2013-05-14

Message Protector: Demonstrating that Manual Encryption Improves Usability

Nathan I. Kim
Brigham Young University - Provo

Message Protector: Demonstrating that Manual Encryption Improves Usability

Nathan I. Kim

A thesis submitted to the faculty of
Brigham Young University
in partial fulfillment of the requirements for the degree of

Master of Science

Kent E. Seamons, Chair
Quinn O. Snell
Yiu-Kai Dennis Ng

Department of Computer Science

Brigham Young University

May 2013

ABSTRACT

Message Protector: Demonstrating that Manual Encryption
Improves Usability

Nathan I. Kim
Department of Computer Science, BYU
Master of Science

Billions of people currently use the Internet. Many Internet users share sensitive information through online services. Several secure data sharing tools have been developed to protect this sensitive information. A common practice in the design of usable security is to provide automatic data encryption that is transparent to users. We hypothesize that automatic encryption can decrease usability and comprehensibility, increasing the likelihood that users will unknowingly disclose sensitive information. This thesis presents Message Protector (MP), a novel Internet secure data sharing tool design that through manual encryption, purposely exposes technical details in a usable manner to increase usability and reduce mistakes. We have developed a rapid prototype that we used to evaluate MP usability via cognitive walkthrough and two usability studies. In the studies, we compared the MP prototype with existing secure data sharing tools. The results of the user studies demonstrate that MP design increases usability, user comprehension, and the likelihood of successful use.

Keywords: Usable security, manual encryption

ACKNOWLEDGMENTS

Gratitude to everyone who has helped me.

Table of Contents

Chapter 1

Introduction

Internet users currently number in the billions [14]. As of December 2012, Facebook [7] has over one billion active users, with more than 600 million users logging in every day. Over four billion Facebook messages are sent every day [29]. Hotmail [21] and Gmail [30] both have over 350 million active users. Twitter [31], has over 100 million active users and handles over one billion messages a week. People regularly use online services such as Facebook and web-based email to communicate sensitive information, such as financial and medical data. Despite impressive usage statistics and the often sensitive nature of messages sent through the Internet, user data is rarely encrypted end-to-end. Unencrypted Internet user data is vulnerable to exploitation by online service providers, eavesdroppers, active attackers, and even acquaintances.

Technical aspects of online services can threaten user privacy. User agreements are casually overlooked and often reserve the right for hosting sites to perform cross-site identification, expose user data to search engines and third-party advertisers, and retain user data following account termination. Third-party application platforms provide attackers an avenue to access user data and technical underpinnings of applications often go unverified [23]. Social networks use unrealistic relationship models that cause unintentional information disclosure [17]. Website privacy management systems are notoriously complicated. Users are often confused regarding the risks of using default privacy settings and insufficiently limiting profile access [13]. Few Internet users understand privacy settings implications as user concerns are rarely reflected in privacy settings [1].

People desire control over their personal information on the Internet [10]. In response to unprotected Internet user data and the desire for privacy, several secure data sharing tools (SDST) have been developed. Many SDST were designed to have high usability to provide a successful user experience [3, 8–10, 19, 20, 22]. Several tools aim to achieve this goal through common design choices. We argue that some well-intended design choices common among SDST actually decrease usability, decrease comprehensibility, and cause users to accidentally disclose sensitive information unencrypted.

Several SDST use website overlays to provide a familiar interface that adds support for secure data sharing to popular websites. However, website overlays can cause an ambiguous user experience because users are unsure of the expected behavior of the tool. Such ambiguity can lead to critical mistakes.

Some SDST automatically encrypt user input. This provides an experience nearly seamless to normal user interaction with targeted websites. This can confuse users regarding whether their information is being protected and hinder their comprehension of the system.

Some SDST feature manual key management. Manual key management makes it clear that the provider and the tool cannot access sensitive data. However, manual key management places more burden on the user and has been shown to be confusing and problematic [34].

SDST are typically tightly-coupled with one website. Targeting a single website such as Facebook is logical due to its large user base and the sensitivity of information it handles. Tight website coupling allows SDST to provide convenient features to improve usability. This design makes tools dependent on website interfaces, where tools break if websites change or remove their interfaces. Tight coupling with a single website prevents use with other websites.

SDST are also typically tightly-coupled with specific web browsers. Tight browser coupling provides a viable way to use website overlays and tightly couple with websites. Tight browser coupling can limit the compatibility of each SDST implementation to one web browser. SDST can be implemented as a different plugin for different browsers, but this approach would incur a high maintenance cost. Web browser-agnostic technologies, such as bookmarklets, could be used to decouple SDST from web browsers. Both tight web browser coupling and website coupling may confuse users regarding whether actions are being performed locally or remotely.

In this thesis, we present Message Protector (MP), a novel SDST design that avoids design choices common in existing tools that we assert are problematic and decrease usability. MP features manual encryption, automatic key management, clear separation from websites, as well as website and web browser agnosticism. We implemented a MP prototype to evaluate our design. We conducted a cognitive walkthrough and two user studies that compared our system with existing SDST. These evaluation methods show that our proposed design choices improve usability, comprehension, and the likelihood of correct use.

Chapter 2

Related Work

This chapter presents prior research to analyze security software usability and also discusses systems designed to provide secure data sharing on the Web. We present SDST that feature website overlays, automatic encryption, manual key management, tight website coupling, and tight web browser coupling.

2.1 Security Software Usability Analysis

Whitten and Tygar [34] showed that email encryption software suffers from major usability issues. They conducted a user study of PGP 5.0, an email encryption solution that had what was considered a state-of-the-art user interface at that time. The user study required 12 participants to complete tasks related to a hypothetical political campaign. Only four participants (33.3%) successfully completed all the tasks. Three participants (25.0%) accidentally sent sensitive information unencrypted. One participant (8.3%) was unable to figure out how to encrypt at all. Seven participants (58.3%) used their own public keys to encrypt messages. Two participants (16.6%) failed to publish their public keys. Four participants (33.3%) failed to retrieve others' public keys. Of the five participants that correctly encrypted an email and published their public keys, four were able to decrypt encrypted replies. Several participants that succeeded at study tasks received promptings from the test monitor to guide them. The results show that users generally do not understand the model for public key cryptography and that comprehension is crucial to successful security software use.

Later security software usability studies continued the trend. Sheng et al. [27] repeated the original Johnny paper study with PGP 9. Their study showed improvement from PGP 5.0, but participants still did not understand the public key model and failed to complete several key management tasks, again showing that comprehension is vital to successful security software use. Garfinkel and Miller [12] conducted a usability study testing Key Continuity Management (KCM) email encryption. KCM automates key generation, key management, and message signing. Garfinkel and Miller developed a KCM prototype that color-codes messages depending on whether they were signed and whether the signer was previously known to the user. Their study results show that KCM improved user comprehension, which along with automatic key management, improved user success rates and usability. Despite significant usability gains from earlier email encryption software, users still made critical errors by succumbing to new key, new identity, and unsigned message attacks.

2.2 Website Overlays

Many SDST display their user interface in a manner that overlays websites that they target. SDST typically use website overlays to prevent webpages from accessing user input and display decrypted plaintext messages. Each SDST website overlay is uniquely designed according to the functionality the tool provides and the manner it displays output. We review general SDST overlay categories.

SDST with overlays similar in appearance to their target websites benefit from providing a near seamless user experience. Close overlay resemblance to targeted websites can also result in users not recognizing whether tools are functioning correctly and not being able to distinguish SDST interfaces from actual webpages. These misunderstandings can cause ambiguous user experience expectations and critical mistakes. Additionally, Fahl et al. [10] showed that invisible security does not generate a feeling of security and is not trusted by users. They also presented FBMCrypt, an encryption system that overlays Facebook message text boxes to encrypt input, warns users when sending unprotected data, and highlights decrypted text. Private Webmail (Pwm) [26] provides Gmail encryption by using a website overlay to intercept user input and prevent direct interaction with Gmail. The Pwm overlay appears similar to the Gmail webpage and provides comparable message composition functionality. Pwm automatically performs key management and cryptographic operations on user input and encrypted messages. Private Facebook Chat (PFC) [25] overlays Facebook chat windows to encrypt conversations. PFC is designed similarly to Pwm and offers the same benefits, including a near seamless user experience. uProtect.it [16] provides Facebook private message encryption with overlays that intercept and encrypt user input and decrypt protected messages from friends. uProtect.it uses overlays to cover various Facebook webpage elements and mimic functionality to provide an experience that users are accustomed to. PoX [5] encrypts Facebook user data submitted to third-party applications. PoX uses hidden HTML iframe overlays to display protected data and to hide user data from PoX-aware Facebook applications. PoX does not indicate to users whether displayed text is protected. Scramble! [3] provides online social network content access control management. Scramble! shows protected content in transparent overlays that are identical to the original website. Overlay transparency can make determining whether the overlay is present difficult.

Some SDST overlays deliberately deviate in appearance from targeted websites to indicate whether data are being sent securely. Confidentiality as a Service (CaaS) [8] is a cloud platform that provides a framework for secure Internet messaging by facilitating automatic key management and transparent message encryption. TrustSplit [9] is a Facebook message encryption tool that builds on the CaaS platform. TrustSplit uses a client-server architecture. The TrustSplit server component exists in the cloud and binds to Facebook accounts. The client component overlays Facebook text boxes and names of friends to inform users whether messages will be sent encrypted. Similarly, Waterhouse [18] overlays email clients to inform users by highlighting important textual information and displaying the Facebook profile picture of message senders. Waterhouse leverages Facebook associations to determine the familiarity of contacts sending emails.

Rather than using custom interfaces, some SDST overlay websites by subtly replacing text they display. Encipher.it [6] uses website overlays to prompt for passwords to perform cryptographic operations. Following encryption, Encipher.it replaces user input with cipher text. Encipher.it replaces ciphertext with original messages following decryption. BlogCrypt [22] encrypts text highlighted by users and automatically decrypts protected messages from contacts. Unbeknownst to users, BlogCrypt replaces user input with ciphertext and encrypted messages from contacts with their original messages. Encrypt Facebook [28] presents an overlay for users to input the URL and shared key of a Facebook group. After initial setup, Encrypt Facebook automatically encrypts, decrypts, and replaces user input with ciphertext and encrypted messages with corresponding plaintext. SDST that replace webpage text do not clearly distinguish that they are working and that user messages are not stored on website servers.

Information swapping is an alternative approach to encrypting Internet user data. FaceCloak [20] utilizes overlays to protect personal information (e.g., name, date of birth) through a swapping scheme that replaces Facebook user information with fake data and stores actual information encrypted on its server. FaceCloak retrieves actual user information when requested by authorized friends and displays it in a website overlay. NOYB [15] uses a swapping scheme similar to FaceCloak with the inclusion of a dictionary to index and protect Facebook user information. Information swapping solutions do not make it clear to the user that information is not actually stored on webmail or Facebook servers. Additionally, such tools actually store user information on their own servers. Information swapping-based tools often suffer from the same shortcomings inherent to website overlays as well.

Most SDST overlay websites with custom interfaces to protect messages and provide users a seamless experience. Although overlay systems generally succeed at achieving these goals, we hypothesize that overlays can also cause ambiguous user experience expectations and lead users to make critical errors. Sheng et al. [27] find that email encryption software transparency is problematic.

2.3 Automatic Encryption

Unbeknownst to the user, some SDST automatically encrypt user data. Among the systems previously mentioned, FBMCrypt, PFC, Pwm, uProtect.it, and Waterhouse feature automatic encryption. Automatic encryption and decryption allow SDST to provide a user experience that is consistent with the normal website workflow. Automatic encryption can also have an effect similar to website overlays, where users cannot recognize whether tools are functioning correctly. This can cause users to accidentally disclose private information.

2.4 Manual Key Management

Manual key management can include key exchange with contacts, key inventorying, specifying keys for encryption, and tracking message/key mappings. Most manual key management SDST use symmetric key encryption. Many manual key management tools have users specify passwords instead of actual encryption keys to improve usability by not imposing specific key length requirements. We review SDST that feature manual encryption key management.

Among the systems previously mentioned, BlogCrypt, Encipher.it, Encrypt Facebook, FaceCloak, and NOYB require manual key management. flyByNight [19] is a third-party Facebook application that acts as a secure messaging platform. Users are required to exchange passwords (used to generate encryption keys) independently and externally from the flyByNight interface. flyByNight is not integrated with Facebook functionality and lacks encryption support for wall posts, private messages, and other data communicated through Facebook.

Manual key management makes it clear that service providers and the tool developers cannot access sensitive data. However, manual key management has been shown to be confusing [34]. Manual key management adds system complexity by requiring users to remember passwords and which passwords were used to encrypt specific messages. Furthermore, security software usability studies show that manual key management can cause users to erroneously send sensitive information unencrypted [27].

2.5 Website and Web Browser Coupling

Most SDST are tightly coupled with one webmail or online social network website. Custom SDST overlay interfaces are typically designed to interoperate with only one webpage. Some tools rely on the availability of certain web services. Tight website coupling makes usability improving features feasible. However, website coupling can compound user experience ambiguity caused by website overlays. Also, this design is prone to interoperability breaks due to webpage dependencies.

Many SDST are tightly coupled with specific web browsers to allow use of website overlays and the ability to target specific websites. Most SDST are designed specifically as web browser plugins. Web browser plugins are compatible with only one web browser and specific browser versions. Implementing plugins for different web browsers is feasible, however would increase maintenance overhead. Web browser plugins can confuse users and increase user experience ambiguity caused by website coupling and website overlays. Not all Internet users are familiar with web browser plugins. Web browser coupling may confuse users regarding whether cryptographic operations are being executed locally or remotely. Tight web browser coupling can thwart users from adopting secure data sharing tools due to not having an implementation for their preferred web browser or web browser version. Tight web browser coupling could be an issue if two friends that use different web browsers want to share data securely.

Table 2.1 compares the design choices of the SDST tools introduced in this chapter. In this thesis, we present a SDST design that avoids the attributes listed. Although our proposal does not include these design choices, we demonstrate it provides usability comparable to systems that have these features. Furthermore, we demonstrate that our design provides improved comprehensibility and increased likelihood of correct use. We demonstrate these

System	**Website Overlays**	**Automatic Encryption**	**Manual Key Management**	**Tight Website Coupling**	**Tight Web Browser Coupling**
BlogCrypt	✓		✓		✓
Encipher.it	✓		✓		
Encrypt Facebook	✓		✓	✓	✓
FaceCloak	✓			✓	✓
flyByNight				✓	
FBMCrypt	✓	✓		✓	✓
NOYB	✓		✓	✓	✓
PFC	✓	✓		✓	
PoX				✓	✓
Pwm	✓	✓		✓	
Scramble!	✓		✓	✓	✓
uProtect.it	✓	✓		✓	✓
Waterhouse	✓	✓		✓	

Table 2.1: Comparison of SDST design attributes

points by prototyping and comparing our proposed design with existing tools through participant-based usability studies.

