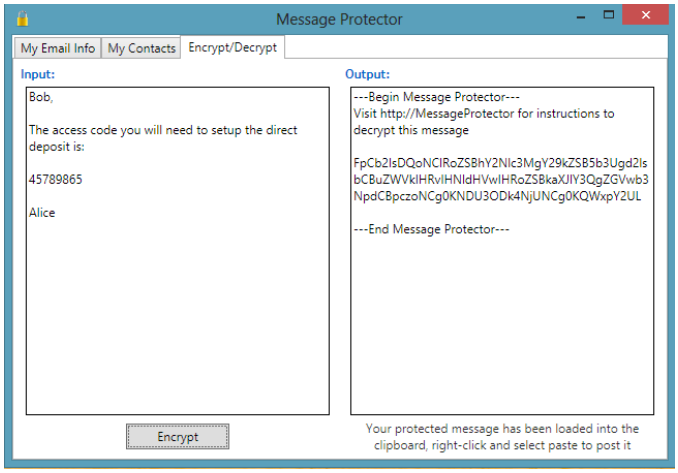

Figure 6: Interface for encrypting a message

Ultimately, we concluded that Pwm's security details were too transparent. This transparency led some participants to author and click "Send" before realizing they had not enabled encryption (a "click-whirr" response [5]). It also caused some participants to not trust Pwm. Since the users did not see the ciphertext, they lacked confidence that encryption and decryption were taking place and didn't understand how the system worked.

We built a mockup of Message Protector (MP), a standalone application that allows users to manually encrypt and decrypt messages. We believed that a separate application with manual encryption could help users better understand what was happening and help them avoid mistakes.

MP is a very simple system. Following installation, MP prompts users to sign in with their email credentials. Users then select email contacts they would like to communicate with securely. These two steps would permit automatic key management and allow users to clearly determine who can read their encrypted messages.

Users are then able to encrypt and decrypt messages (Figure 6). To encrypt a message, they input their desired message into the "Input" text field and click "Encrypt". Their message is then encrypted (base-64 encoded in the mockup) and the ciphertext is placed in the "Output" field. Users then copy the ciphertext to whatever application they wish to use to store or transmit it.

To decrypt a message, users input the MP ciphertext into the "Input" field and then click the "Decrypt" button ("Encrypt" changes to "Decrypt" when ciphertext is detected in the "Input" field). The plaintext is placed in the "Output" field.

5. MP USER STUDIES

We conducted two IRB-approved user studies to evaluate MP. The goal of these studies was to compare MP to existing systems and determine if users would trust the system and find it usable, and if they would make fewer mistakes.

As with the Pwm studies, the participants for these two studies were recruited at Brigham Young University using posters that invited students to participate in either a 45

minute Gmail and Facebook study or a 30 minute Gmail study for the first and second studies respectively. The compensation for both studies was $10.

All participants used the same computer and browser as in the previous user studies. Once again we provided participants with Gmail and Facebook accounts to use during the studies. Before beginning, participants completed a demographic questionnaire (Appendix B.1). Users were not required to identify themselves and we did not record the identity of any participant.

5.1 Encipher.it Comparison Study

In the first MP study we wanted to test MP against an existing secure data sharing tool with manual encryption. We selected Encipher.it[4] because it is a relatively well-known tool, which unlike most other similar tools is currently functional. Encipher.it is a generic bookmarklet-based secure data sharing tool that can encrypt text in any HTML textbox. To use it the user types a message in a textbox and clicks the Encipher.it bookmarklet. Encipher.it then prompts the user to supply a password which is used to encrypt their message. This password must be transmitted out-of-band to the recipient. Following encryption, Encipher.it displays ciphertext in place of the original plaintext message. When a recipient receives an encrypted message and clicks the Encipher.it bookmarklet, Encipher.it prompts the user for the sender's password. After the recipient supplies the password, the message is decrypted and displayed in place of the ciphertext on the webpage.

5.1.1 Setup

This study was comprised of 28 participants. Participants were told that this was a usability study but were not made aware of its security focus. Of the 28 participants, 25 (89%) used webmail daily and 27 (96%) used Facebook weekly. Sensitive information had previously been sent over webmail or Facebook by 24 (86%) of the participants, while only three (11%) of them had encrypted those messages. All (100%) participants reported that protecting the contents of sensitive information was important.

At the beginning of the study, participants were presented with a document that described the study (Appendix B.2). The study was a within-subjects study, where participants were given simple tasks to complete using both Encipher.it and MP (Appendix B.3). The order in which the systems were used was randomly chosen; 16 (57%) participants used MP first and 12 (43%) used Encipher.it first. After completing the tasks for one system, participants were then given a survey to rate their experiences (Appendix B.4). Participants would then complete the tasks and associated survey for the other system. Participants were finally given a post-study survey asking them to state which system they preferred and why (Appendix B.5).

5.1.2 Tasks

Users were given step-by-step instructions on how to complete three tasks using both systems. Task 1 instructed users to install the given system.

Task 2 instructed participants to open Gmail and send an encrypted message containing the text "The last four digits of my SSN is 6789" to the study coordinator. Participants then received an encrypted response to this message and were instructed to decrypt it. To continue they had to input the decrypted message.

Task 3 instructed participants to open Facebook and send an encrypted message containing the text "My bank account password is cougars" to the account's friend named "Alice Jones." Participants then received an encrypted response to this message and were instructed to decrypt it.

[4] `https://encipher.it/`

5.1.3 Results

MP Results.

Of the participants, 25 (89%, CI ±11.45) correctly completed the Gmail tasks and 27 (96%, CI ±7.17) completed the Facebook tasks. The mistakes were split between not understanding how to use the tool and not understanding which portion of the ciphertext to submit to correctly complete the task.

Participants largely succeeded in understanding how MP worked. Twenty-five participants (89%, CI ±11.45) correctly identified who could read encrypted messages. Additionally, 25 participants (89%, CI ±11.45) were able to correctly identify how to decrypt a message using MP.

Encipher.it Results.

