Private Facebook Chat

Chris Robison, Scott Ruoti, Timothy W. van der Horst, Kent E. Seamons
Computer Science Department, Brigham Young University
Provo, Utah, USA
seamons@cs.byu.edu

Abstract—**The number of instant messages sent per year now exceeds that of email. Recently users have been moving away from traditional instant messaging applications and instead using social networks as their primary communications platform. To discover attitudes related to instant messaging and its security, we have conducted a user survey. This paper also presents the design of PFC (Private Facebook Chat), a system providing convenient, secure instant messaging within Facebook Chat. PFC offers end-to-end encryption in order to thwart any eavesdropper, including Facebook itself. Finally, we have conducted a usability study of a PFC prototype.**

Keywords-**instant messaging, Facebook, privacy, useable security**

I. Introduction

Instant messaging is an increasingly popular form of synchronous communication over the Internet. Every year the instant messaging user base grows by 200 million people, and the number of instant messages sent per year now exceeds that of email. A recent report shows that users are moving away from traditional instant messaging applications and are instead using social networks as their primary communications platform.[1]

Facebook Chat was introduced in 2008 and already has a large user base. Unfortunately, Facebook Chat is not secure. For example, Facebook can read all messages sent through the system. Even if we trust Facebook with our messages, Facebook does not transmit data over HTTPS by default. This means eavesdroppers on the network have potential access to Facebook Chat conversations. Additionally, Facebook Chat is susceptible to session hijacking attacks as demonstrated by Firesheep [1]. Facebook users need to explicitly turn on HTTPS in Facebook's account settings for the browser to communicate with Facebook over an encrypted channel. Even if a user secures their own connection to Facebook, there is no way to guarantee that the other party in a chat session also has HTTPS turned on.

This paper first presents the results of a survey concerning the awareness and attitudes of users regarding the security and privacy of instant messaging. The paper then presents PFC (Private Facebook Chat), a system that enables convenient, secure instant messaging within Facebook Chat.

The system prevents Facebook from accessing the plaintext of a chat session. The design and implementation of PFC represents the first system that provides end-to-end security in a browser-based instant messaging service. PFC uses a *security overlay* that is placed over the current Facebook Chat interface to allow users to easily secure chat sessions that contain sensitive information. PFC also uses an automated key escrow system to transparently manage encryption keys, removing the need for users to establish shared secrets or obtain public keys in advance. The paper includes the results of a usability study to determine whether users could easily use the system to accomplish specific tasks.

Facebook manages the storage and transmission of chat messages while the key server manages and distributes encryption keys. Assuming these parties do not collude with each other, they each have too little information during normal operation to be able to access the contents of an encrypted chat message. There is also a server that provides the client-side software necessary to encrypt and decrypt chat messages. This server is in a more powerful position to unilaterally compromise the system. Since the software it provides runs on the user's machine, it can be audited and monitored to detect attacks.

The remainder of this paper is organized as follows: Section 2 contains the results of a user survey regarding attitudes and opinions about secure chat. Section 3 describes the design and implementation of PFC. Section 4 contains the results of a usability study of the PFC prototype. Section 5 discusses related work. And Section 6 provides conclusions and future work.

II. User Survey

We conducted a three-part survey to determine user awareness and attitudes regarding the privacy of instant messaging. Part one gathers information about the chat systems users are currently using. Part two seeks to determine how users feel about transmitting sensitive information over chat. Finally, part three gathers information about opinions on the privacy of chat. The survey was distributed via word-of-mouth, email, and various social networks and resulted in 65 responses.

A. Chat Systems

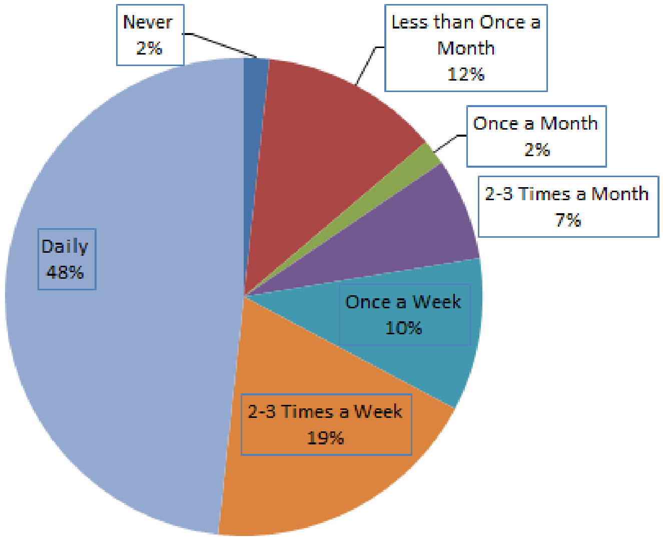
Table I lists chat systems users reported to have used regularly. The survey allowed for users to select more than one option. Most users reported that their personal preference in which chat system to use is based on how easy it is to learn and use. Multiple users also stated that they use a particular system because their work mandates it or it integrates with other online services they use regularly, such as email or social networks. Google Talk, Facebook Chat,