Chapter 3

Design

We present the goals and details of our proposed secure data sharing tool design, Message Protector (MP).

3.1 Goals

- **Data Encryption** - Users must be able to encrypt data shared with their contacts over the web. MP is intended for user-to-user communication, not for secure sharing with web services. This thesis focuses on encrypting textual data since it is the most common data type shared through the Internet.
- **Distinct Use** - Users must be able to clearly distinguish whether they are using the tool.
- **Automatic Key Management** - Key management must be automated. This simplifies the user experience and reduces the likelihood of accidental data disclosure.
- **Website Agnostic** - Interoperability with any website must be provided for robustness and broad coverage. MP is website agnostic in order to improve maintainability.
- **Web Browser Agnostic** - Interoperability with any web browser must be provided for robustness. Tools using MP design could be implemented as a different plugin for different browsers, but this approach would incur a high maintenance cost.
- **Usability** - MP must make users aware of the security tasks they need to perform. Users must be able to successfully setup MP, encrypt messages, and decrypt messages.

Users should not make dangerous errors (i.e., accidentally disclose sensitive data). Ciphertext must be self-documenting so first time encrypted message recipients can successfully decrypt.

- **Comprehensibility** - MP users must be able to confidently and successfully perform security tasks. Users must understand who can access their encrypted messages and how they are accessed by recipients.
- **Security** - MP must provide message confidentiality, integrity, and authenticity.

3.2 Threat Model

Our threat model includes the following attackers:

- **Online Service Provider** - Online service providers can scan and disclose user information to third parties. Providers can also retain user data after deletion.
- **Network Eavesdropper** - Eavesdroppers can intercept information in transit between users, Internet service providers, and online service providers.
- **Active Attacker** - MP thwarts active attackers that have gained access to online service provider systems. Unless the attacker accesses a system that MP uses as the basis of sharing, the attacker would only have access to MP users' ciphertext. The attacker would not be able to decrypt messages without obtaining users' decryption keys as well.
- **Acquaintances** - We define acquaintances as known contacts that have not been authorized to access protected messages. MP prevents acquaintances from accessing publicly posted messages intended for a private audience.

3.3 Identity Verification

MP requires identity verification to use as the basis to control information sharing. Rather than creating a custom identification system, our design leverages existing systems, such as web-based email, Facebook, instant messaging services, OpenID, etc. Using established systems with large user bases are preferable to benefit from online relationships that exist between users.

As an example, we use web-based email as our identification system. Email-based Identification and Authentication (EBIA) [11] is the most prevalent authentication scheme for website accounts and naturally integrates with MP. Email addresses provide a good basis for sharing in the Internet landscape. Sensitive information communicated through the Internet is often transmitted through email. Email account security is typically handled with caution. Online relationships exist between users through email exchange. Contact lists can be utilized to identify relationships and allow users to securely communicate with contacts. Identities are verified by proving that the user can retrieve an email sent to a specified address.

3.4 Key Management

MP requires automatic key management. Key management processes must be transparent to users. Users cannot be required to memorize, manually exchange, or submit passwords related to encryption keys. Message confidentiality, integrity, and authenticity must be provided. These requirements provide an abstraction to follow when implementing MP. This abstraction is flexible regarding the approach taken to provide required security assurances. We give an example key management scheme in Chapter 4.

3.5 Contact Management

After identity verification, known contacts should be displayed to allow the user to select who can access encrypted messages. Granting contacts access to protected messages is the only contact management task users should perform.

3.6 Components

3.6.1 Architecture Overview

Our design uses a client-server architecture. Communication between the client-side application and the server is transparent to users in order to insulate them from key management.

3.6.2 Client-side Application

The client-side application facilitates identity verification, contact management, encryption, and decryption. The client is an executable with minimal technical dependencies.

Installation

Installation consists of downloading the client-side executable.

User Interface

The client user interface must be external to websites and browsers, and cannot use website overlays. The interface should be minimalistic and intuitive so users can setup, encrypt, and decrypt without instructions. The user experiences for message encryption and decryption should be consistent with each other. Information should be displayed to keep users informed.

To illustrate client-side application functions, we present user interface images of our MP prototype, which we describe in detail in Chapter 4. The prototype prompts users to verify their identities (Figure 3.1). Our MP prototype also allows users to manage contacts (Figure 3.2). Our client prototype allows users to encrypt (Figure 3.3) and decrypt (Figure 3.4) messages as well.

3.6.3 Server

The server facilitates identity verification, key management, and contact management. The server consists of a key manager and a database.

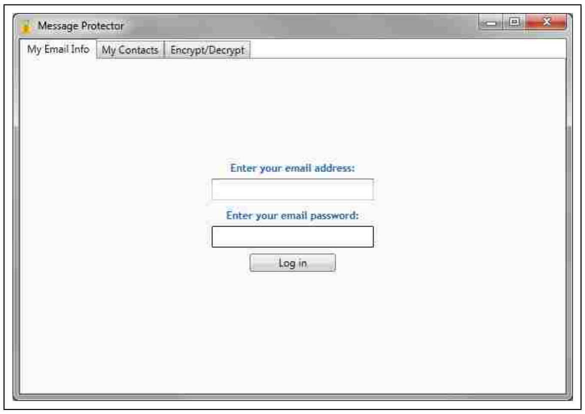
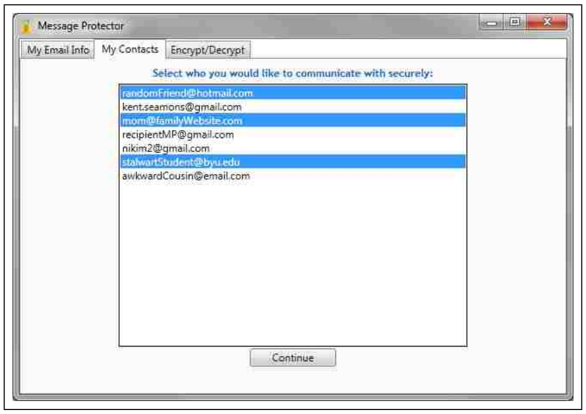


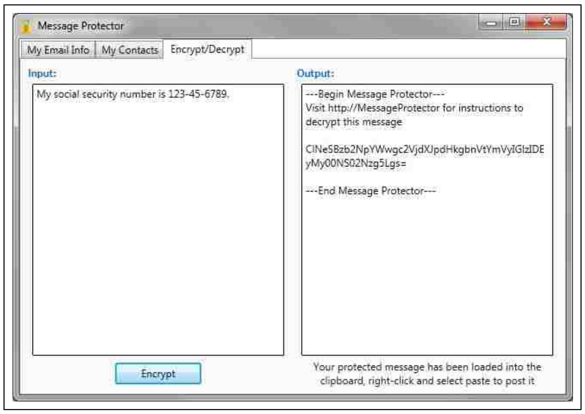
Figure 3.1: Identity verification



Select who you would like to communicate with securely:

randomFriend@hotmail.com
kent.seamons@gmail.com
mom@familyWebsite.com
recipientMP@gmail.com
nikim2@gmail.com
stalwartStudent@byu.edu
awkwardCousin@email.com

Figure 3.2: Contact selection

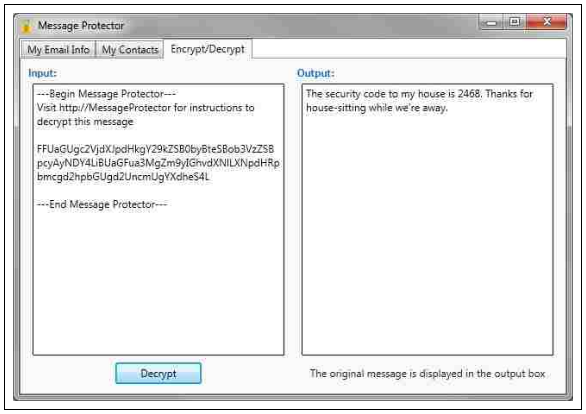


My social security number is 123-45-6789.

```
---Begin Message Protector---
Visit http://MessageProtector for instructions to
decrypt this message

ClNeSBzb2NpYWwgc2VjdXJpdHkgbnVtYmVyIGlzIDE
yMy00NS02Nzg5Lgs=

---End Message Protector---
```

Your protected message has been loaded into the clipboard, right-click and select paste to post it

Figure 3.3: Message encryption

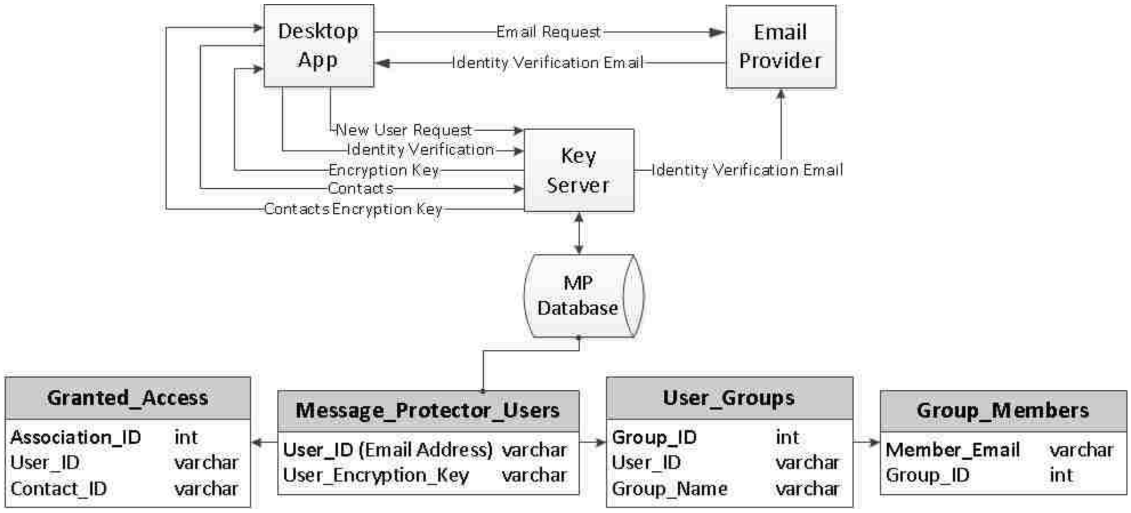


```
---Begin Message Protector---
Visit http://MessageProtector for instructions to
decrypt this message

FFUaGUgc2VjdXJpdHkgY29kZSB0byBteSBob3VzZSB
pcyAyNDY4LiBUaGFua3MgZm9yIGhvdXNlLXNpdHRp
bmcgd2hpbGUgd2UncmUgYXdheS4L

---End Message Protector---
```

The security code to my house is 2468. Thanks for house-sitting while we're away.

The original message is displayed in the output box

Figure 3.4: Message decryption



Granted_Access	
Association_ID	int
User_ID	varchar
Contact_ID	varchar

Message_Protector_Users	
User_ID (Email Address)	varchar
User_Encryption_Key	varchar

User_Groups	
Group_ID	int
User_ID	varchar
Group_Name	varchar

Group_Members	
Member_Email	varchar
Group_ID	int

Figure 3.5: MP Architecture Overview

Key Manager

The key manager interfaces with the client-side component and the database to orchestrate key management and contact management tasks.

Database

The database stores encryption keys and contact data as instructed by the key manager. The database is a relational database that has four tables.

- **Message_Protector_Users** - Stores user identifiers and encryption keys.
- **User_Groups** - Stores contact groups names specified by users. The current MP design allows a single group per user as multiple groups are out of the scope of this research and not currently supported. The design of this table allows future work to include group encryption extensibility.
- **Group_Members** - Stores contact information and user-specified group assignments.
- **Granted_Access** - Stores a list of other users that have granted access to a user.

3.7 Design Limitations

Message Protect design has the following limitations:

- **Identity Verification Provider Spoof** - Identity verification providers could spoof users to obtain encryption keys. If keys are obtained, providers could fraudulently send encrypted messages as the user and read encrypted messages from contacts. This risk already exists with services that use email-based password reset systems.
- **Man-in-the-Middle Attacks** - Communication between the client and server is vulnerable to TLS man-in-the-middle attacks. Attackers could intercept communication between the client and server to obtain encryption keys.

- **User Relationship Storage** - The server database stores user relationships. This allows server administrators to identify relationships between users, which may cause concern for some users.

- **Assurances** - MP does not provide message revocation nor non-repudiation.

- **Recipient-unspecific Encryption** - MP does not allow message senders to specify individual recipients at the time of encryption. Instead, messages can be decrypted by any of the sender's contacts that were selected during the initial contact management task when MP was installed. However, this limitation is mitigated by the ability users have to specify recipients when sending email or Facebook messages. Recipient-unspecific encryption is in contrast to Public Key Infrastructure (PKI), a prominent email encryption scheme, where users specify an individual recipient when encrypting messages. Recipient-specific message encryption ensures that only the intended recipient is able to decrypt the message. Despite this advantage, Whitten and Tygar [34] demonstrated that specifying recipients at the time of encryption is error prone.

Chapter 4

Implementation

We present the implementation details of our MP client prototype. The purpose of the implementation is to evaluate our design goals, which we accomplish through a cognitive walkthrough (Chapter 5) and two user studies (Chapter 6). We simulated the MP server component because the usability of the client is the focus of evaluation.

4.1 User Interface

We implemented our MP client prototype as a Windows executable. We exclusively used .NET 3.5 Framework libraries to build the executable. The user interface of the client prototype is built on the System.Windows.Controls namespace. The prototype UI is implemented with the TabControl class to provide a tab-structured interface, which is common among web browsers and other popular applications. Each tab provides the interface for separate tasks. The prototype has three tabs: *My Email Info*, *My Contacts*, and *Encrypt/Decrypt*.