Many participants were not able to get Encipher.it to allow them to encrypt or decrypt messages. Only 16 (57%, CI ±18.34) participants were able to decrypt a message in Gmail and only 14 (50%, CI ±18.52) were able to send an encrypted email. Similar to MP, participants fared a little better using Encipher.it with Facebook, as 17 (61%, CI ±18.07) participants successfully decrypted a message and 23 (82%, CI ±14.23) participants successfully encrypted a message. Four participants (14%, CI ±12.85) failed the encryption tasks because they never communicated to the test coordinator the password they had used to encrypt the message.

Participants largely understood how Encipher.it worked, but not as clearly as they understood MP. Twenty-three (82%, CI ±14.23) correctly identified who could read encrypted messages, but only 20 (71%, CI ±16.81) understood how to decrypt a message.

5.1.4 System Usability Scale

MP had a calculated SUS score of 72.23 (SD 13.02, CI ±4.96). Encipher.it had a calculated SUS score of 61.25 (SD 20.11, CI ±7.65). This is a statistically significant difference (paired two tailed t-test, $p = 0.0176$). According to Bangor's adjective ratings, both systems qualify for an adjective rating of "Good." MP was in the third quartile and above the mean of 69.25, while Encipher.it was in the second quartile below the mean. According to Bangor's acceptability ranges, MP qualifies as "acceptable" while Encipher.it ranks as "low marginal"

5.1.5 Lessons Learned

MP was much better at helping the participants avoid making mistakes (paired two tailed t-test, Gmail Decryption – $p = 0.0171$, Gmail Encryption – $p = 0.0052$, Facebook decryption – $p = 0.0022$, Facebook encryption – $p = 0.1033$ [Not significant]). This is likely due to the higher usability marks received by MP, as users found it far easier to use.

MP also performed better at helping participants understand who could read encrypted messages (paired two tailed t-test, $p = 0.0114$) and also how to successfully decrypt mes-

sages (paired two tailed t-test, p = 0.1610), though the second result is not statistically significant.

Perhaps the greatest surprise was that MP's SUS score was as high as Pwm in our previous studies. We had not anticipated this outcome, as we felt that the extra effort of manual encryption would cause participants to reject the system. It is clear from participant responses that they felt more confident using MP precisely because it helped them understand what they were doing. This is reflected by the majority of participants who indicated that the usability of the system was important to them in deciding whether they would use it in their personal lives (MP – 24 [86%, CI ±12.85], Encipher.it – 22 [79%, CI ±15.09]), and more people found MP easy to understand (MP – 23 [82%, CI ±14.23], Encipher.it – 17 [54%, CI ±18.46]).

At the conclusion of the study, we asked participants which system they preferred and why (Appendix B.6, Table 1). First, most participants preferred integrating encryption with the browser and several participants preferred Encipher.it primarily for this reason. Participants that preferred MP also indicated that they wish it had been more integrated. Still, there was a small number of users who felt that MP being a separate application increased security. Second, we observe that users recognize the problem of distributing keys, and several disliked that this was a necessary step of Encipher.it.

5.2 Pwm Comparison Study

The results of the initial MP study were very positive and so we decided to compare it against Pwm. MP was a mockup to study manual encryption and lacks any functionality to help first time recipients of an MP message know how to proceed. For this reason, we selected to replicate the previous MP study (with Pwm replacing Encipher.it) instead of modeling this study after the original Pwm studies. The goal of the study was to see how well MP fosters user understanding when compared to Pwm, and also to explore users' attitudes when comparing the two systems.

5.2.1 Setup

This study was comprised of 28 participants. Participants were told that this was a usability study but were not made aware of its security focus. Of the 29 participants, 28 (97%) used webmail daily and Facebook weekly. Sensitive information had been sent over webmail or Facebook by 27 (93%) of the participants, while only one (3%) had encrypted those messages. Once again, all (100%) participants reported that protecting the contents of sensitive information was important.

The setup and tasks for this system were similar to the Encipher study, but did not include the Facebook tasks since Pwm does not support Facebook. The order in which the systems were used was randomly chosen; 15 (52%) participants used MP first and 14 (48%) used Pwm first.

5.2.2 Results

MP Results.

Of the participants, 27 (93%, CI ±9.29) correctly decrypted a message and 28 (97%, ±6.21) successfully encrypted a message. Comprehension was also high, as 27 (93%, CI ±9.29) correctly identified who would be able to read encrypted messages and all 29 (100%) participants correctly identified how to decrypt a message using MP.

Pwm Results.

Twenty-five (86%, CI ±12.63) participants were able to decrypt a message and 24 (83%, CI ±13.67) were able to send an encrypted email. This is lower than previous results for both Pwm and PFC, and this is possibly due to the study not as closely mimic real world conditions. Pwm is designed to help first time recipients of unsolicited encrypted messages, and so does not provide step-by-step documentation like Encipher.it or MP.

A number of participants fared poorly in understanding how Pwm was functioning. Only 22 (76%, CI ±15.54) of the participants correctly identified who could read a Pwm message, and only 21 (72%, CI ±16.34) knew the proper steps to decrypt a message. Perhaps even more interesting is that 6 (21%, CI ±14.82) participants stated they were unsure of who could read messages and 4 (14%, CI ±12.63) were unsure how to read an encrypted message, whereas no users (0%) reported being unsure of how to use MP in either category.

5.2.3 System Usability Scale

MP had a calculated SUS score of 73.96 (SD 14.23, CI ±5.42). Pwm had a calculated SUS score of 75.69 (SD 16.31, CI ±6.21). This was not a statistically significant difference (paired two tailed t-test, p = 0.61633).

In comparison to Bangor's findings both systems qualify for an adjective rating of "Excellent." Both were in the third quartile and above the mean of 69.25 and both qualifies as "acceptable" on Bangor's acceptability scale.

Extension-based Pwm scored higher in this study then in the second Pwm user study, but this difference was not statistically significant (unequal variance two tailed t-test, p = 0.1867). Extension-based Pwm in this study scored nearly identically to the bookmarklet-based version of Pwm from the first Pwm user study (unequal variance two-tailed t-test, p = 0.9980). In aggregate across all studies Pwm had a SUS score of 73.84 (SD 14.17, CI ±3.04) and MP had a SUS score of 73.14 (SD 13.56, CI ±3.60). This was not a statistically significant difference (unequal variance two tailed t-test, p = 0.7596).