[1] Email Statistics Report, 2009-2013, http://www.radicati.com/?p=3229

TABLE I
CHAT SYSTEM USAGE

System	Percent of Users
Google Talk	67%
Facebook Chat	50%
Skype	41%
Windows Live Messenger	19%
Yahoo Instant Messenger	19%
AIM	12%
IRC	3%
ICQ	2%
Others	19%

Fig. 1. Frequency of use



and Skype are the predominant chat platforms used by the survey participants. Google Talk and Facebook Chat are used primarily in the browser. The users that selected the *Others* option used commercial products regularly (e.g., Microsoft Lync, Microsoft Communicator, and IBM Sametime).

Figure 1 shows how frequently respondents use chat to communicate. Seventy-seven percent of respondents indicated they use chat at least once a week, with the majority of them using it multiple times a week. These frequent users predominantly used Google Talk, Skype, Facebook Chat, and other commercial products.

B. Trust

In this portion of the survey we asked questions to gauge how safe users feel sending information through a chat system. We then compared those answers with what types of information respondents send via chat. We provided the users with a list of sensitivities ranging from non-sensitive to highly sensitive and asked them to select the options that describe the kinds of information they have ever sent via chat in both a personal and business setting. Fifty-nine percent (Group 1) of respondents reported they feel safe or very safe when using chat, 24% (Group 2) admitted they have never thought about how safe they felt and were uncertain, and 17% (Group 3) reported they feel unsafe when they chat. Table II breaks down what percentage of each group have sent what type of information.

TABLE II
TYPE OF INFORMATION SENT OVER CHAT

Group	Type of information	Percent
Group 1 (59%)	Non-sensitive personal info	84%
	Moderately sensitive personal info	44%
	Highly sensitive personal info	15%
	Non-sensitive business info	56%
	Moderately sensitive business info	24%
Group 2 (24%)	Non-sensitive personal info	93%
	Moderately sensitive personal info	36%
	Any business info	<15%
Group 3 (17%)	Non-sensitive personal info	90%
	Moderately sensitive personal info	40%
	Non-sensitive business info	50%
	Moderately sensitive business info	30%

Of those in Group 1, 15% say they send highly sensitive personal information (e.g., social security number, credit card, bank information). Those who send highly sensitive personal information report they use Google Talk, Skype, and Facebook Chat to do so. This is of particular interest because Google Talk and Facebook Chat do not enforce secure connections to communicate via their chat systems. With Google Talk, secure connections are available when connecting through Gmail, Google+, or third party software, but that does not guarantee the other party is connected securely. Facebook defaults to unencrypted HTTP connections for most of their online services. There is a greater risk of an eavesdropper seeing sensitive information sent through the Google Talk or Facebook Chat networks.

Those in groups 2 and 3 reported to have never sent highly sensitive information of any kind over chat. It is interesting that even though those in group 3 indicated that they felt unsafe or very unsafe about the security of chat, most of them still reported using chat systems on a daily basis.

The feeling of safety that a user has is not only affected by the system they are using but also the context in which it is being used. We asked a set of questions to ascertain how users would react to communicating sensitive information to their trusted friends or family members.

When asked if they were more likely to send sensitive information to a trusted friend or family member (see Figure 2), 68% answered in the affirmative, 24% answered negatively, and 9% answered that it would depend on the kind of information. Most of the 9% who answered *Depends* identified information relevance and security of the communication medium as their deciding factors. However, one respondent was more concerned with the speed of the medium rather than the security of it.

When asked if they would reply over chat with sensitive information requested by a trusted friend of family member (see Figure 3), 41% answered that it would depend on the kind of information, 29% answered that they would use the phone instead, 12% answered yes, 10% answered they would use email, 3% answered yes if there was a need to know, and the remaining 5% responded no.