The *My Email Info* tab (Figure 4.1) provides the identity verification user interface. The *My Email Info* tab is similar in appearance to a web email authentication page, containing instances of the TextBox, PasswordBox, Button, and TextBlock classes. The TextBox shows input as the user types it. The PasswordBox displays asterisks in place of user input to conceal the password. The user clicks the Button instance to submit the credentials provided. The TextBlock instance informs the user regarding identity verification status.

The *My Contacts* tab (Figure 4.2) provides the contact management user interface. The *My Contacts* tab contains instances of the Label, ListBox, Button, and TextBlock classes.

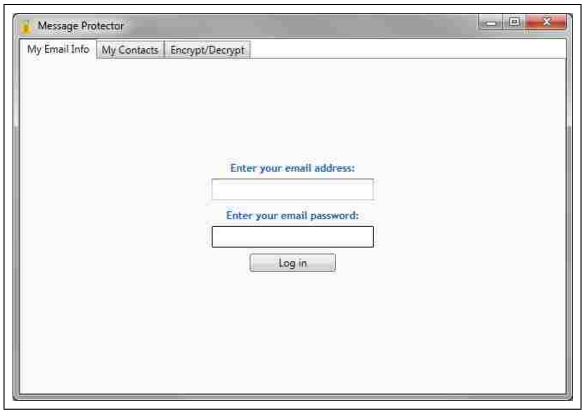
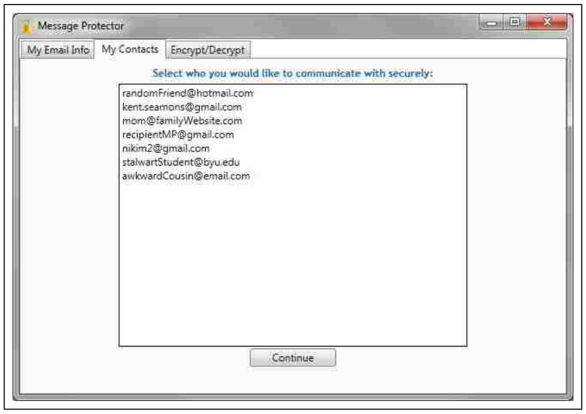


Figure 4.1: My Email Info Tab



randomFriend@hotmail.com
kent.seamons@gmail.com
mom@familyWebsite.com
recipientMP@gmail.com
nikim2@gmail.com
stalwartStudent@byu.edu
awkwardCousin@email.com

Figure 4.2: My Contacts Tab

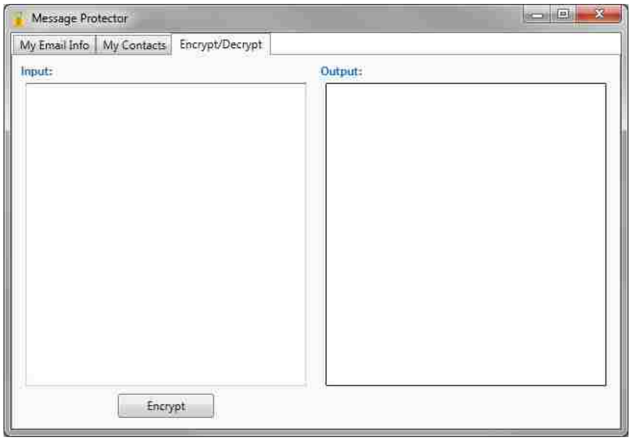


Figure 4.3: Encrypt/Decrypt Tab

The Label instance prompts users to select contacts they would like to communicate securely with. The ListBox displays email addresses of contacts that were imported from the user's email account during identity verification. The ListBox SelectionMode property is set to Multiple to allow users to easily select multiple items. The Button is used to complete contact management. The TextBlock instance informs the user regarding contact management status. All UI elements on the *My Contacts* tab are centered horizontally and vertically to enhance their prominence and to draw user attention.

The *Encrypt/Decrypt* tab (Figure 4.3) provides the user interface for encryption and decryption. The user interface for both tasks are presented in the same tab due to the similarity in their processes. This tab contains instances of the Label, TextBox, Button, and TextBlock classes. Two Label instances inform users where to submit input and where to receive output. Two TextBox instances appear below the Label instances, one to receive user input and the other to display output. Several properties of the TextBox instances have been customized to provide an experience similar to composing and reading messages with web-based email. The input TextBox instance automatically detects whether it contains ciphertext. The Button below the input TextBox displays Encrypt by default, however, if ciphertext is detected, it displays Decrypt.

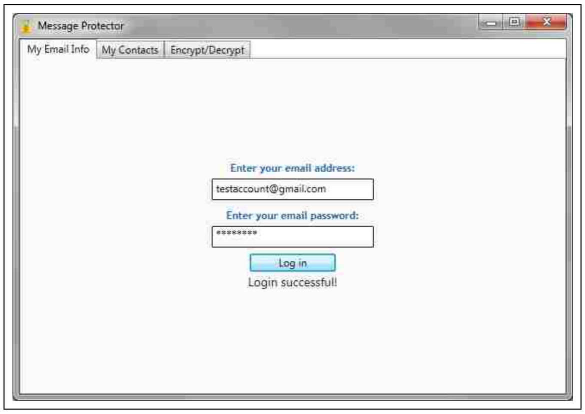
4.2 Workflow

We discuss the user workflow when using the MP client prototype. The prototype simulates identity verification, contact management, encryption, and decryption. We discuss task simulation in each of the coming subsections.

4.2.1 Identity Verification

The first step in the MP prototype workflow is identity verification. The users begins the identity verification process by submitting his email credentials on the *My Email Info* tab. The users types his email address in a text box. The text box displays input as it is typed. The users then types the corresponding password in the password box directly below. The password box displays asterisks in place of user input to conceal the user's actual password. After entering his email credentials, the user then clicks the enter button to invoke the client to verify his identity. The MP user interface displays a message indicating successful authentication if identity verification is successful (Figure 4.4). Upon successful identity verification, the prototype automatically changes the tab selection from *My Email Info* to *My Contacts*. Conversely, MP displays a message indicating login failure if identity verification fails (Figure 4.5).

From the user's perspective, the MP prototype authenticates with his email provider. However, the prototype does not actually authenticate with the user's email provider with the credentials provided. Instead, the prototype verifies the syntactic validity of the email

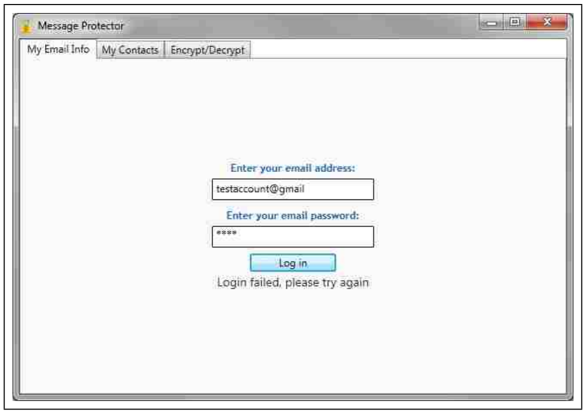


Enter your email address: testaccount@gmail.com

Enter your email password: ********

Login successful!

Figure 4.4: Identity verification success



Enter your email address: testaccount@gmail

Enter your email password: ****

Login failed, please try again

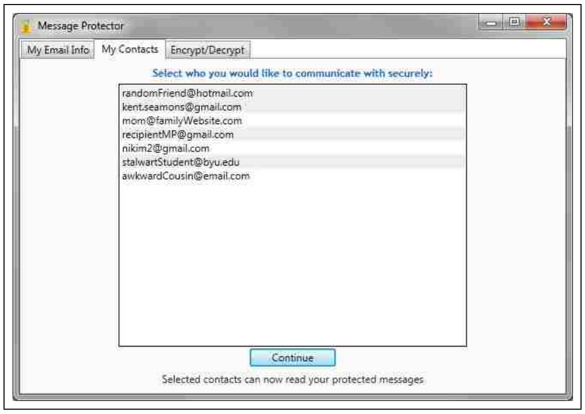
Figure 4.5: Identity verification failure

address provided and that submitted passwords have a length of at least eight characters. Authentication is simulated to prevent user study participants from disclosing their actual email credentials.

A real-world implementation would call the server to request identity verification with the provided email address. The server would then send an unique message to the email address. The client application would then authenticate with the user's credentials, retrieve the email from the server, and provide its contents to the server to verify the user's identity.

4.2.2 Contact Management

After successful identity verification, the client prototype changes the tab selection from *My Email Info* to *My Contacts*. The *My Contacts* tab contains a prominent listbox that displays the email addresses of contacts imported from the user's email account. The user manages his contacts by authorizing contacts access to his encrypted messages. Contacts are selected by clicking on their email addresses in the listbox. After contacts are selected, the user completes contact management by clicking the Continue button located below. If contacts were selected when the user clicks the Continue button, the prototype displays a message indicating that contacts can read the user's protected messages (Figure 4.6). The client prototype also changes to the *Encrypt/Decrypt* tab upon successful contact management.

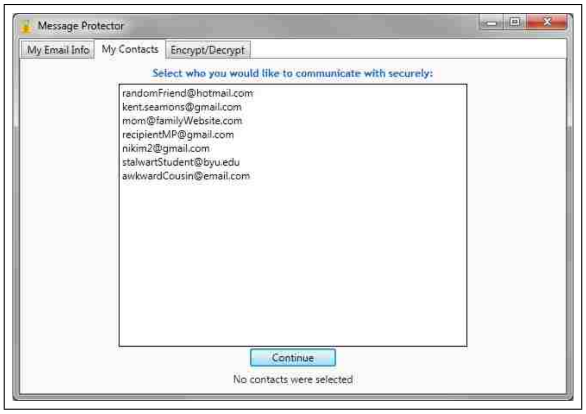


Select who you would like to communicate with securely:

randomFriend@hotmail.com
kent.seamons@gmail.com
mom@familyWebsite.com
recipientMP@gmail.com
nikim2@gmail.com
stalwartStudent@byu.edu
awkwardCousin@email.com

Selected contacts can now read your protected messages

Figure 4.6: Contact selection success



Select who you would like to communicate with securely:

randomFriend@hotmail.com
kent.seamons@gmail.com
mom@familyWebsite.com
recipientMP@gmail.com
nikim2@gmail.com
stalwartStudent@byu.edu
awkwardCousin@email.com

No contacts were selected

Figure 4.7: Contact selection failure

If no contacts were selected, the prototype informs the user that no contacts were selected (Figure 4.7).

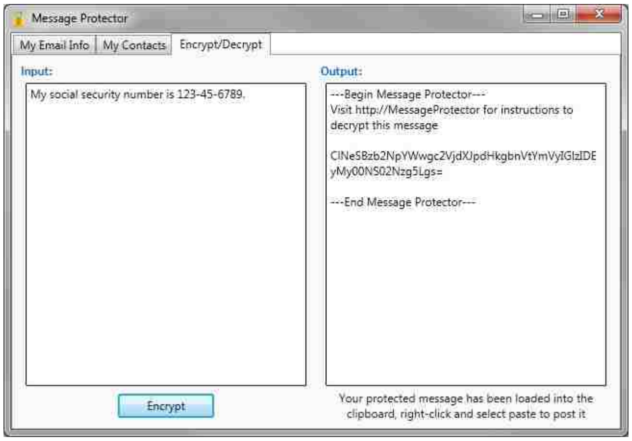
To the user, contact selection simply allows contacts to read his protected messages. However, this step also includes transparent key management. Rather than communicating with a MP server, the client prototype writes the email addresses selected to a local file.

A full implementation would provide the email addresses of selected contacts to the server. The client would also retrieve the keys of contacts that have authorized the user to read their protected messages.

4.2.3 Encryption

After contact management completion, the user can author encrypted messages or decrypt messages from contacts. To encrypt, the user types a message in the Input box on the *Encrypt/Decrypt* tab and then clicks the Encrypt button. The Output text box displays the encryption ciphertext output. The client informs the user that his message has been protected and provides instructions to access it (Figure 4.8). The ciphertext is also automatically loaded into the clipboard, allowing the user to access it by pasting.

After encrypting a message, the user inserts the ciphertext in a webform of the online service that the user will use to share the message. The user can either right-click and select paste, or the user can select the webform, press control-v.



My social security number is 123-45-6789.

```
---Begin Message Protector---
Visit http://MessageProtector for instructions to
decrypt this message

ClNeSBzb2NpYWwgc2VjdXJpdHkgbnVtYmVyIGlzIDE
yMy00NS02Nzg5Lgs=

---End Message Protector---
```

Your protected message has been loaded into the clipboard, right-click and select paste to post it

Figure 4.8: Encryption

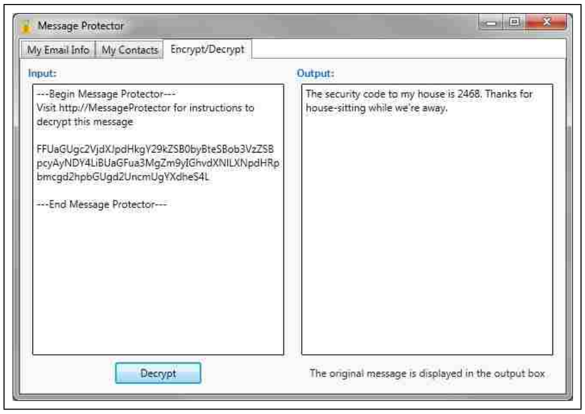
Ciphertext of a working implementation would have to be base64 encoded to be sent through a website. Our client prototype base64 encodes plaintext input to represent ciphertext output. From the user's perspective, the output of our MP prototype is indistinguishable from ciphertext that would result from encryption with actual keys.

Full MP implementations would perform cryptographic operations and base64 encode output to allow the user to apply the ciphertext to webpages for sharing. The cryptographic operations and the approaches taken to execute them are key management implementation details. We provide an example key management scheme in subsection 4.5.

4.2.4 Decryption

To decrypt a message, the recipient accesses first accesses it through the online service that the sender used to share the message. The recipient then highlights the ciphertext with the cursor and copies it.

The user then pastes the ciphertext in the Input text box on the *Encrypt/Decrypt* tab and then clicks the Decrypt button. The Output text box displays the decryption plaintext output. The client informs the user that the original message is displayed in the Output box (Figure 4.9). If decryption fails, the MP prototype displays a message that indicates decryption failure and instructs users to re-attempt decryption (Figure 4.10).