5.2.4 Lessons Learned

MP's manual encryption and clear separation led to nearly all participants correctly understanding who could read messages (paired two tailed p-test, p = 0.0225) as well as how to decrypt a message (paired two tailed p-test, p = 0.0028). Since Pwm keeps more security details transparent to its users, they did not understand how Pwm works and were aware of their lack of understanding.

We were once again surprised that MP performed on par with Pwm in terms of usability. Contrary to our initial thinking, users are not opposed to manual encryption. Users preferred manual encryption because they felt it helped them understand, and thereby trust, the system. Even though MP is a mockup and Pwm is a working system, participants felt that MP was more secure based on its manual encryption.

At the end of the study, we again asked participants which system they preferred and why (Appendix B.6, Table 2). Their answers are helpful in understanding how these results should guide our research. First, users preferred the integration provided by Pwm. Even users who preferred MP were likely to state that they felt Pwm was more usable, but

choose MP because they didn't feel they could trust Pwm. Second, participants felt that manual encryption was necessary to their understanding. Without seeing the ciphertext, they did not feel that Pwm was actually encrypting messages and so were unwilling to use it, and accordingly did not feel that Pwm's other usability benefits were enough to overcome this concern.

6. LIMITATIONS

There are three key limitations in our user studies:

1. The first MP study was an exploratory study designed to measure the potential benefits of manual encryption. MP was not designed to spread in a grass root fashion like Pwm, so the first MP user study assumed the user had already installed the necessary software and shared secret keys before they received an encrypted message. Once the results from the first study indicated the potential benefits of manual encryption, we decided to compare MP against Pwm. We modeled this comparison study after the first MP study for consistency. Thus, the second MP study assumed the user had already installed Pwm. The user tasks focused on encryption and decryption and not software installation. While this proved sufficient for comparing comprehension, the results of this study are not fully representative of all aspects of Pwm, and potentially biased participants by not allowing them to experience one of Pwm's key usability features.

2. Our user studies were short-term laboratory studies. Short-term studies have several inherent limitations: first, it is hard to accurately address trust in a laboratory setting [16], and second, it does not allow us to analyze whether participants would correctly use the prototype over an extended period of time. Perhaps MP's lack of integration would become a considerable issue after repeated use. Similarly, the perception of Pwm's usability could change over time. In the future, we plan to conduct long-term studies to address these issues.

3. The first SUS question reads "I think that I would like to use this system frequently." In the post-study interviews, we determined that users were giving Pwm lower scores for this question because they did not feel that they would send secure email very often, even though they were enthusiastic about using Pwm whenever they would send secure email. Thus, SUS scores may be negatively impacted by an important yet infrequent activity, even if the tool for performing that activity is highly usable.

7. RELATED WORK

Whitten and Tygar conducted a usability study of PGP 5.0 in their seminal paper on usable security [19]. It served as a wakeup call to the security community because a large percentage of users failed to complete basic tasks installing and using a state-of-the-art secure email tool. In their study, 3 of the 12 users (25%) mistakenly sent the secret message unencrypted. In our work, we demonstrated a secure webmail tool with very high success rates sending encrypted email, but we also observed a small percentage of users mistakenly sending out plaintext.

Sheng et al. [15] repeated the Whitten and Tygar user study with PGP 9.0. They noted some improvements due to automatic encryption, but they identified a number of problematic issues surrounding key management and digital signatures. One of their major findings was that encryption was so transparent that users were unsure whether it had occurred or not. The paper recommends that users be given the option to designate in advance whether an email is to be encrypted or not. We designed Pwm to follow this suggestion, but our own studies indicate that users can still mistakenly send out a sensitive message without encryption.

Garfinkel and Miller [9] created a secure email system that combined the idea of Key Continuity Management (KCM) with S/MIME. They introduced a tool to Outlook Express that would alert users through visual indicators if a sender that had previously sent them secure email was now sending an email that was not signed or was signed by a different key. They repeated the original Johnny experiment with some additional tasks to test how users reacted to attacks against the KCM system. Their work demonstrated that automatic key management provides significant usability compared to earlier studies that burdened users with key management tasks. They observed that their tool "was a little too transparent" in how well it integrated with Outlook Express, and sometimes users failed to read the instructions accompanying the visual indicators. Our work also illustrates the benefits of automatic key management, but we use a very different key management paradigm based on identity-based cryptography since we focus on making it easy for users to obtain our software after they have received an encrypted message and to start encrypting their webmail. We also observed some issues related to too much transparency. Our work is complimentary, and we could incorporate KCM to address the kinds of attacks they describe in their paper.

Clark et al. [6] analyze the P25 short-range wireless two-way communications protocol used for emergency and law enforcement personnel. They discovered that a small amount of sensitive traffic is inadvertently sent unencrypted due to individuals and groups being confused about when encryption is actually turned on. One contributing factor is the user interface design that enables encryption by rotating a knob to a specific position. They observed that users occasionally assume encryption is on and mistakenly communicate in the clear. We experienced a similar phenomenon with our software interface that lets users turn encryption on and off.

Fahl et al. [7] conducted usability studies for various design options for Facebook private messaging. They determined a strong user preference for automatic key management. They also selected automatic encryption, but there wasn't a significant preference for it compared to manual encryption. They suspected that making encryption details too transparent could fail to generate a feeling of trust among the users of a system, and they recommended that this issue be explored in more detail in the future. Our work provides evidence to confirm this suspicion. We had users report that they had more trust in a system that exposes more security details.

Sun et al. [17] examined the usability of OpenID, a promising Web single sign-on system. They identified concerns and misconceptions among users that inhibit the adoption

of OpenID. They illustrate how the OpenID login flow promotes an inaccurate mental model to users. They describe an alternative to the OpenID login flow that assists users in forming a more accurate mental model and believe that this will help users be more likely to adopt OpenID. In our research, we observed that some users were wary about adopting our email system even though they found it easy to use because the security details were too transparent.

8. CONCLUSIONS AND FUTURE WORK

The contributions of this paper are:

- An overview of the design of Pwm, which layers encryption over existing webmail solutions using a novel approach of security overlays that are functionally transparent but visually distinctive. Pwm is specifically designed to spread in a grass roots fashion so that a user can send any other user an encrypted message before the recipient generates any cryptographic keys or installs any software.

- The results of a series of usability studies of Pwm that compare it to several existing secure email tools. The systems are compared using a standard usability metric, System Usability Score (SUS). Pwm is shown to be highly usable and compares favorably to the other tools used in the studies.

- Even though most Pwm users in the study encrypted and decrypted messages correctly, a few users mistakenly sent out secure messages in the clear. Users were unsure whether to trust the system because security details are too transparent. We compared Pwm to MP, a mockup that uses manual encryption. We were surprised that users rated the usability of MP on par with Pwm. They had more trust in MP and avoided mistakes. Our results suggest that designers may want to reconsider manual encryption as a way to reduce transparency and foster greater trust.

Usable secure webmail has been a long unsolved problem. Pwm (Private WebMail) is a system that adds end-to-end encryption and message integrity to existing webmail systems. Pwm addresses the problem of usability in secure email in three ways: First, Pwm integrates tightly with existing webmail interfaces, providing functionally transparent interfaces, to relieve the burden users feel when learning new systems. Second, Pwm features fully automatic key management that requires no interaction from users. Third, Pwm provides transparent and automatic encryption, whereby users can trivially encrypt and sign their messages. Overall, Pwm was designed to maximize usability while still providing good enough security and is advantageous to those already sending sensitive information over webmail.

To verify the usability of Pwm we conducted two IRB-approved user studies with 25 and 32 participants respectively. In a laboratory setting, the participants were sent an encrypted email that also contained plaintext instructions on how to set up and use Pwm in order to read the message. Every participant, except one who misunderstood the instructions, was able to decrypt and read the message. The second study included a comparison with Voltage, an existing depot-based email encryption system.

Even though participants gave Pwm high usability marks, a small but consistent percentage of participants (approximately 10%) forgot to enable encryption. Some users did not trust that the system was secure because the security details (key management and encryption) were so transparent that they did not have a clear idea about how the system actually worked. We speculated that a combination of manual encryption and a clear separation of duties would help users trust the system and avoid sending information without encryption. To test this hypothesis we built Message Protector (MP), a mockup that included manual encryption in an application separate from the browser.

Using MP, we conducted two more IRB-approved user studies with 28 and 29 participants respectively. In the first study we compared MP against an existing secure data sharing tool, Encipher.it. MP proved to be significantly more usable than Encipher.it and also helped users better understand what security was being provided. We then tested MP against Pwm and found evidence that manual encryption can foster greater trust and reduce user errors. The user studies also revealed that participants preferred that security systems be tightly integrated with the browser.

Thus, in the effort to balance security and usability, we argue that a combination of exposing some encryption details and tight integration will produce a system that users trust and help them to secure their data without making mistakes.

8.1 Future Work

We were surprised that users were accepting of the extra effort that manual encryption requires in MP compared to the transparent encryption in Pwm. Even more significant was the feeling of trust fostered by manual encryption in MP. We are not yet convinced that MP should replace Pwm because there was a strong preference for tight integration with the website from a number of users. Also, we believe MP would be unacceptable to users when sensitive information is exchanged so rapidly (e.g., secure chat) that it would require repeated switching between applications. Instead, we intend to combine the advantages of Pwm and MP into a hybrid system that leverages the strengths of MP in order to overcome weaknesses in Pwm.

Our experience demonstrates that automatic encryption hides so many details that users are confused about what precisely is occurring and can sometimes lead users to mistakenly disclose plaintext when the encryption option is too similar to no encryption. We plan to take these lessons learned and apply them to our next-generation secure Webmail tool. We plan to support manual encryption and explore varying degrees of separation in order to eliminate confusion and mistakes. For example, we plan to add manual encryption to Pwm's existing security overlays. Instead of treating encryption and transmission of email as one step, we will have the compose security overlay produce cipher text, and then require the users to separately click "Send" on the webmail interface. Additionally, we will require that users manually choose to decrypt Pwm email messages by clicking a button, instead of having this occur automatically.

Another potential way to incorporate manual encryption, but with more separation, would be to move encryption and decryption into a sidebar. This sidebar would be hosted in a security overlay and would be independent of the underlying website. This independence would allow it to support

all webmail providers, as well as any other websites that allow users to share text. It could also work in conjunction with the more integrated portions of Pwm by becoming the fallback mechanism used when the more integrated Pwm is unable to parse a website.

We will experiment with these and similar ideas to strike a balance between manual encryption/separation and usability/integration. Hopefully this would raise the SUS score of Pwm to increase the probability that users will recommend it to their friends.[5] We will conduct further laboratory studies to verify how well our improved system is trusted and helps users avoid mistakes. Finally, we will then conduct a long-term usability study to determine if our results carry over into the real world where users would use Pwm to protect their sensitive data.

9. ACKNOWLEDGMENTS

The authors would like to thank Trevor Florence, Kimball Germane, Scott Robertson, Chris Robison, and Ryan Segeberg for their work in the Internet Security Research Lab (ISRL) at BYU to develop ideas and software related to this work. They also thank the anonymous reviewers and shepherd for their valuable feedback.

10. REFERENCES

[1] H. Abelson, R. Anderson, S.M. Bellovin, J. Benaloh, M. Blaze, W. Diffie, J. Gilmore, P.G. Neumann, R.L. Rivest, J.I. Schiller, et al. The risks of key recovery, key escrow, and trusted third-party encryption. *World Wide Web Journal*, 2(3):241–257, 1997.

[2] A. Bangor, P. Kortum, and J. Miller. Determining what individual sus scores mean: Adding an adjective rating scale. *Journal of Usability Studies*, 4(3):114–123, 2009.

[3] J. Brooke. Sus-a quick and dirty usability scale. *Usability evaluation in industry*, 189:194, 1996.

[4] O. Chevassut, P.A. Fouque, P. Gaudry, and D. Pointcheval. Key derivation and randomness extraction. In *In Crypto'05*. Citeseer, 2005.

[5] Robert B Cialdini. *Influence: Science and practice*, volume 4. Allyn and Bacon Boston, MA, 2001.

[6] Sandy Clark, Travis Goodspeed, Perry Metzger, Zachary Wasserman, Kevin Xu, and Matt Blaze. Why (special agent) johnny (still) can't encrypt: a security analysis of the apco project 25 two-way radio system. In *Proceedings of the 20th USENIX Conference on Security*. USENIX Association, 2011.