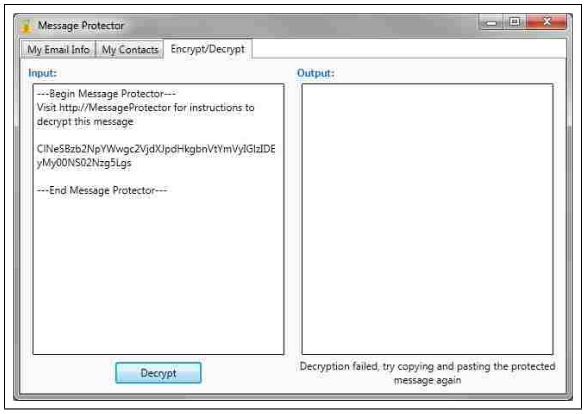


```
---Begin Message Protector---
Visit http://MessageProtector for instructions to
decrypt this message

FFUaGUgc2VjdXJpdHkgY29kZSB0byBteSBob3VzZSB
pcyAyNDY4LiBUaGFua3MgZm9yIGhvdXNlLXNpdHRp
bmcgd2hpbGUgd2UncmUgYXdheS4L

---End Message Protector---
```

The security code to my house is 2468. Thanks for house-sitting while we're away.

The original message is displayed in the output box

Figure 4.9: Decryption success



```
---Begin Message Protector---
Visit http://MessageProtector for instructions to
decrypt this message

ClNeSBzb2NpYWwgc2VjdXJpdHkgbnVtYmVyIGlzIDE
yMy00NS02Nzg5Lgs

---End Message Protector---
```

Decryption failed, try copying and pasting the protected message again

Figure 4.10: Decryption failure

Our prototype detects whether the contents of the Input text box starts and begins with MP tags. Decryption with our MP prototype consists of base64 decoding input.

Full MP implementations would base64 decode ciphertext and perform cryptographic operations on the message. The cryptographic operations and the approaches taken to execute them are key management implementation details. We provide an example key management scheme in subsection 4.5.

4.3 Input/Output Handling

The MP desktop application provides users with one input text box and one output text box on the *Encrypt/Decrypt* tab. On application launch, the input text box displays the text, “Type your message or paste a protected message here, then click the button below...” The Input text box is cleared of this message when clicked. The Output text box is empty by default.

All input is entered into the Input text box. Both plaintext intended for encryption and ciphertext intended for decryption are submitted to the input text box. Similarly, all output is displayed in the output text box. This provides a consistent experience for users to submit input into the same location regardless of the task. Task results will be displayed in the same location as well. Encryption loads ciphertext output into the Clipboard. We use the Clipboard.SetText method of the System.Windows.Forms namespace to achieve this functionality.

The contents of the Input text box are actively monitored by waiting for the Control.TextChanged event, which is logged when text characters in the text box change. Text changes are monitored to determine whether the input text box contains plaintext or ciphertext. As previously described, the Decrypt button is activated and the Encrypt button is deactivated when ciphertext is detected in the input box.

4.4 Context Providers

The MP client prototype educates the user regarding what is taking place by presenting information via Context Providers. We define Context Providers as application information delivery mechanisms that provide operational instructions and task results. The two Context Provider types that the client prototype uses are the TextBlock and ToolTip classes.

The TextBlock class provides a lightweight control for displaying small amounts of flow context. The MP prototype implements TextBlock Context Providers on each tab. TextBlock instances inform the user whether tasks were successful and how to proceed. After the user completes a task, a TextBlock appears and begins to fade shortly thereafter. We have implemented a fading method that decreases the opacity of any TextBlock. Fading is implemented with several methods and members of the System.Windows.Media.Animation namespace.

The ToolTip class provides a brief description of a UI element's purpose when the user rests the pointer on the element. Multiple ToolTips instances appear on each tab to guide the user through each task.

4.5 Example Key Management Scheme

Encryption

Each user is given a master key, K, that is used to derive a symmetric encryption key YK, as well as a signing key, SK. K is securely shared with the user's contacts and used by them to derive YK and SK. A one-time symmetric encryption key TK is generated and used each time the user encrypts a message, producing M_{TK}. TK is encrypted with YK, producing TK_{YK}. A MAC is generated using SK and M_{TK} as input, producing MAC(SK, M_{TK}). The MAC, encrypted one-time key, and encrypted message are concatenated, forming MAC(SK, M_{TK})$M_{TK}TK_{YK}$.

Decryption

MAC(SK, M_{TK}), M_{TK}, and TK_{YK} are extracted from MAC(SK, M_{TK})$M_{TK}TK_{YK}$. MAC(SK, M_{TK}) is generated by the recipient and compared with the MAC extracted from the package. TK_{YK} is decrypted with YK, producing TK. M_{TK} is decrypted with TK to yield the original message.

Limitation

This example key management scheme is susceptible to impersonation by contacts that have received K. However, we assume the SDST is used in the context of an online service provider that handles authentication, thereby mitigating this risk.

4.6 Platform Support

We implemented our MP client prototype as a Windows executable. We exclusively used .NET 3.5 Framework libraries to build the executable. .NET 3.5 is deployed by default with all versions of Windows 7. .NET 3.5 is also available for installation on other Windows operating systems. We specifically decided to build on .NET 3.5 due to Windows 7 being the most used computer operating system in the world [32]. We compiled the MP client executable with the AnyCPU build configuration to allow execution on computers with x86 and x64 processors. Additionally, using only .NET eliminates external library dependencies and statically linked libraries. This allows flexible deployment of the MP prototype executable.

4.7 Installation and Execution

Installation of the MP client prototype consists of downloading the executable to a computer. No other libraries are required because we have referenced only the .NET Framework. The prototype can be started through any Windows application invocation method. No setup wizard is required to complete installation. As a consequence of this simple installation process, MP will not appear under the Programs and Features list and users will have to delete the executable to uninstall.

The executable only writes to the %LOCALAPPDATA% directory. The %LOCALAPPDATA% directory is a Windows operating system, user-specific folder reserved for application use. Applications typically create and exclusively use a subdirectory under %LOCALAPPDATA%. The prototype does not modify files outside of the MP subdirectory in %LOCALAPPDATA%, which prevents MP from affecting the operating system and other applications. The client prototype performs read and write operations on the %LOCALAPPDATA%\MP\EncryptionKey.data file.

Chapter 5

Cognitive Walkthrough

We present a cognitive walkthrough of our Message Protector prototype.

5.1 Background Information

Polson et al. [24] introduced the cognitive walkthrough usability inspection method to perform theory-based analysis of user interface designs. Wharton et al. [33] refined the cognitive walkthrough method and provided a detailed description for application by software developers. Whitten and Tygar [34] utilized a cognitive walkthrough as part of their analysis of PGP 5.0. We performed a cognitive walkthrough of our MP prototype and present the results in this chapter.

5.2 Inputs to the Walkthrough

5.2.1 Users

MP is intended for webmail and online social network users. MP users regularly use these services and can navigate their websites. Users are aware that data they send through these sites could potentially be compromised, however they are not necessarily experts in Internet security. Given their awareness of these risks, MP users are cautious of losing their data and want to protect it. Users are expected to have read the instructional documentation on the MP website.

5.2.2 Tasks to Analyze

The MP prototype allows users to perform message encryption and decryption. We describe and analyze these tasks in terms of credible end-to-end user stories. The first credible story relates the experience of an Internet user wanting to encrypt an email or Facebook message. The Internet user has found the Message Protector website through an online search engine and has no prior experience with MP. In the second story, an Internet user receives an email or Facebook message protected by MP, however has never previously downloaded or used it. We describe the steps that users would take to successfully complete these two stories.

5.2.3 User Interface Definition

The MP client prototype is a graphical Windows application built on the .NET 3.5 Framework Windows Presentation Foundation.

5.3 Task Sequence

We introduce the user story tasks and the sequence of actions necessary for their completion.

5.3.1 New User Encrypting a Message

- Download MP
    - Navigate to website
    - Click download link
    - Specify download location
    - Click download
- Launch MP
- Submit email credentials
    - Type email address

    - Type email account password
    - Click the Enter button
- Select email contacts
    - Highlight contacts' email addresses
    - Click the Enter button
- Encrypt message
    - Type message in the Input text box
    - Click the Encrypt button
- Apply Ciphertext
    - Paste ciphertext in webpage text box

5.3.2 New User Decrypting a Protected Message

- Follow link
- Click link
- Download MP
    - Navigate to website
    - Click download link
    - Specify download location
    - Click download
- Launch MP
- Submit email credentials
    - Type email address

  - Type email account password
  - Click the Enter button
- Select email contacts
  - Highlight contacts' email addresses
  - Click the Enter button
- Decrypt message
  - Highlight encrypted text from webpage
  - Copy highlighted text
  - Paste ciphertext in MP Input text box
  - Click the Decrypt button

5.4 Walking through the Actions

Next we step through the actions required to complete message encryption and decryption. We illustrate the first-time user experience when attempting each task. We also describe how users could correctly complete each action and scenarios where they could make mistakes.

5.4.1 New User Encrypting a Message

Download MP

The user must navigate to the MP website to download the prototype. This will likely be achieved by using a search engine. Once the MP website has loaded, the user will need to identify the MP executable download link. The MP website plainly informs users how to download the executable. Once the download link has been identified, the user will click the download link, specify the download location, and click the download button. Users are expected to know how to navigate to websites and download files from prior experience.

Successful website navigation is dependent on the users familiarity with the web browser in use. MP allows use with any web browser, however certain situations may prevent use of the users preferred web browser. The user may find this task difficult if forced to use an unfamiliar browser. Difficulty may also arise if the user is not familiar with the Windows platform.

Locating the download link may be difficult if the user does not read the instructions on the website. Recognizing download hyperlinks can be problematic if the user has not previously downloaded files from the Internet.

Specifying a download location can be difficult if the user does not know how to navigate the file system. The user will still have to remember the location he specified. The web browser may automatically download files to a default location, which may increase the likelihood of failure to find the executable following download.

The term “download” may be confusing, especially if the user has never previously downloaded files.

Launch MP

The user can start the MP prototype through any Windows application invocation method, the most likely is double-clicking the graphical icon. Launching the MP client is the same as starting other applications.

One issue with launching MP is that the user may forget the download location. Forgetting the location would effectively prohibit the user from using MP.

Submit Email Credentials

The user must type his email address and corresponding password on the *My Email Info* tab. The user submits the credentials input by clicking the Enter button located below the text boxes. Submitting email credentials through the MP prototype is nearly identical as authenticating with the users webmail provider through the Internet. Prior to successfully submitting email credentials, other MP graphical user interface objects are disabled to prevent users from skipping this task.

One issue is ambiguity about the email credentials to be submitted. The prototype plainly prompts for users to enter their email credentials, however, the user may not know whether MP is compatible with certain webmail sites. Similarly, the user may be confused whether to submit a new MP password or the password corresponding to the email address provided.

The user may hesitate to submit email credentials to a program external to the webmail provider website, however, this is a trust issue rather than a usability issue.

Another issue is that the user may not be able to recall his exact email address and password.

Select Email Contacts

After the MP client simulates importing contacts from the users email provider, the user selects email contacts to securely communicate with by highlighting the email addresses of contacts displayed in a listbox. Contact selection is completed by clicking the Enter button located below the contacts list box.

One problem is that the user may not understand the function of list boxes. The concept of clicking to highlight rows is frequently used in webpages, but it is possible that the user may not understand.

A probable issue is that the user may not be able to recall the email addresses of all contacts if the user desires to communicate securely with several contacts. We expect that the user will select a few known contacts for secure communication.

The user may not communicate with all contacts via email, but may want to send secure messages to Facebook contacts. If the users email account does not include the Facebook contacts email address, the user will not be able to send secure messages to that contact. However, sensitive information communicated through the Internet is typically communicated via email, so MP should be able to import the users closest contacts.

Encrypt Message

Message encryption begins with the user typing a message in the Input text box, similar to typing in web forms. The user then clicks the Encrypt button.

One issue is that the term “Encrypt” may confuse the user.

Apply Ciphertext

The user must paste the ciphertext in a text box displayed by the website that will share the message. This is likely the most confusing step in the user story.

After encryption, the MP prototype provides feedback that the protected message has been loaded into the clipboard and can be accessed by right-clicking and selecting paste. However, the user may not understand these instructions, particularly the term “clipboard.” If this is misunderstood, the user can and is likely to manually copy ciphertext from the MP client, allowing successful completion of this step.

The user may incorrectly highlight ciphertext when manually copying. Although ciphertext is marked with human-readable tags that signify its start and end, the user may be confused whether to include tags when applying ciphertext.

Having the user provide email credentials during identity verification may cause confusion that would manifest during encryption. This may lead to the misconception that MP will automatically send secure messages through the email provider on behalf of the user. This may prevent the user from attempting to apply ciphertext.

5.4.2 New User Decrypting a Protected Message

We describe our second credible story where a new user receives and attempts to decrypt an encrypted message. This story includes tasks previously described. Here we describe tasks unique to this user story.

Follow Link

MP-encrypted messages contain a hyperlink and instructions to visit the Message Protector website to decrypt the message. The user can access the MP webpage by clicking the hyperlink.

The user may hesitate to click on the hyperlink, as it may resemble a phishing attempt. However, this is a trust issue rather than a usability issue.

Decrypt Message

Ciphertext is identified with tags intended to inform the user what to highlight. Once highlighted, the user must copy the ciphertext by right-clicking and selecting copy; or by pressing Control-C. The user has likely highlighted and copied text from webpages previously. The user then pastes the ciphertext in the Input text box and clicks the Decrypt button.

The user could highlight extraneous text from another part of the webpage or highlight only a part of the ciphertext. In both cases, decryption would fail.

Another possible issue is the user not knowing how to copy or paste. Although these functions are commonly used, some Internet users may not know how to copy and paste. Instructions are provided on the MP website and client UI to mitigate this potential stumbling point.

If the user fails to include ciphertext tags, the Encrypt button display will not change to “Decrypt” to provide the user indication that this step was completed unsuccessfully. If the ciphertext is entered absent textual tags, the client will handle input as plaintext for encryption.