[7] Sascha Fahl, Marian Harbach, Thomas Muders, Matthew Smith, and Uwe Sander. Helping johnny 2.0 to encrypt his facebook conversations. In *Proceedings of the Eighth Symposium on Usable Privacy and Security*, SOUPS '12, pages 11:1–11:17, Washington, D.C., 2012. ACM.

[8] Simson L. Garfinkel. Email-based identification and authentication: an alternative to PKI? *IEEE Security & Privacy*, pages 20–26, 2003.

[9] Simson L. Garfinkel and Robert C. Miller. Johnny 2: a user test of key continuity management with s/mime and outlook express. In *Proceedings of the 2005 symposium on Usable privacy and security*, SOUPS '05, pages 13–24, Pittsburgh, Pennsylvania, 2005. ACM.

[10] S. Gaw, E.W. Felten, and P. Fernandez-Kelly. Secrecy, flagging, and paranoia: adoption criteria in encrypted email. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 591–600. ACM, 2006.

[11] C. Herley. So long, and no thanks for the externalities: the rational rejection of security advice by users. In *Proceedings of the 2009 Workshop on New Security Paradigms Workshop*, pages 133–144. ACM, 2009.

[12] H. Krawczyk. Cryptographic extraction and key derivation: The hkdf scheme. *Advances in Cryptology–CRYPTO 2010*, pages 631–648, 2010.

[13] Chris Robison, Scott Ruoti, Timothy W van der Horst, and Kent E Seamons. Private facebook chat. In *2012 ASE/IEEE International Conference on Social Computing (SocialCom) and 2012 ASE/IEEE International Conference on Privacy, Security, Risk and Trust (PASSAT)*, pages 451–460. IEEE, 2012.

[14] Adi Shamir. Identity-based cryptosystems and signature schemes. In George Blakley and David Chaum, editors, *Advances in Cryptology*, volume 196 of *Lecture Notes in Computer Science*, pages 47–53. Springer Berlin / Heidelberg, 1985. 10.1007/3-540-39568-7_5.

[15] S. Sheng, L. Broderick, CA Koranda, and JJ Hyland. Why johnny still can't encrypt: evaluating the usability of email encryption software. In *2006 Symposium On Usable Privacy and Security - Poster Session*, 2006.

[16] Andreas Sotirakopoulos, Kirstie Hawkey, and Konstantin Beznosov. "i did it because i trusted you": Challenges with the study environment biasing participant behaviours. In *SOUPS Usable Security Experiment Reports (USER) Workshop*, 2010.

[17] San-Tsai Sun, Eric Pospisil, Ildar Muslukhov, Nuray Dindar, Kirstie Hawkey, and Konstantin Beznosov. What makes users refuse web single sign-on?: an empirical investigation of openid. In *Proceedings of the Seventh Symposium on Usable Privacy and Security*, SOUPS '11, pages 4:1–4:20, Pittsburgh, Pennsylvania, 2011. ACM.

[18] Timothy van der Horst and Kent Seamons. Simple authentication for the web. In *Security and Privacy in Communication Networks*, September 2007.

[19] Alma Whitten and J. D. Tygar. Why Johnny can't encrypt: A usability evaluation of PGP 5.0. In *8th USENIX Security Symposium*, 1999.

[5] `http://www.measuringusability.com/usability-loyalty.php`

APPENDIX

A. PWM USER STUDIES

A.1 Demographic Questionnaire

Are you a student?
What is your major?
What is your occupation?
What is your gender?
What is your approximate age?
How long have you been a Gmail user?
Approximately how often do you use Gmail?

A.2 Bookmarklet Tasks

Introduction

Thank you for your participation. During this study, you will be asked to perform certain tasks using Gmail and then provide feedback to help us improve our software. During the course of this study, all acts taking place on the screen will be recorded along with audio of anything we discuss. This will help us learn whether or not our software is easy to use. None of the video or audio content captured during the study will be released publicly or given to a third party. Before beginning the study, we will also ask you to provide some demographic information. None of the results published as part of this research will personally identify you as a participant.

You will have access to a temporary Gmail account for use in completing tasks during this study. You will not be asked to use your own Gmail login name or password at any time. Do not enter or access any of your own personal data during the study since everything on the screen will be recorded.

You will receive $10.00 as compensation for your participation in this study. The expected time commitment is 20-30 minutes. If you feel uncomfortable with any aspect of this study you may quit at any time.

Please advance to the next screen when ready.

Task 1

Please login to our test Gmail account with the login name and password shown below. Read the first message and follow the instructions given in the message. Close Gmail when you are finished and advance to the next page of instructions.

Click here to open Gmail
Username: pwmstudy@Gmail.com
Password: pwmusability

Task 2

Please log back into our test Gmail account with the user name and password shown below. Send a secure message to Gmailstudy@isrl.cs.byu.edu using Pwm. Include the ID number you were given in your message. Wait for a reply with further instructions.

Click here to open Gmail
Username: pwmstudy@Gmail.com
Password: pwmusability

A.3 Bookmarklet Post-study Survey

SUS Questions:
Please answer the following question about Pwm. Try to give your immediate reaction to each statement without pausing to think for a long time. Mark the middle column if you don't have a response to a particular statement.
Choose from 1 (strongly disagree) to 5 (strongly agree).

1. I think that I would like to use this system frequently
2. I found the system unnecessarily complex
3. I thought the system was easy to use
4. I think that I would need the support of a technical person to be able to use this system
5. I found the various functions in this system were well integrated
6. I thought there was too much inconsistency in this system
7. I found the system very cumbersome to use
8. I would imagine that most people would learn to use this system very quickly

9. I felt very confident using the system
10. I needed to learn a lot of things before I could get going with this system

Remaining Questions:
Please give your response to the following general statements. Try to give your immediate reaction to each statement without pausing to think for a long time. Mark the middle column if you don't have a response to a particular statement.
Choose from 1 (strongly disagree) to 5 (strongly agree).

1. I trust Gmail with my sensitive email messages
2. I am concerned about Gmail scanning my messages
3. I feel safe sending important information through email
4. I worry that some messages aren't really from who they say they are from
5. I found the bookmarklet easy to use (The button you dragged to your toolbar is called a bookmarklet)

Have you used a bookmarklet before this study?
What did you like about Pwm?
What did you dislike about Pwm and how would you like it to be changed?
Have you ever been asked to send sensitive information you were not comfortable sending through email?
What type of sensitive information were you asked to send?
Did you send the requested information?
Have you ever received information you were not comfortable receiving through email?
What type of sensitive information did you receive?
If you started using Pwm on your own, would you prefer protection for new messages to be? *Choose one: Always on; Only on for the message that was open when you clicked "Secure my Email"; Off, unless I click a separate button on the Gmail page*

A.4 Extension Post-study Survey

Please answer the following question about Pwm. Try to give your immediate reaction to each statement without pausing to think for a long time. Mark the middle column if you don't have a response to a particular statement.
Choose from 1 (strongly disagree) to 5 (strongly agree).
Same SUS questions from A.3.

What did you like about Pwm?
What did you dislike about Pwm and how would you like it to be changed?
Other comments on Pwm

A.5 Additional Post-study Questions

Please answer the following general statements. Try to give your immediate reaction to each statement without pausing to think for a long time. Mark the middle column if you don't have a response to a particular statement.
Choose from 1 (strongly disagree) to 5 (strongly agree).

1. I trust Gmail with my sensitive email messages
2. I am concerned about Gmail scanning my messages
3. I worry that some messages aren't really from who they say they are from
4. I feel safe sending important information through email
5. I feel safe creating accounts with usernames and passwords on new sites
6. I feel safe installing browser extensions or plugins
7. Creating accounts for new websites is easy
8. Installing browser extensions is easy
9. I feel safe clicking on links in email messages
10. I feel safe clicking on links in email messages from people I know
11. I never click on links in email messages

Have you installed browser extensions, add-ons or plugins before today?
What has prevented you from installing browser extensions, add-ons or plugins in the past?
When deciding whether you will trust a browser extension, add-on or plugin, what influences your decision?
Have you ever been asked to send sensitive information you were not comfortable sending through email?
What type of sensitive information were you asked to send?

Did you send the requested information?
Have you ever received information you were not comfortable receiving through email?
What type of sensitive information did you receive?

B. MESSAGE PROTECTOR USER STUDIES

B.1 Demographic questions

What is your age?
What is your gender?
What is your major?
How do you rate your level of computer expertise?
How often do you use webmail?
How often do you use Facebook?
Have you ever sent private or sensitive information via Web email or Facebook?
How did you send that information *Select all that apply: Web email; Facebook private message; Facebook wall post; Instant message; Other (please specify below):*
How important is maintaining the privacy of your messages containing sensitive information? *Very important; Important; Neither important nor unimportant; Unimportant; Very unimportant*
Have you ever encrypted an email or Facebook message?

B.2 Study Introduction

Purpose

The purpose of this study is to compare two Internet encryption systems, Message Protector (MP) and Encipher.

What to Expect

In the study, you will attempt a set of tasks that Internet users regularly perform. You will do each set of tasks twice, once with MP and once with Encipher. Following the completion of each set, you will complete a survey about your experience with the application. During this study, all actions taking place on the screen will be recorded along with audio content of anything we discuss, however we will not record video of you. This will help us to analyze our software's usability. None of the video or audio content recorded during the study will be released publicly or given to third parties. Prior to the study beginning you will complete a short survey about yourself. None of the results published as part of this research will personally identify you as a participant.

Introduction

MP and Encipher are programs that allow Internet users to encrypt text that they communicate through websites. In this study you will perform common Internet tasks with MP and Encipher. This study will take about 45 minutes. Try to perform each task as quickly and accurately as you can. If you get stuck at any point, please call the proctor for assistance. You will receive $10.00 as compensation for your participation in this study. If you feel uncomfortable with any aspect of this study, you may quit at any time. Thank you for participating!

B.3 Message Protector Tasks

Message Protector Tasks

Message Protector (MP) is a computer program that allows users to protect Internet messages (e.g., email, Facebook private messages) via encryption. In this portion of the study, you will execute various tasks that comprise the primary functionality of MP and answer a few related questions.

Scenario 1: Installation

In this scenario, you will install MP on a computer. Please follow the instructions as closely as possible.

Scenario 1 Task 1: MP Installation

Access http://MessageProtector and follow the instructions in section 1 "Installing Message Protector."

MP requires an email address and the email account password to allow the user's contacts to be able to read their protected messages. For this study, we have created the following test account for you to use:

Email Address: userstudyMP@Gmail.com

Password: mpUserStud

Allow the following contacts to read your protected messages: randomFriend@hotmail.com, mom@familyWebsite.com, recipientMP@Gmail.com, stalwartStudent@byu.edu

Scenario 2: Gmail

In this scenario, you will encrypt and decrypt email messages with MP. Open Chrome and click the Gmail bookmark on the Favorites bar. A test account will already be logged in.

Scenario 2 Task 1: MP Email Encryption

Access http://MessageProtector and follow the instructions in section 2 "Encrypting Messages" to send an email to recipientMP@Gmail.com. Include the phrase "The last four digits of my SSN is 6789" in the message.

Scenario 2 Task 2: MP Email Decryption
After completing the previous task, you will receive a protected reply email from recipientMP@Gmail.com. Access http://MessageProtector and follow the instructions in section 3 "Decrypt Message" to decrypt the protected message.

Type the decrypted message below:

Scenario 3: Facebook Private Message
In this scenario, you will encrypt and decrypt Facebook private messages with MP. Open Chrome and click the Facebook bookmark on the Favorites bar. A test account will already be logged in.

Scenario 3 Task 1: MP Private Message Encryption
Access http://MessageProtector and follow the instructions in section 2 "Encrypting Messages" to send an encrypted Facebook private message to the user study account's friend named "Alice Jones." Include the phrase "My bank account password is cougars" in the message.

Scenario 3 Task 2: MP Private Message Decryption
After completing the previous task, you will receive a reply private message from "Alice Jones". Access http://MessageProtector and follow the instructions in section 3 "Decrypting Messages" to decrypt the protected message.
Type the decrypted message below:

Finished
This concludes the Message Protector portion of the study. Please answer the questions below about your experience. Please record your immediate response to each question. If you feel that you cannot respond to a particular question, please mark the center point of the scale.