5.5 Improvements

We used elements of the cognitive walkthrough method to informally evaluate the MP prototype during the design and implementation stages. This was especially useful early on to identify and resolve potential stumbling points. The following list describes the issues that we identified and improvements that have been made:

- Cryptographic mode selection
  - The *Encrypt/Decrypt* tab originally included radio buttons that required users to specify whether to encrypt or decrypt input.
  - The encryption and decryption mode selection radio buttons were removed and automatic detection is used instead.
- Convert button
  - The *Encrypt/Decrypt* tab previously contained a button labeled “Convert”, which was used to invoke encryption or decryption depending on the cryptographic mode selected.
  - We deemed “convert” to be technically inaccurate and potentially misleading. Removal of the cryptographic mode selection radio buttons made using the terms “encrypt” and “decrypt” a logical choice in place of “convert”. The word displayed depends on the contents of the Input text box.
- Automatic plaintext/ciphertext detection
  - The contents of the Input text box on the *Encrypt/Decrypt* tab were previously evaluated only when the convert button was invoked. This did not allow real-time feedback to detect whether plaintext or ciphertext was input.
  - The Input text box was modified to monitor changes to its contents to determine whether it contains plaintext or ciphertext. Automatic monitoring was necessary due to the removal of the cryptographic operation mode selection radio buttons.

- Disabling the output text box
  - The Output text box originally allowed users to modify and delete its contents.
  - The editable property of the Output text box was disabled to prevent users from modifying its contents. Disabling the editable property causes the box to appear gray, signifying that its contents cannot be modified.
- Ciphertext tags
  - MP ciphertext tags were changed from XML format (i.e., <MessageProtector>, </MessageProtector>) to a more human readable custom format.
  - This reduces the resemblance of MP ciphertext to a programming language and increases user-friendliness.
- Input text box
  - The contents of the Input text box on the *Encrypt/Decrypt* tab previously would be cleared when users click it.
  - We disabled this clearing functionality to allow users to edit the contents of the Input text box without the possibility of accidentally clearing its contents. A scroll bar was also added to appear when the Input text box is filled with text. This allows users to compose messages of any length.

Chapter 6

User Studies

We conducted two IRB-approved user studies to analyze our SDST design. The goal of our studies was to compare our prototype with two existing SDST (Encipher.it and Pwm) in terms of task success rates, usability, and system comprehension.

Study participants were recruited through fliers posted on bulletin boards in buildings on the BYU campus. The incentive for participating was $10. Approximately 70 people responded to the fliers through email and phone. The first 30 to inquire for each study were scheduled to participate. Two participants did not appear for the Encipher.it study and one did not appear for the Pwm study. We required participants to already be familiar with Gmail and Facebook in order to minimize website usability issues. To avoid bias, we did not indicate whether the tools under evaluation were built by us.

All study participants used the same computer that had a 3.0 GHz Intel Core 2 Quad CPU, 8 GB of RAM, and Windows 7 Professional and the Google Chrome Canary web browser installed. We added bookmarks for the survey website, Gmail, and Facebook (Encipher.it study only). We provided fake Gmail and Facebook accounts for participants to use instead of requiring them to use their personal accounts. We authenticated with Gmail and Facebook using the test accounts prior to the study beginning. We also installed screen capture software and recorded participants' actions.

Both studies were conducted in a research lab on the BYU campus. Upon arrival, participants filled out a form from the proctor that they used to obtain a $10 BYU Cashier's Office voucher. Study participants received compensation regardless of performance. We guided study participants through an online survey. Participants then began the study by accessing the bookmarked survey webpage. Once the study started, the proctor spoke to participants only to answer questions inconsequential to user performance and attitude toward the systems being tested.

6.1 Encipher.it Comparison Study

To find a SDST to compare with MP, we searched with Google, Bing, and Yahoo for Facebook and Gmail encryption systems. Encipher.it was one of the top results returned by all three search engines. Also, Encipher.it was one of few systems we found that is currently functional. Our first study compared MP with Encipher.it. Encipher.it is a generic bookmarklet-based SDST that can encrypt text in any HTML text box. When the user types a message in a text box and clicks the Encipher.it bookmarklet, the user is prompted to supply a password that is used to encrypt his 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. Encipher.it features website overlays, manual key management, and manual encryption. Encipher.it provides instructions through a website, so we built a similar website in terms of content and design to present MP instructions (Appendix A.3).

6.1.1 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 (89.3%) used webmail daily and 27 (96.4%) used Facebook weekly. Sensitive information had previously been sent over webmail or Facebook by 24 of the participants (85.7%). Only 3 participants (10.7%) had previously encrypted email or Facebook messages. All participants (100%) 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 A.4.1). The study was a within-subjects study, where participants were given simple tasks to complete using both Encipher.it and MP (Appendix A.4.2). The order in which the systems were used was randomly chosen; 16 participants (57.1%) used MP first and 12 participants (42.9%) used Encipher.it first. After completing the tasks for one system, participants were then given a survey to rate their experiences (Appendix A.4.3). 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 A.4.6). Participants were scheduled for one hour to complete the study. The average completion time was 35.2 minutes.

6.1.2 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.

6.1.3 Results

MP Results

All participants (100%) successfully installed MP. Of the participants, 25 (89.3%, CI ±11.46) correctly completed the Gmail tasks and 27 (96.4%, CI ±6.87) correctly 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 building correct mental models for MP. Twenty-five participants (89.3%, CI ±11.46) correctly identified who could read encrypted messages. Additionally, 26 participants (92.9%, CI ±9.54) were able to correctly identify how to decrypt a message using MP. Twenty-three participants (82.1%, CI ±14.19) thought that MP was easy to understand.

Encipher.it Results

Although some users mistakenly added bookmarks to the Encipher.it website in attempts to add the bookmarklet, all eventually installed it correctly. Many participants were not able to get Encipher.it to allow them to encrypt or decrypt messages. Only 16 participants (57.1%, CI ±18.33) were able to decrypt a message in Gmail and only 14 participants (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 participants (60.7%, CI ±18.09) successfully decrypted a message and 23 participants (82.1%, CI ±14.19) successfully encrypted a message.

Four participants (14.3%, CI ±12.96) failed the encryption tasks because they never communicated to the test coordinator the password they had used to encrypt the message. Participants largely built correct mental models for Encipher.it, but these models were not as reliable as those for MP. Twenty-three participants (82.1%, CI ±14.19) correctly identified who could read encrypted messages, but only 20 participants (71.4%, CI ±16.73) understood how to decrypt a message. Seventeen participants (60.7%, CI ±18.09) thought that Encipher.it was easy to understand.

6.1.4 SUS

We used the System Usability Scale (SUS) [4], a usability evaluation metric developed at Digital Equipment Corp., to rate the usability of the systems evaluated in our studies. SUS works by asking participants to respond to ten statements on a Likert scale (Appendix A.2). We included these statements as part of the survey we administered to participants. Based on the participants' responses we calculated a SUS score of 72.23 out of 100 (standard deviation (SD) SD 13.02, CI ±5.05) for MP. Encipher.it had a calculated SUS score of 61.25 (SD 20.11, CI ±7.80). This is a statistically significant difference (paired two tailed t-test, p = 0.0176).

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. 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."

6.1.5 Analysis

MP was much better at helping the participants avoid making mistakes (paired two tailed t-test, Gmail decryption: p = 0.0045, Gmail encryption: p = 0.0027, Facebook decryption: p = 0.0006, Facebook encryption: p = 0.0432). 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 = 0.1610) and also how to successfully decrypt messages (paired two tailed t-test, p = 0.0115), though the first result is not statistically significant.

It is clear from participant responses that they felt more confident using MP precisely because it helped them build a clear mental model. 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 [85.7%, CI ±12.96], Encipher.it: 22 [78.6%, CI ±15.20]), and more people found MP easy to understand (MP: 23 [82.1%, CI ±14.19], Encipher.it: 17 [60.7%, CI ±18.09]).

At the conclusion of the study, we asked participants which system they preferred and why (Appendix A.4.6). Thirteen participants (46.4%) preferred MP, 9 (32.1%) preferred Encipher.it, and 6 (21.4%) were neutral. Preference for MP was primarily due to users feeling secure and its ease of use. Encipher.it preference was primarily due to integrating encryption with the browser. We also observe that users recognize the problem of distributing keys, and several disliked that this was a necessary step of Encipher.it.

MP design has attributes we hypothesize are beneficial and avoids common SDST attributes we hypothesize are detrimental. This user study determined task success rates, evaluated usability, and analyzed user comprehension. The study results indicate that MP encourages successful use, has better usability than Encipher.it, encourages user comprehension, and is more preferable. Study task success rates show that MP design provides an unambiguous user experience. MP encryption and decryption success rates were approximately 30% higher than Encipher.it. Post-study discussions about Encipher.it task errors revealed that many users were confused regarding passwords and not knowing how to encrypt or decrypt specific messages. Successful MP use was consistently attributed to having distinct separation from webpages, as well as participants seeing, personally applying, and submitting ciphertext.

6.2 Private Webmail Comparison Study

In addition to comparing MP with Encipher.it, we also compared MP with an academic secure data sharing solution. We evaluated all systems mentioned in Chapter 2 to determine whether a system provided functionality comparable to MP. Additionally, we desired to use a SDST that had different attributes than Encipher.it. The second study compared MP with Private Webmail (Pwm). Pwm was the only supported academic Gmail encryption solution we found. Pwm uses secure overlays to prevent users from directly interacting with the Gmail website. Pwm attempts to a provide a user experience nearly seamless to the experience Gmail provides by insulating users from technical details and hiding ciphertext. Pwm features website overlays, automatic encryption, tight website-coupling, tight web browser-coupling, and automatic key management.

6.2.1 Setup

This study was comprised of 29 participants. Participants were told that this was a usability study but were not made aware of its security focus. Of the 29 participants, 28 (96.6%) used webmail daily and Facebook weekly. Sensitive information had been sent over webmail or Facebook by 27 of the participants (93.1%). Only one participant (3.4%) had previously encrypted an email or Facebook message. Once again, all participants (100%) reported that protecting the contents of sensitive information was important. Participants were scheduled for thirty minutes to complete the study. The average completion time was 21.8 minutes.

The setup and tasks for this system were similar to the Encipher.it 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 participants (51.7%) used MP first and 14 participants (48.3%) used Pwm first.

6.2.2 Results

MP Results

All participants (100%) successfully installed MP. Of the participants, 27 (93.1%, CI ±9.22) correctly decrypted a message and 28 (96.6%, CI ±6.64) successfully encrypted a message. Comprehension was also high, as 27 participants (93.1%, CI ±9.22) correctly identified who would be able to read encrypted messages and all 29 participants (100%) correctly identified how to decrypt a message using MP. Twenty-five participants (86.2%, CI ±12.55) thought that MP was easy to understand.

Pwm Results

Twenty-five participants (86.2%, CI ±12.55) were able to decrypt a message and 24 (82.8%, CI ±13.75) were able to send an encrypted email.

Participants fared poorly in building correct mental models for Pwm. Only 22 of the participants (75.9%, CI ±15.57) correctly identified who could read a Pwm message, and only 21 (72.4%, CI ±16.27) knew the proper steps to decrypt a message. Perhaps even more interesting is that 6 participants (20.7%, CI ±14.74) stated they were unsure of who could read messages and 4 (13.8%, CI ±12.55) were unsure how to read an encrypted message, whereas no users (0%) reported being unsure of how to use MP in either category. This demonstrates that not only does Pwm struggle to help users build a correct mental model, but users are acutely aware of this problem. Twenty-one participants (72.4%, CI ±16.27) thought that Pwm was easy to understand.

6.2.3 SUS

MP had a calculated SUS score of 73.97 (SD 14.23, CI ±5.41). Pwm had a calculated SUS score of 75.69 (SD 16.31, CI ±6.20). This was not a statistically significant difference (paired two tailed t-test, $p = 0.66962$). 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 qualify as "acceptable" on Bangor's acceptability scale. In aggregate across both studies MP had a SUS score of 73.11 (SD 13.56, CI ±3.60). In aggregate across previous user studies and our studies, Pwm had a SUS score of 73.84 (SD 14.17, CI ±3.04). This was not a statistically significant difference (unequal variance two tailed t-test, $p = 0.7596$).

6.2.4 Analysis

MP proved capable at helping users form correct mental models for 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 = 0.0029). As mentioned in the Pwm results, perhaps the most interesting statistic was that participants were aware that they did not understand how Pwm works. This clearly demonstrates that Pwm does far too little to build correct mental models. On the other hand, MPs manual encryption and clear separation led to nearly all participants building correct mental models.

MP performed on par with Pwm in terms of usability. This study showed that 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 A.5.6). Twelve participants (41.4%) preferred MP, 12 (41.4%) preferred Pwm, and 5 (17.2%) were neutral. Participants that preferred MP primarily attributed their responses to "feeling safe", having obvious encryption, and comprehensibility. 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. MP preference related to obvious security is consistent with the findings of Fahl et al. [10] that indicate that seeing the tool "do something - displaying ciphertext for example - heightened their perceived protection." Participants that preferred Pwm liked its integration with the web browser. Some participants that preferred Pwm liked its ease of use, however, most users that preferred MP cited its ease of use as well. Even some 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.

MP design has attributes we hypothesize to be beneficial and avoids common SDST attributes we hypothesize are detrimental. This study determined task success rates, evaluated usability, and analyzed user understanding. Study results indicate that MP design encourages successful use, has comparable or better usability than existing systems, encourages user comprehension, and is more or comparably preferable. Study task success rates show that MP design provides an unambiguous user experience. MP encryption and decryption success rates were about 10% higher than Pwm. Post-study discussions about Pwm task errors revealed that many users did not know what to expect or did not realize the system was working. Some participants sent unencrypted messages under the false notion they were protected. Successful MP use was consistently attributed to having distinct separation from webpages, as well as participants seeing, personally applying, and submitting ciphertext.

Chapter 7

Conclusion

7.1 Contributions

A common practice in the design of usable SDST development is to make the encryption they provide nearly or completely transparent to users. We present Message Protector (MP), a novel SDST design that purposely exposes technical details in a usable manner to increase usability, user comprehension, and the likelihood of successful use. MP utilizes explicit encryption and avoids design choices that we assert are problematic and decrease usability. MP also encourages clear separation from websites instead of utilizing website overlays. MP is website and web browser agnostic.

We built a MP prototype to validate our design. We conducted a cognitive walkthrough and two participant-based usability studies with our prototype. Our cognitive walkthrough shows that MP possesses few stumbling points and is unlikely to experience failure. In our user studies, MP outperformed both Encipher.it and Pwm in security task success rates. Our studies show that MP usability is better than Encipher.it and comparable to Pwm. Study participants found MP easier to understand than its counterparts. This was confirmed by participant comments that the MP approach of having users handle ciphertext promotes comprehension of the security MP provides. Participants noted that visually seeing and personally submitting ciphertext through online services increased their understanding and informed them whether their messages were being encrypted.

Our research provides evidence that certain encryption details, exposed in a usable manner, can increase SDST usability and the chances of successful use. Having separation between SDST and websites makes it clear whether users are using the tool and whether their messages are encrypted. Our work illustrates that Internet users are able and willing to be involved in encrypting their messages. We show that having users submit ciphertext enhances their comprehension of the security provided, as well as improve encryption and decryption success rates. Our research also shows that users do not necessarily prefer SDST with more encryption transparency. Additionally, our design is maintainable and prevents interoperability breaks. MP also benefits from website and web browser agnosticism, which provides robustness and broad coverage, allowing use with any website and any web browser.

7.2 Future Work

Message Protector Improvements

MP is website and web browser agnostic. This design can be improved with modification to be platform (operating system and device) agnostic. One obvious possible approach to achieving this goal is a cloud-based solution that allows users to access MP functionality through the Internet. This would provide interoperability with all Internet-capable devices, regardless of operating system.

A potentially related improvement would be to replace web-based email with another form of identity verification. Efforts to make MP a cloud solution could include supporting another existing identification system, such as Facebook. Alternatively, a stand-alone MP identity verification system could be developed. Using a different identity verification system could increase MP robustness and allow use in more scenarios.

Automatic encryption benefits from convenience and usability. Manual encryption benefits from users clearly distinguishing whether they are using the tool, high use success rates, and clear user comprehension regarding how to use the tool and the security it provides. Future work should include an automatic/manual encryption hybrid system that provides benefits from both approaches. Such a tool may automatically encrypt input, but show ciphertext or explicitly inform the user that ciphertext is being submitted to the website.

Study Improvements

The user studies described in this thesis emulate the scenario where Internet users were motivated to protect messages and sought out MP. Future studies should include the scenario where an Internet user that has not previously used MP receives a protected message and attempts to decrypt it.

Our studies relied on user instructions hosted at an external website to guide participants through message encryption and decryption. Future studies could include a scenario where participants are not provided with any external instructions. This would determine whether the user interface is self-documenting or the MP workflow is sufficiently intuitive to guide users through encryption and decryption.

Our studies validated short-term MP use with dummy accounts. Future tests could determine whether users can successfully use MP over an extended period of time. Long-term testing could determine the frequency in which users would actually protect messages or receive protected messages. Long-term tests should require users to encrypt and decrypt messages at realistic intervals. Future studies could also have participants use their personal accounts for enhanced realism. Use of personal accounts could provide insights pertaining to user trust in regards to SDST.

Lastly, we have identified an issue that relates to SDST in general, rather than MP in specific. MP and many other SDST provide useful functionality that is desired by Internet users. User studies have also shown that these tools are usable, however, to the authors' knowledge, there has been no widespread adoption of any SDST. Future studies should determine what has hindered ubiquitous use of SDST and more importantly, what will encourage increased adoption.

References

[1] Alessandro Acquisti and Ralph Gross. Imagined communities: Awareness, information sharing, and privacy on the facebook. In *Proceedings of the 6th International Conference on Privacy Enhancing Technologies*, pages 36–58. Springer-Verlag, 2006.

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

[3] Filipe Beato, Markulf Kohlweiss, and Karel Wouters. Scramble! your social network data. In *Proceedings of the 11th International Conference on Privacy Enhancing Technologies*, pages 211–225. Springer-Verlag, 2011.

[4] John Brooke. SUS-A quick and dirty usability scale. *Usability Evaluation in Industry*, 189:194, 1996.

[5] Manuel Egele, Andreas Moser, Christopher Kruegel, and Engin Kirda. PoX: Protecting users from malicious facebook applications. In *2011 IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*, pages 288–294. IEEE, 2011.

[6] Anton Ermak. Encipher.it. `https://encipher.it`, 2011. Accessed: June 2012.

[7] Facebook. Facebook newsroom. `http://newsroom.fb.com/Key-Facts`, December 2012. Accessed: February 2013.

[8] Sascha Fahl, Marian Harbach, Thomas Muders, and Matthew Smith. Confidentiality as a service–usable security for the cloud. In *2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, pages 153–162. IEEE, 2012.

[9] Sascha Fahl, Marian Harbach, Thomas Muders, and Matthew Smith. Trustsplit: Usable confidentiality for social network messaging. In *Proceedings of the 23rd ACM Conference on Hypertext and Social Media*, pages 145–154. ACM, 2012.

[10] Sascha Fahl, Marian Harbach, Thomas Muders, Matthew Smith, and Uwe Sander. Helping Johnny 2.0 to encrypt his facebook conversations. In *Proceedings of the 8th Symposium on Usable Privacy and Security*, pages 1–17. ACM, 2012.

[11] Simson L Garfinkel. Email-based identification and authentication: An alternative to PKI? *IEEE Security & Privacy*, 1(6):20–26, 2003.

[12] 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*, pages 13–24. ACM, 2005.

[13] Tabreez Govani and Harriet Pashley. Student awareness of the privacy implications when using facebook. Unpublished paper presented at the "Privacy Poster Fair" at the Carnegie Mellon University School of Library and Information Science, `lorrie.cranor.org/courses/fa05/tubzhlp.pdf`, 2005. Accessed: April 2013.

[14] Miniwatts Marketing Group. Internet world stats. `http://www.internetworldstats.com/stats.htm`, June 2012. Accessed: February 2013.

[15] Saikat Guha, Kevin Tang, and Paul Francis. NOYB: Privacy in online social networks. In *Proceedings of the First Workshop on Online Social Networks*, volume 1, pages 49–54. ACM, 2008.

[16] Reputation.com Inc. uprotect.it. `https://uprotect.it/index`, June 2012. Accessed: June 2012.

[17] Harvey Jones and José Hiram Soltren. Facebook: Threats to privacy. *Project MAC: MIT Project on Mathematics and Computing*, 1, 2005.

[18] Alex P Lambert, Stephen M Bezek, and Karrie G Karahalios. Waterhouse: Enabling secure e-mail with social networking. In *Proceedings of the 27th International Conference Extended Abstracts on Human Factors in Computing Systems*, pages 4099–4104. ACM, 2009.

[19] Matthew M Lucas and Nikita Borisov. flyByNight: Mitigating the privacy risks of social networking. In *Proceedings of the 7th ACM Workshop on Privacy in the Electronic Society*, pages 1–8. ACM, 2008.

[20] Wanying Luo, Qi Xie, and Urs Hengartner. Facecloak: An architecture for user privacy on social networking sites. In *2009 International Conference on Computational Science and Engineering (CSE)*, volume 3, pages 26–33. IEEE, 2009.

[21] Matthew O'Day. The official Microsoft blog. blogs.technet.com/b/microsoft_blog/archive/2011/07/05/hotmail-still-new-and-cool-even-after-15-years.aspx, July 2011. Accessed: February 2013.

[22] Tamás Paulik, Ádám Máté Földes, and Gábor György Gulyás. Blogcrypt: Private content publishing on the web. In *Proceedings of the 2010 4th International Conference on Emerging Security Information, Systems and Technologies (SECURWARE)*, pages 123–128, 2010.

[23] Sarah Perez. Facebook hacked again. readwrite.com/2008/05/01/facebook_hacked_again, 2008. Accessed: February 2013.

[24] Peter G Polson, Clayton Lewis, John Rieman, and Cathleen Wharton. Cognitive walkthroughs: A method for theory-based evaluation of user interfaces. *International Journal of Man-machine Studies*, 36(5):741–773, 1992.

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

[26] Scott Ruoti, Ben Burgon, Chris Robison, Timothy W van der Horst, and Kent E Seamons. Pwm: A framework for useable secure webmail. *Under submission*, 2013.

[27] Steve Sheng, Levi Broderick, CA Koranda, and JJ Hyland. Why Johnny still can't encrypt: Evaluating the usability of email encryption software. In *Symposium On Usable Privacy and Security*, 2006.

[28] Shanmuga Subramanian. Encrypt facebook. http://www.spacenext.com/encrypt-facebook.php, 2011. Accessed: February 2013.

[29] TechCrunch. Facebook: 350M people using messaging; more than 4B messages sent daily. http://techcrunch.com/2010/11/15/facebook-350m-people-using-messaging-more-than-4b-messages-sent-daily/, November 2010. Accessed: March 2013.

[30] TechCrunch. Gmail now has 425 million users. http://techcrunch.com/2012/06/28/gmail-now-has-425-million-users, June 2012. Accessed: February 2013.

[31] Twitter. Twitter blog. http://blog.twitter.com/2011/03/numbers.html, March 2011. Accessed: January 2012.

[32] w3schools.com. OS platform statistics. `http://www.w3schools.com/browsers/browsers_os.asp`, March 2013. Accessed: March 2013.

[33] Cathleen Wharton, John Rieman, Clayton Lewis, and Peter Polson. The cognitive walkthrough method: A practitioner's guide. In *Usability Inspection Methods*, pages 105–140. John Wiley & Sons, Inc., 1994.

[34] Alma Whitten and J Doug Tygar. Why Johnny can't encrypt: A usability evaluation of PGP 5.0. In *Proceedings of the 8th USENIX Security Symposium*, volume 99. McGraw-Hill, 1999.

Appendix A

User Study Surveys

This appendix contains the online user study surveys.

A.1 Demographic Questions

What is your age?
What is your gender?
What is your major?
How do you rate your level of computer expertise?
Beginner; Intermediate; Advanced
How often do you use webmail?
Multiple times a day; Daily; Weekly; Monthly; Hardly ever; Never
How often do you use Facebook?
Multiple times a day; Daily; Weekly; Monthly; Hardly ever; Never
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?

A.2 System Usability Scale

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. Participants could choose from the following responses: *Strongly Agree; Agree; Neither Agree Nor Disagree; Disagree; Strongly Disagree*
Questions:

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

A.3 Message Protector Website

Welcome to the Message Protector Home Page!

1. Installing Message Protector

- Download Message Protector to your desktop by right-clicking here and select “Save target as...”
- Launch MP by double-clicking the Message Protector icon on the computer’s desktop that looks like this:
- Enter your email credentials and hit Enter
- Select your imported contacts that you would like communicate securely and hit Enter

2. Encrypting Messages

- Type your message in the Input text box on the Encrypt/Decrypt tab
- Click the Encrypt button
- Your protected message will automatically get loaded into the copy/paste clipboard, right-click and select paste to enter it into Gmail or Facebook

3. Decrypting Messages

- Highlight and copy the protected message from Gmail or Facebook (starting with “—Begin Message Protector—” and ending with “—End Message Protector—”)
- Paste the protected message in the Input text box on the Encrypt/Decrypt tab. If you entered the complete protected message, the Decrypt button will display.
- Click the Decrypt button

A.4 Encipher.it Comparison Study

A.4.1 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!

Researchers

- Nathan Kim, Master's Candidate, BYU Computer Science Department
- Dr. Kent Seamons, Associate Professor, BYU Computer Science Department

A.4.2 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:

A.4.3 Message Protector Post Study Survey

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.

Please provide your responses to each of the following statements:
Participants could choose from the following responses: *Strongly Agree; Agree; Neither Agree Nor Disagree; Disagree; Strongly Disagree*
SUS questions from A.2
I felt that MP is easy to understand
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)

Please explain your answer to the previous question:

A.4.4 Encipher.it Tasks

Encipher is a web browser bookmarklet that allows you 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 Encipher and answer a few related questions.

Scenario 1 - Installation In this scenario, you will install Encipher on a computer. Please follow the instructions as closely as possible.

Scenario 1 Task 1 - Encipher Installation Access https://encipher.it/ and follow the instructions in section 1 "Add Bookmark."

Scenario 2 - Gmail In this scenario, you will encrypt and decrypt email messages with Encipher. Open Internet Explorer and click the Gmail bookmark on the Favorites bar. A test account will already be logged in.

Scenario 2 Task 1 - Encipher Email Encryption Access https://encipher.it/ and follow the instructions in section 2 "Encrypt Message" to send an email to recipientMP@gmail.com. Include an encryption key of your choosing in the subject line. Include the phrase "The PIN to my debit card is 1234" in the message.

Scenario 2 Task 2 - Encipher Email Decryption After completing the previous task, you will receive a protected reply email from recipientMP@gmail.com. The reply email will be encrypted with the encryption key "gmailEncryptionKey". Access the webpage https://encipher.it/ 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 Encipher. Open Internet Explorer and click the Facebook bookmark on the Favorites bar. A test account will already be logged in.

Scenario 3 Task 1 - Encipher Private Message Encryption Access https://encipher.it/ and follow the instructions in section 2 "Encrypt Message" to send an encrypted Facebook private message to the user study account's friend named "Alice Jones". Include the phrase "I have a terrible boss" in the body of the message. Use the word, "work" as the encryption key.

Scenario 3 Task 2 - Encipher Private Message Decryption After completing the previous task, you will receive a protected reply private message from the User Study account's friend named "Alice Jones". The reply private message will be encrypted with the encryption key "facebookEncryptionKey". Access https://encipher.it/ and follow the instructions in section 3 "Decrypt Message" to decrypt the protected message. Type the decrypted message below:

A.4.5 Encipher.it Post Study Survey

This concludes the Encipher 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.

Please provide your responses to each of the following statements:
Participants could choose from the following responses: *Strongly Agree; Agree; Neither Agree Nor Disagree; Disagree; Strongly Disagree*
SUS questions from A.2
I felt that Encipher is easy to understand
My level of understanding of Encipher directly affects whether I would use it to

protect my email and Facebook messages
Who can read messages that you protect with Encipher? (Choose one)

- Anyone that has Encipher installed
- Anyone that receives the message and that has my email address
- Anyone that receives the message and that receives my encryption key
- Anyone that receives the message, receives my encryption key, and has Encipher installed
- I don't know

After Encipher is installed, what actions must recipients take to read Encipher protected messages? (Choose one)

- Click the Encipher bookmark
- Open the message, highlight the message you want decrypted, and click the Encipher bookmark
- Open the message, click the Encipher bookmark, and use an encryption key the sender specifies
- Open the message, click the Encipher bookmark, and use an encryption key that the recipient specifies
- I don't know

How often would you use Encipher to protect your email and Facebook messages?
Choose one: Always; Very Often; Occasionally; Rarely; Very Rarely; Never
What did you like about Encipher?
How could Encipher 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):

Please explain your answer to the previous question:

A.4.6 Post Study Survey Additional Questions

Please provide your responses to each of the following statements:
Participants could choose from the following responses: *Strongly Agree; Agree; Neither Agree Nor Disagree; Disagree; Strongly Disagree*
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:

A.5 Pwm Comparison Study

A.5.1 Study Introduction

Introduction from A.4.1, with Encipher.it references replaced with Pwm.