B.4 Message Protector Post-study Survey

Please answer the following question about Voltage. Try to give your immediate reaction to each statement without pausing to think for a long time. Mark the middle column if you don't have a response to a particular statement.
Choose from 1 (strongly disagree) to 5 (strongly agree).
Same SUS questions from A.3.
My level of understanding of MP directly affects whether I would use it to protect my email and Facebook messages.

Who can read messages that you protect with MP? *Choose one: Anyone that has MP installed, receives the message, and that I have selected to communicate with securely; Anyone who receives the message and who I have selected to communicate with securely; Anyone who receives the message; Anyone who has MP installed; I don't know*
After MP is installed, what actions must recipients take to read MP protected messages? *Choose one: Access the MP website; Copy the message and paste to MP; Copy the message, paste to MP, click the Encrypt button; Copy the message, paste to MP, click the Decrypt button; I don't know*
How often would you use MP to protect your email and Facebook messages? *Choose one: Always; Very Often; Occasionally; Rarely; Very Rarely; Never*
What did you like about MP?
How could MP be improved *Select all that apply: Provide better operating instructions; Provide more information about MP to the user; Provide an easier way to select trusted contacts; Provide a more intuitive user interface; Provide a less intrusive or cumbersome experience; Other (please specify below):*

B.5 Additional Post-study Survey Questions

Please answer the following question about Voltage. Try to give your immediate reaction to each statement without pausing to think for a long time. Mark the middle column if you don't have a response to a particular statement.
Choose from 1 (strongly disagree) to 5 (strongly agree).
I feel that it is important to encrypt my emails and Facebook messages that contain sensitive or private information
I would use a different Internet Encryption tool for every website that I store or share sensitive information
I trust Gmail employees to not disclose, misuse, or abuse my email and Facebook messages
I trust Facebook employees to not disclose, misuse, or abuse my email and Facebook messages
I would trust a company other than Facebook or Gmail (i.e., Encipher, MP) to protect my email and Facebook messages

Which system would you prefer to use? *Choose one: Message Protector; Encipher; Both; Neither*
Please explain your answer to the previous question:

B.6 Survey Responses

Table 1: Encipher.it Comparison Study Results

Preferred System	Reason
Message Protector	I had trouble decrypting the messages when using Encipher.
Encipher	I felt I understood how Encipher works more clearly. Also, I liked having the bookmark tab available.
Message Protector	The program was easy to use and did not require any kind of key.
Neither	I felt that anyone with Message Protector would decipher my emails so there is no point in encrypting them. / Encipher was a little clunky. I don't understand how I would set up a passcode and how the receiver would know what it was / Plus I don't send private information that frequently.
Neither	Both were too complicated. I probably wouldn't use either because it takes too much time and i would just take the risk of me getting my stuff stolen
Encipher	Encipher is easier to use even though the person receiving the message has to have the encryption key.
Message Protector	Encipher didn't work either time and was slow. I think it would be easier than message protector though as it is already integrated into the website and i dont have to leave the page I am on. So I like the idea of encipher better but since it didn't work for me, I am biased to MP
Message Protector	It felt safer. Encipher just felt like a pop-up that I should block and I wasn't sure why it was safe or how I would get the security key thing or give it to the people I would want to see my private message.
Message Protector	Message protector is most easy to use i like it!!!
Message Protector	IT is easy to use
Message Protector	I have a hard time remembering passwords, and I don't really understand how I could send a password privately to the recipient of an encrypted message via Encipher it. Message Protector was simpler for me to understand and use.
Message Protector	I liked that I didn't need to specify an encryption key in Message Protector and my secure contacts were already recognized by the computer.
Message Protector	Unlike Encipher, it doesn't require a new key each time you want to encrypt or decrypt a message.
Encipher	I didn't have to load my contacts into it. You also need a decryption key from the sender, so you can't just decrypt it if you have the program, like Message Protector allows you to do.
Encipher	Encipher is already integrated into the internet so it's much more convenient to use. I would use message protector if I wasn't in an HTPPS website.
Message Protector	Just was eaiser to use. No password needed, and I felt that it was more secure.
Encipher	Encipher is a quick and easy tab that requires everytime a new password that can be complex. That's it. you use the tab, you use a password. that simple. MP requires installing something and going from window to window, and someone can decipher your messages with your email and email password, or your contact's email or email password. Most people don't have that complex passwords and so I would be concerned a bit with Personal Information. Encipher uses a whole new level of protection.
Encipher	I think that Encipher is better because it has you pick your secure contact every time. With the MP, you could accidentally send sensitive info to one of your 'safe' contacts, but not the one that you wanted to see that info. Encipher is more user friendly.
Encipher	Even though Encipher requires a decryption key, it doesn't require pasting your message into a separate window.
Message Protector	
Message Protector	Easier and cleaner experience. It didn't take me even half the time to figure out how to use this as it did with Encipher.
Both	They both are nice. I like the toolbar aspect of Encipher, but did not like that the encryption key did not work for me on the first task.
Encipher	I can send protected messages to all my friends who have the key but not only with some limited contacts.

Continued on next page

Table 1 – continued from previous page

Preferred System	Reason
Both	If I have to send some private information over the internet, it's most likely that I would do so with my father who is not very good at using computer. Encipher would be easier for him to start with. If he gets used to it, I would switch to Message Protector. Also, I can use Encipher at any computer with the least effort-just adding to favorites. It's extremely convenient. However with other people who are good at computer, I would use Message Protector right away. And if I have to send a very important message, I would use Message Protector, since I feel like it's a more secured program.
Message Protector	I don't know if it was me, but I couldn't make the Encipher program decrypt the messages. However, I worked with Message Protector much better and I was able to decrypt the message. However, I feel Encipher has a better idea in just simply typing in a password instead of copying and pasting. If Encrypt would've worked for me I would've liked it more because of the simplicity.
Encipher	more protection. I could forget who I selected from my contacts if I use message protector.
Both	Enchipher was quicker and easier, but Message Protector was easier to do without contacting the sender/recipient for a password which I feel is a plus.
Both	With important sensitive information, such as my SSN or Bank account number and password, I would use MP because the fact alone that it requires a download and then a separate window from the conversation makes it feel more secure, although it probably has no actual security difference. With more routine information, such as the fact that I can't stand my untrustworthy boss, I would use Encipher because it is fast and conveniently located in the bookmarks bar, rather than requiring that I open a program I have saved somewhere on my computer. / / On top of this, if I were to have a long conversation, all of which I would like to have encrypted, I would use MP because it requires less time to function and to carry out operations. However for a quick encrypted message, I would use Encipher becasue it is more convenient than opening a separate program.