A.5.2 Message Protector Tasks

Message Protector tasks from A.4.2, excluding the Facebook tasks.

A.5.3 Message Protector Post Study Survey

Message Protector survey from A.4.3.

A.5.4 Pwm Tasks

Pwm is a Google Chrome web browser extension that allows you to protect Internet messages (e.g, email) via encryption. In this portion of the study, you will execute various tasks that comprise the primary functionality of Pwm and answer a few related questions.

Scenario 1 - Installation In this scenario, you will install Pwm on a computer. Please follow the instructions as closely as possible.

Scenario 1 Task 1 - Pwm Installation Access https://pwm/ and follow the instructions to install Pwm.

Scenario 2 - Gmail In this scenario, you will encrypt and decrypt email messages with Pwm. Open Google Chrome and click the Gmail bookmark on the Favorites bar. A test account will already be logged in.

Scenario 2 Task 1 - Pwm Email Encryption Access https://pwm/ and follow the instructions to send an email to recipientMP@gmail.com. Include the phrase "The PIN to my debit card is 1234" in the message.

Scenario 2 Task 2 - Pwm Email Decryption After completing the previous task, you will receive a protected reply email from recipientMP@gmail.com. Access the webpage https://pwm/ and follow the instructions to decrypt the protected message. Type the decrypted message below:

A.5.5 Pwm Post Study Survey

This concludes the Pwm 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.

Please provide your responses to each of the following statements:
Participants could choose from the following responses: *Strongly Agree; Agree; Neither Agree Nor Disagree; Disagree; Strongly Disagree*
SUS questions from A.2
I felt that Pwm is easy to understand
My level of understanding of Pwm directly affects whether I would use it to protect my email
Who can read messages that you protect with Pwm? (Choose one)

- Anyone that has Pwm installed
- Anyone that receives the message and Pwm installed
- Anyone that receives the message and that has my email address

- Anyone that receives the message, receives my encryption key, and has Pwm installed
- I don't know

After Pwm is installed, what actions must recipients take to read Pwm protected messages? (Choose one)

- Enable the Pwm extension
- Enable the Pwm extension and open the message
- Open the message and use an encryption key the sender specifies
- Open the message, highlight the message you want decrypted, and invoke the Pwm extension
- I don't know

How often would you use Pwm to protect your email and Facebook messages?
Choose one: Always; Very Often; Occasionally; Rarely; Very Rarely; Never
What did you like about Pwm?
How could Pwm be improved? (Select all that apply)

- Provide better operating instructions
- Provide more information about Pwm to the user
- Provide an easier way to enable Pwm
- Provide a more intuitive user interface
- Provide a less intrusive or cumbersome experience
- Other (please specify below):

Please explain your answer to the previous question:

A.5.6 Post Study Survey Additional Questions

Please provide your responses to each of the following statements:
Participants could choose from the following responses: *Strongly Agree; Agree; Neither Agree Nor Disagree; Disagree; Strongly Disagree*
I feel that it is important to encrypt my emails 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

I would trust a company other than Gmail (i.e., Pwm, MP) to protect my email

Which system would you prefer to use?

Choose one: Message Protector; Pwm; Both; Neither

Please explain your answer to the previous question:

Appendix B

Encipher.it Study Survey Results

This appendix contains the Encipher.it comparison study survey results.

B.1 Demographics Results

Age

- **18-25**: 20
- **26-35**: 7
- **36-45**: 1
- **46-55**: 0
- **55+**: 0

Gender

- **Male**: 12
- **Female**: 16

College Majors: Economics; Information Systems; Health Education; Genetics and Biotechnology; PUblic Health; Exercise Science; Neuroscience; Family Life, Emphasis: Human Development; Bissnes administration; Elementary Education, Post Bacc.; Creative Writing; Mathematics; BA Music; Nutritional Science; Chemical Engineering; Accounting; Geography; Graphic Design; Portuguese; Speech-Language Pathology; communication; Pre-management; Nutritional Science; Public Health; Professional Writing; Undeclared

B.2 Computer Background Survey Results

Computer Expertise

- **Beginner**: 4
- **Intermediate**: 22

- **Advanced**: 2

Webmail Use

- **Multiple times a day**: 19
- **Daily**: 6
- **Weekly**: 1
- **Monthly**: 0
- **Hardly ever**: 1
- **Never**: 1

Facebook Use

- **Multiple times a day**: 14
- **Daily**: 9
- **Weekly**: 4
- **Monthly**: 1
- **Hardly ever**: 0
- **Never**: 0

Previously Sent Sensitive Information via Web Email or Facebook

- **Yes**: 24
- **No**: 4

Method Used (follow up to previous question)

- **Web email**: 23
- **Facebook private message**: 17
- **Facebook wall post**: 1
- **Instant message**: 5
- **Other**: 1 (SMS)

Importance of Sensitive Information Privacy

- **Very important**: 18
- **Important**: 10

Question	**Strongly Agree**	**Agree**	**Neither agree nor disagree**	**Disagree**	**Strongly disagree**
I think that I would like to use MP frequently	4	13	7	4	0
I found MP unnecessarily complex	2	4	4	13	5
I thought MP was easy to use	8	15	2	3	0
I think that I would need the support of a technical person to be able to use MP	0	2	1	15	10
I found the various functions in MP were well integrated	1	20	7	0	0
I thought there was too much inconsistency in MP	0	0	7	18	3
I found MP very cumbersome to use	8	17	1	2	0
I would imagine that most people would learn to use MP very quickly	0	4	3	17	4
I felt very confident using MP	7	16	3	2	0
I needed to learn a lot of things before I could get going with MP	0	2	2	16	8
I felt that MP is easy to understand	9	14	4	1	0
My level of understanding of MP directly affects whether I would use it to protect my email and Facebook messages	7	17	1	3	0

Table B.1: Message Protector Survey Responses

- **Neither important nor unimportant**: 0
- **Unimportant**: 0
- **Very unimportant**: 0

Previously Enrypted Email or Facebook Message

- **Yes**: 3
- **No**: 25

B.3 Message Protector Survey Results

The Message Protector survey results are presented in table B.1

Who can read messages that you protect with MP?

- **Anyone that has MP installed, receives the message, and that I have selected to communicate with securely**: 25

- **Anyone who receives the message and who I have selected to communicate with securely**: 2
- **Anyone who receives the message**: 0
- **Anyone who has MP installed**: 0
- **I don't know**: 1

After MP is installed, what actions must recipients take to read MP protected messages?

- **Access the MP website**: 0
- **Copy the message and paste to MP**: 1
- **Copy the message, paste to MP, click the Encrypt button**: 1
- **Copy the message, paste to MP, click the Decrypt button**: 26
- **I don't know**: 0

How often would you use MP to protect your email and Facebook messages?

- **Always**: 1
- **Very Often**: 4
- **Occasionally**: 15
- **Rarely**: 5
- **Very Rarely**: 3
- **Never**: 0

What did you like about MP?

- It makes it easier and safer to send secure information online.
- It was very simple and straightforward to use. I can see how I might use it for certain things.
- I liked that it was easy to use, and I felt more secure when sending sensitive content.
- it was really simple. There are not too many fuctions
- That you could send personal information and know its a little bit safer than just sending it reguarly
- It was relatively easy to use.

- Security. Would it be available on a phone app as well?
- It was pretty straight forward once I actually understood what I was doing.
- Is very diferent but is very confidential, I think that I would like to use MP frequently
- I like that is safe; however I cannot said that is very safe
- Personal information can be shared privately, and it doesn't take much time or effort to encrypt or decrypt messages.
- It was very easy to understand how to use it.
- the program doesn't require any technical know-how - it feels intuitive
- The ease of access and user-friendliness.
- It gives me a feeling of safety and security when I'm sending sensitive information over the internet. I like knowing that all but the most diehard hackers probably can't read what I'm sending.
- I liked how easy it was to encrypt and decrypt. I wish there was some way just to select the contacts in MP and have it transfer to Gmail.
- it required installing something so those who try to crack the system could possibly be tracked.
- It's good once you get used to it. But it took me a little longer to get used to it.
- It didn't require a decryption key.
- It is easier to navigate and use than the previous program
- Compared to the other program Encipher, this program was easy to follow. You input text and then you receive and encrypted message to paste into your message and send. It was intuitive and easy to follow. It was just a much cleaner experience than the other program. MP is my choice for use if I have to encrypt messages.
- it seemed to work What I was unsure about...when using with facebook,.cutting and pasting a message into MP seems to eliminate the need for the "my contacts" in the MP toolbar. I mean, because you are cutting and pasting from one program to another, does it really know who I am contacting on Facebook? It seems like if I am sending a FB message using MP, then the recipient just needs the MP to download, right? Maybe I am missing something. (My children are getting antsy.)
- Very easy to use.

- It felt like it had another layer of protection by allowing you choose to which people would be selected for private communication. And also, the encrypted message was much shorter than “encypher”
- I liked that MP was easy to use and functioned properly.
- there is no password involved
- You didn’t have to learn a password from the sender.
- I liked that it automatically copied to the clipboard. I also liked that there were no codes involved.

How could MP be improved?

- **Provide better operating instructions**: 5
- **Provide more information about MP to the user**: 12
- **Provide an easier way to select trusted contacts**: 4
- **Provide a more intuitive user interface**: 6
- **Provide a less intrusive or cumbersome experience**: 4
- **Other (please specify below)**: 6

Other reasons provided:

- The only thing i did not like was that I had to switch between screens constantly to use MP
- somehow integrate it into gmail and facebook so that you don’t have to keep copying and pasting but then again, I don’t know if that’s a part of the security
- Security. If someone knows my internet password and they installed MP on their computer with my email and password (as they could see MP was the program used, that’s the one they need to get my info), and then there isn’t any added protection beyond the email to email transfer which email servers likely protect.
- Design/look of the application
- Allow both and “Encrypt” and a “Decrypt” button and have the MP program recognize different input as needing to be encrypted or decrypted and highlight the appropriate button when recognized. That is a suggestion. I say that because when I was about to put new text in to encrypt, I clicked inside the input box which had a leftover message that was previously decrypted and I got momentarily confused. OR, even better, have an encrypt tab, and a decrypt tab, so that one can toggle between the two without confusion.

Question	Strongly Agree	Agree	Neither agree nor disagree	Disagree	Strongly disagree
I think that I would like to use Encipher frequently	1	12	8	4	3
I found Encipher unnecessarily complex	1	5	6	14	2
I thought Encipher was easy to use	4	14	5	4	1
I think that I would need the support of a technical person to be able to use Encipher	2	4	4	13	5
I found the various functions in Encipher were well integrated	4	10	12	1	1
I thought there was too much inconsistency in Encipher	1	7	7	9	4
I found Encipher very cumbersome to use	7	13	2	6	0
I would imagine that most people would learn to use Encipher very quickly	1	6	7	9	5
I felt very confident using Encipher	3	10	6	7	2
I needed to learn a lot of things before I could get going with Encipher	1	3	2	19	3
I felt that Encipher is easy to understand	6	11	6	4	1
My level of understanding of Encipher directly affects whether I would use it to protect my email and Facebook messages	5	17	5	0	1

Table B.2: Encipher.it Survey Responses

- Allow user to specify for window to stay on top of all other windows.

B.4 Encipher.it Survey Results

The Encipher.it survey results are presented in table B.2

Who can read messages that you protect with Encipher?

- **Anyone that has Encipher installed**: 1
- **Anyone that receives the message and that has my email address**: 1
- **Anyone that receives the message and that receives my encryption key**: 2
- **Anyone that receives the message, receives my encryption key, and has Encipher installed**: 23
- **I don't know**: 1

After Encipher is installed, what actions must recipients take to read Encipher protected messages?

- **Click the Encipher bookmark**: 2
- **Open the message, highlight the message you want decrypted, and click the Encipher bookmark**: 3
- **Open the message, click the Encipher bookmark, and use an encryption key the sender specifies**: 20
- **Open the message, click the Encipher bookmark, and use an encryption key that the recipient specifies**: 2
- **I don't know**: 1

How often would you use Encipher to protect your email and Facebook messages?

- **Always**: 0
- **Very Often**: 3
- **Occasionally**: 14
- **Rarely**: 7
- **Very Rarely**: 3
- **Never**: 1

What did you like about Encipher?

- It doesn't require me to open other windows. It can be done quickly.
- The bookmark tab made it extremely easy to use.
- I feel indifferent about this program, it is no better or no worse than the first program. I did experience some technical difficulties.
- The password made it seem more secure but I don't know how you would securely send someone your password
- I didn't i hated it
- It was easy to use, no copying and pasting.
- I like the idea of being able to not have my information as accessible. I am somewhat frustrated that the passwords didn't work either time.. What happens if the password doesn't work? Can you recreate it? Who makes the password? This was all unclear to me.

- it's a little bit complicated but is confident
- I like is safe
- you have to enter a password to send and to receive encrypted messages, which makes it more secure
- It seemed extra secure with the use of an encryption key.
- Although I didn't follow every word of the background info provided, it was nice to know how this program worked. I also appreciated the warning about man-in-the-middle attacks.
- I liked how you didnt' have to load your contacts into it.
- It's easier to use than message protector although I was nervous to see that at the bottom it was still susceptible to man in the middle attacks
- I liked not having to download an app, but at the same time I didn't like having to have a password to access the code. I also thought that the encryption was way to long.
- once I understood how it worked, it was extremely simple. it is something I would want to use if i need to send sensitive info via the internet :)
- Easy to install, easy to use.
- It's very easy and intuitive.
- The bookmark feature is easy to use and access
- I didn't. There was a flaw in the instructions about how to make sure a facebook message got sent encrypted. I typed in my message, per the instructions, and then opened Encipher and it tried to "decrypt" my message. Apparently I had to highlight the message I wanted to send and THEN click Encipher, which step was not on the instructions. It caused me to dislike my user experience because I distrusted the functionality on my first attempt at use inside Facebook.
- the bookmark/toolbar feature; during the first task, the encryption key did not work for me – this makes me think the program is not reliable.
- It can protect my message and it is easy to use.
- The fact that once you install it, you don't have log in or anything to use the service. And it's a very quick process.
- I think Encipher is a good idea to protect e-mails and messages. Especially because everything is technology based in the world and being able to protect important information sent to others is necessary.