Table 2: Pwm Comparison Study Results

Preferred System	Reason
Both	I like the encryption for the Message Protector, but it is not as convenient as Pwm since it is right in Gmail.
Message Protector	The reason i would prefer Message Protector is because it seemed more reliable and safe. There seamed to be a more secure connection between you and the recipient due to the fact that you had to add the secure email. I'm not sure how it works and that not just any one can decrypt your message with the same software. I think that PWM is defiantly easier to use but doesn't seem as secure. If i am wanting to encrypt some content I would take a little but longer to make sure that it is safe
Both	They both seemed effective and useful. I would use the system that others are using.
Pwm	Pwm was integrated into Gmail which I use and many of my friends as well.
Pwm	I liked MP, but Pwm being directly in the email makes it more user friendly and less of a hassle.
Message Protector	Message Protector gives me a reason to believe that my message was actually encrypted. Pwm, on the other hand, was very easy to use, but seemed almost too easy. I can still read my message after encrypting, which makes me think that perhaps it wasn't actually encrypted.
Pwm	I would use Pwm simply because it's an extension easily integrated into Gmail. It required little customization, just a simple click of the button. However, I felt a little more confident that I was using MP correctly and that it was encrypting my messages. My choice is mostly out of design and convenience, trusting that both programs do the task effectively.
Pwm	Mainly because I don't have to have to separate windows to encrypt/decrypt stuff. It just seems less of a hassle when it's built into the Gmail system.
Message Protector	Although MP was a little (not by much) more difficult to use, I can be certain of who is able to view the encrypted data.
Pwm	I found it faster and easier for the program to encrypt and decrypt messages for me than copying and pasting it myself.
Pwm	It was simpler, and easier to use.
Pwm	Much more convenient. MP is too much of a hassle, although if it was more secure than Pwm, I might use it, but I would still use it much less than Pwm.

Continued on next page

Table 2 – continued from previous page

Preferred System	Reason
Pwm	Pwm was much easier to use and the instructions were easier to understand.
Both	Depending on who I was corresponding with, I may switch between encrypting programs. It seems useful, but it may require the other party to have some decrypting program installed, which is not very convenient for banks or places that I might be sending messages to.
Both	I would use both because I am not very educated on either of them. I would want to use both to see which would become more comfortable for me. When I understood more about them I could then decide which was more effective for me.
Pwm	I like Pwm's integration right into the browser. The only fault is that you only see the encryption after it has been sent, so if it fails, your information isn't encrypted.
Message Protector	I found message protector to be a little easier to understand
Message Protector	It was easier to use and had much better instructions.
Message Protector	See the reasons stated before for using MP - offline encryption, more intuitive, etc
Message Protector	Even though it is slightly more difficult to use, it seemed more secure. I also liked that you could choose what contacts could communicate with.
Both	Pwm is much easier to use.... but Message Protector seems like it might be safer in only letting certain people see it? DEfinitely prefer pwm if it's just as secure.
Pwm	I would prefer Pwm, because you don't have to copy and paste your message, then press the encrypt or decrypt button, as you need to do with MP. I think that it is easier just to have pwm enabled, which is a lot faster and smoother, and it just protects your message for you, without needing to encrypt the message manually.
Message Protector	It's much easier to use, and makes more sense than Pwm. I like that you can see all the steps of what you're doing, so you feel more in control of the process.
Message Protector	Pwm didn't work for me.
Message Protector	MP had a simpler design and was much more user friendly. (i felt like i was in a catch-22 with Pwm.)
Pwm	PWM requires fewer steps and was less complicated to me.
Message Protector	I feel that Pwm, as a Gmail extension would be easy for anyone to get. If I were to send sensitive information to the wrong address, to my understanding the could simply install the extension and view it. With MP, they not only need a 2nd party program installed, (one not as easily located) and I would need to have them on my selected contact list. It feels safer.
Message Protector	Though it was slightly more complex in the set up I personally found it more easy to use. I like that I could see that it was encrypted.
Pwm	It's easier than Message protector.

C. BANGOR'S SUS SCALE

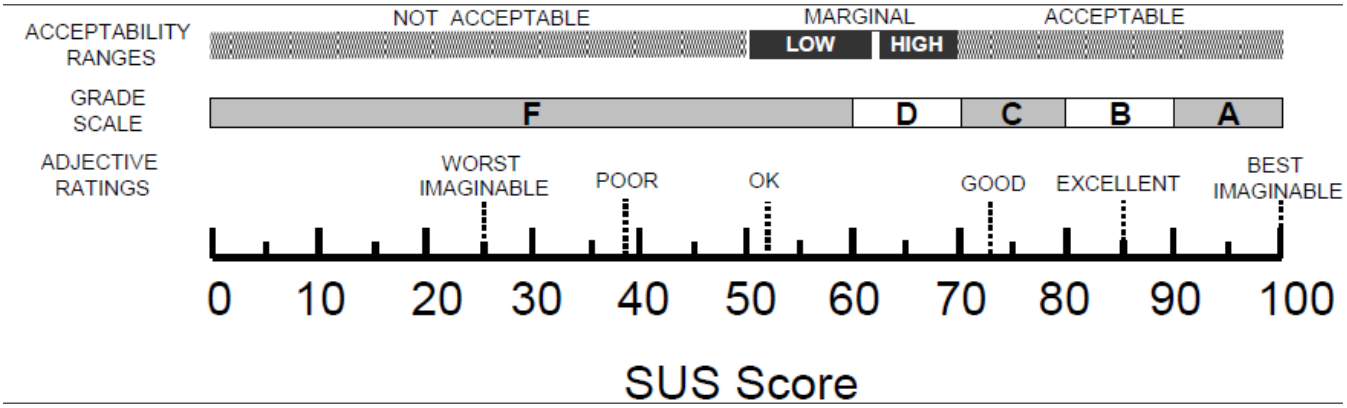

Figure 1: An adjective oriented interpretation of SUS scores [2]