- protects sensitive information
- It was quick.
- Very simple installation and low complexity. Few options, but gets the job done, and done quickly.

How could Encipher be improved?

- **Provide better operating instructions**: 13
- **Provide more information about Encipher to the user**: 9
- **Provide an easier way to send encryption key to contacts**: 16
- **Provide a more intuitive user interface**: 7
- **Provide a less intrusive or cumbersome experience**: 6
- **Other (please specify below)**: 5

Other reasons provided:

- I was given the decryption key in the instructions but I don't know how I would get it otherwise because it doesn't seem to come with the message.
- Include images like scanned documents
- Provide more accurate instructions
- during the first task, the encryption key did not work for me
- Stability: did not work in Gmail

B.5 Post Study Survey Results

Which system would you prefer to use?

- **Message Protector**: 13
- **Encipher**: 9
- **Both**: 4
- **Neither**: 2

Table B.3: 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.

Continued on next page

Table B.3 – continued from previous page

Preferred System	Reason
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.
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.

Continued on next page

Table B.3 – continued from previous page

Preferred System	Reason
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.

Appendix C

Pwm Study Survey Results

This appendix contains the Pwm comparison study survey results.

C.1 Demographics Results

Age

- **18-25**: 29
- **26-35**: 0
- **36-45**: 0
- **46-55**: 0
- **55+**: 0

Gender

- **Male**: 14
- **Female**: 15

College Majors: Public Health; business management; International Relations; Chemical Engineering; English; Business Management; Exercise Science; Genetics and Biotechnology; Electrical Engineering; Exercise Science; Management; Statistics; Family Life; Public Health; Economics; Neuroscience; marketing; Social Work; Information Systems; Human Development; Neuroscience/Biochemistry; Biophysics; Pre-Communication Disorders; Communication Disorders; communication disorders; PD Biology; Computer Science; Chemistry/French Studies Double Major; pre-management

C.2 Computer Background Survey Results

Computer Expertise

- **Beginner**: 4

- **Intermediate**: 23
- **Advanced**: 2

Webmail Use

- **Multiple times a day**: 21
- **Daily**: 6
- **Weekly**: 1
- **Monthly**: 0
- **Hardly ever**: 1
- **Never**: 0

Facebook Use

- **Multiple times a day**: 18
- **Daily**: 7
- **Weekly**: 3
- **Monthly**: 0
- **Hardly ever**: 0
- **Never**: 1

Previously Sent Sensitive Information via Web Email or Facebook

- **Yes**: 27
- **No**: 2

Method Used (follow up to previous question)

- **Web email**: 27
- **Facebook private message**: 13
- **Facebook wall post**: 1
- **Instant message**: 3
- **Other**: 0

Importance of Sensitive Information Privacy

- **Very important**: 17

Question	**Strongly Agree**	**Agree**	**Neither agree nor disagree**	**Disagree**	**Strongly disagree**
I think that I would like to use MP frequently	5	11	7	5	1
I found MP unnecessarily complex	0	1	10	13	5
I thought MP was easy to use	9	15	4	1	0
I think that I would need the support of a technical person to be able to use MP	1	1	0	12	15
I found the various functions in MP were well integrated	5	19	4	0	1
I thought there was too much inconsistency in MP	0	1	4	17	7
I found MP very cumbersome to use	6	16	2	5	0
I would imagine that most people would learn to use MP very quickly	1	1	3	15	9
I felt very confident using MP	11	11	5	2	0
I needed to learn a lot of things before I could get going with MP	0	2	1	20	6
I felt that MP is easy to understand	7	18	2	2	0
My level of understanding of MP directly affects whether I would use it to protect my email and Facebook messages	10	16	2	1	0

Table C.1: Message Protector Survey Responses

- **Important**: 12
- **Neither important nor unimportant**: 0
- **Unimportant**: 0
- **Very unimportant**: 0

Previously Enrypted Email or Facebook Message

- **Yes**: 1
- **No**: 28

C.3 Message Protector Survey Results

The Message Protector survey results are presented in table C.1

Who can read messages that you protect with MP?

- **Anyone that has MP installed, receives the message, and that I have selected to communicate with securely**: 27
- **Anyone who receives the message and who I have selected to communicate with securely**: 2
- **Anyone who receives the message**: 0
- **Anyone who has MP installed**: 0
- **I don't know**: 0

After MP is installed, what actions must recipients take to read MP protected messages?

- **Access the MP website**: 0
- **Copy the message and paste to MP**: 0
- **Copy the message, paste to MP, click the Encrypt button**: 0
- **Copy the message, paste to MP, click the Decrypt button**: 29
- **I don't know**: 0

How often would you use MP to protect your email and Facebook messages?

- **Always**: 0
- **Very Often**: 4
- **Occasionally**: 13
- **Rarely**: 7
- **Very Rarely**: 4
- **Never**: 1

What did you like about MP?

- I liked that after reading through the instructions a few times I was able to get the idea and actually practice. Seeing that I understood the instructions made me feel confident in being able to use it.
- I like the Idea that you are able to send an email that is personal and private between two people. I would very much like the assurance that my emails are being read by the intended recipients. I like how it was easy and simple. It was good to see how it works on both sides and you just copy and paste..

- It seems very simple.
- It's simple to understand and doesn't look like it's an easy code to crack.
- It was the same to encrypt as to decrypt. I think that consistency is the most important, and there were few steps as well. It's similar to google translate, which is a program many are familiar with.
- It is simple. There are two options...encrypt or decrypt.
- It's simple. Appears to be very safe. Easy to customize with different recipients.
- There are only three tabs and it seems to work quickly and without much confusion. I like that it is simple. It would kind of be a bother to have to open up MP every time you wanted to encrypt or decrypt a message. I really like that the encrypted message automatically goes onto the paste board though. That is really convenient.
- I liked that there were instructions on the MP website that were easy to follow.
- It was easy to use. I could see it being useful at times.
- Simple, quick, allows selection of different contacts with whom you want to communicate securely.
- Instructions were easier to understand than gmail.
- I have often been concerned about sending private information and it seems like a great way to send it securely. H
- Simple Input and Output boxes. Visually it is easy to see what you are doing.
- Good UI, easy to use
- I like how you are able to send messages to specific people and have them read it without others being able to.
- It was really easy to use and it had good instructions.
- It seemed really easy to encrpyt things. Since I was doing the encrypting in a program, not online, I felt more secure that the encryption would work
- I liked that you could choose what contacts could read encrypted messages. It seemed secure and relatively easy to use.
- Almost nothing
- I liked how it wasn't a website on the internet, but rather it is a program on your computer, that you can use to encrypt and decrypt messages.

- The instructions were much more detailed and you can see all the steps of the encryption and decryption process.
- I know that my message would be secure
- i like input/output methods, so it was simple to understand and to use.
- It feels safer in the sense that I select what users are able to view and decrypt my encrypted messages
- It felt safer than the Pwm. It was relatively easy to use as well. It can be used for a variety of communication systems, not just gmail.
- Good tool to know, but probably won't use it often, because not a lot of people know how to use it.

How could MP be improved?

- **Provide better operating instructions**: 3
- **Provide more information about MP to the user**: 14
- **Provide an easier way to select trusted contacts**: 4
- **Provide a more intuitive user interface**: 4
- **Provide a less intrusive or cumbersome experience**: 3
- **Other (please specify below)**: 9

Other reasons provided:

- It's fine
- You can decrypt the information without copying and pasting.. You should be able to just right high light and right click, open which program and it automatically decrypts it
- It's looks to be a lot of back and forth. If it could be integrated into G-Mail as an encryption method with the click of a button then that's great. I just don't see myself going back and forth and requiring my friends to do the same as well as download the software.
- bulleted or numbered action steps would be helpful. Also an encrypt/decrypt button side by side...not a changing encrypt/decrypt.
- Integrating it into the browser could make it even easier to use, or possibly making it operable via keyboard functions or right-click.

Question	Strongly Agree	Agree	Neither agree nor disagree	Disagree	Strongly disagree
I think that I would like to use Pwm frequently	7	13	3	6	0
I found Pwm unnecessarily complex	0	2	5	7	15
I thought Pwm was easy to use	13	11	2	2	1
I think that I would need the support of a technical person to be able to use Pwm	0	2	3	10	14
I found the various functions in Pwm were well integrated	5	18	4	1	1
I thought there was too much inconsistency in Pwm	0	0	6	15	8
I found Pwm very cumbersome to use	13	11	5	0	0
I would imagine that most people would learn to use Pwm very quickly	0	1	6	10	12
I felt very confident using Pwm	7	8	8	5	1
I needed to learn a lot of things before I could get going with Pwm	0	2	3	14	10
I felt that Pwm is easy to understand	11	10	5	3	0
My level of understanding of Pwm directly affects whether I would use it to protect my email and Facebook messages	10	14	2	2	1

Table C.2: Pwm Survey Responses

- Should read your gmail contacts
- If someone were to hack into my computer, what is to prevent them from using MP to decrypt them as well? I suppose I am still a bit blurry on who exactly I need to protect them from. If I knew the real risk of sending my secure information, then I might be more inclined to use it.
- Provide instructions in the program so you don't have to go to the website
- In regards to contacts, display the name rather than just the email address. Many (if not most) email users don't know their contacts specific email address, but rather what name or contact that address is saved to.

C.4 Pwm Survey Results

The Pwm survey results are presented in table C.2

Who can read messages that you protect with Pwm?

- **Anyone that has Pwm installed**: 0
- **Anyone that receives the message and Pwm installed**: 16
- **Anyone that receives the message and that has my email address**: 1
- **Anyone that receives the message, receives my encryption key, and has Pwm installed**: 6
- **I dont know**: 6

After Pwm is installed, what actions must recipients take to read Pwm protected messages?

- **Enable the Pwm extension**: 1
- **Enable the Pwm extension and open the message**: 20
- **Open the message and use an encryption key the sender specifies**: 3
- **Open the message, highlight the message you want decrypted, and invoke the Pwm extension**: 1
- **I dont know**: 4

How often would you use Pwm to protect your email?

- **Always**: 1
- **Very Often**: 5
- **Occasionally**: 12
- **Rarely**: 7
- **Very Rarely**: 2
- **Never**: 2

What did you like about Pwm?

- Right on the chrome browser. There is no external program.
- It was super easy and fast.. Very convenient. Not very sure on the actual security of PWM(or how reliable it is)
- I was a little confused at first, but it is very simple and quick.
- It's integrated into my gmail so I don't have to go back and forth.
- I especially liked how it was directly integrated into the email account; I find that this fact alone would make it the most usable of the two programs.

- It was user-friendly.
- In general, the user interface seemed more friendly. Less intimidating, simple design. Minimal work required for the user.
- I like that it is integrated into Google. I don't need to have a second window open with another program like the MP system.
- I liked that it was activated by clicking a button.
- It's easier to use than MP because you don't have to copy and paste.
- It was super easy. It's about as simple as you can make it.
- Built into the browser, one click instead of having to copy and paste and click a button. Much more convenient!
- Much easier to use than the other encryption method.
- It seems really easy to encrypt your secure messages. No second program necessary.
- It seems like Pwm does all the encrypting and decrypting for you. If I knew more about how it worked I think I would like to use it.
- Very easy to use
- I like how it keeps things protected from anyone else on the internet.
- It was easy to figure out how to use.
- It provides a feeling of security
- It seemed very secure and allows people to be protected and have more privacy while sending and receiving emails. It was easy to use and install.
- protection for emails s awesome
- I liked that Pwm allowed me to send a secure, encrypted message. If I ever needed to send a message that was high profile, and I needed to use and encrypted email I would use pwm.
- It's user friendly and simple enough for all to use frequently if needed.
- It was easy to send an encrypted message
- hard to say, ha ha
- I didn't have to do very much or learn very much to use it.
- It was very straight-forward and easy to understand and used.
- It was relatively simple, though I was confused at first. I didn't notice the lock symbol right away. Also, the instructions for the decryption were not very clear.

- It's easy to use and it is not complicated. There wasn't too many steps that I had to do.

How could Pwm be improved?

- **Provide better operating instructions**: 8
- **Provide more information about Pwm to the user**: 16
- **Provide an easier way to send encryption key to contacts**: 4
- **Provide a more intuitive user interface**: 4
- **Provide a less intrusive or cumbersome experience**: 1
- **Other (please specify below)**: 14

Other reasons provided:

- I don't like how the message was there. On MP the message was encrypted and on the Pwm the message is right there highlighted
- Provide more information on how secure it would be and that not everyone could read the message
- The reply message had a lot of details and wasn't formatted very well. I could see the message clearly it was the other stuff that might confuse people
- Perhaps specify that once locked, your message screen will look slightly different, so they know it has worked.
- Give me a reason to believe that Pwm is actually encrypting my messages. I can't see that it was encrypted, so it makes me nervous.
- I wasn't quite sure that my message was being encrypted. Not sure if I missed instructions, but I just assumed I should click the lock. Also, when it came to decrypting the message I received, I wasn't sure if the entire message was encrypted or if I needed to "unlock" something to see the whole message.
- It was unclear to me that the message I opened was already decrypted, which is nice but it would have been nice if it said "Decrypted message:" or something. It just wasn't intuitive for me.
- Inform the user about what it would required for others to view the message.
- I was confused when I first sent it what it was supposed to look like when it was encrypted. In the instructions, it would have been nice to see an example of an encrypted message. Also, I'm not sure what is required to be able to see/return emails

that are sent with this this extension. Are replies automatically encrypted? I haven't a clue.

- it wasn't clear how to enable and disable it for individual messages,
- I thought it was good!
- I couldn't open the encrypted message
- I would maybe like to know a little more about how it works.
- I don't necessarily see/understand the difference of how it makes information safer.

C.5 Post Study Survey Results

Which system would you prefer to use?

- **Message Protector**: 12
- **Pwm**: 12
- **Both**: 5
- **Neither**: 0

Table C.3: 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.

Continued on next page

Table C.3 – continued from previous page

Preferred System	**Reason**
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.
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.)

Continued on next page

Table C.3 – continued from previous page

Preferred System	**Reason**
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.