Fig. 2. More likely to send sensitive information to trusted friend or family member

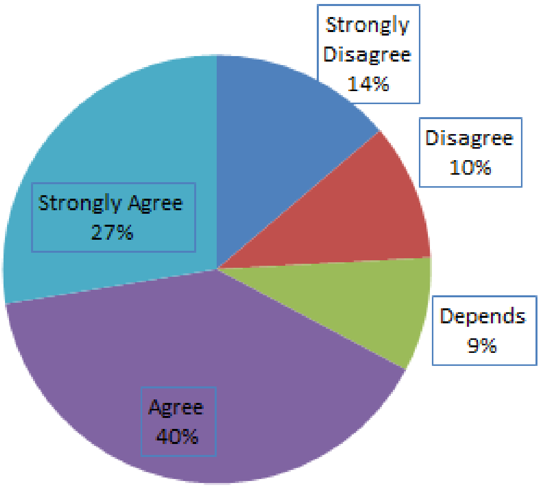
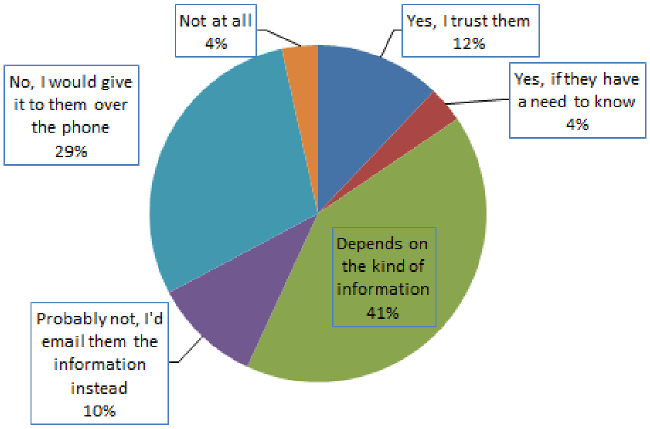


Fig. 3. Would reply with sensitive information to trusted friend or family member

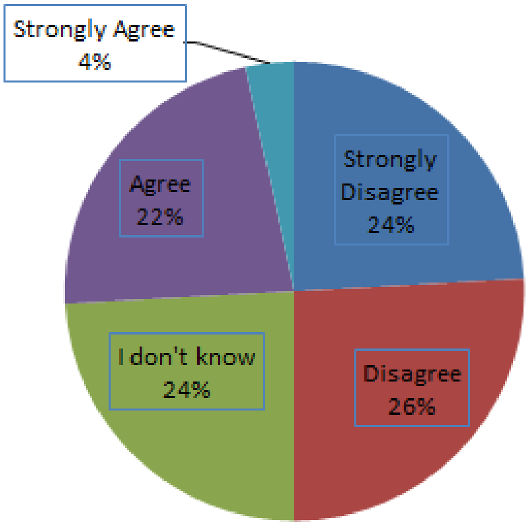


Users' responses in this part of the survey have been positive towards chat systems. Most feel safe chatting over their service of choice. This could be because their feeling of safety has never been challenged. It is apparent that users seem to have preconceived notions about which communication mediums are "safe" and which are not. The respondents mentioned email and phone systems as more secure alternatives to chat. However, email suffers from many of the same vulnerabilities as chat and the phone system has a long history of compromises [2].

C. Privacy

Some services, such as Google Talk, offer the ability for users to connect to the service using an HTTPS connection. HTTPS allows data to be transmitted in encrypted form, preventing eavesdroppers from reading the data. While this transport security is important, it does not offer full privacy because each communique is stored unencrypted and mined for data by Google. In addition, an HTTPS connection with Google does not guarantee that the other chat party is connected to Google with an HTTPS connection. We asked a set of questions to gauge the respondent's awareness of privacy and security issues.

Fig. 4. Question: I'm confident that my chat conversations are private



There was a lack of agreement among users when asked if they were confident that their chat conversations were private (see Figure 4). Approximately half disagreed or strongly disagreed that chat conversations were private, approximately a quarter did not know and the remaining quarter agreed or strongly agreed. These results suggest that many users lack an understanding of privacy. Those who agreed or strongly agreed were also in the group that indicated they felt safe or very safe when using a chat system. Additionally, 71% of those that answered *I don't know* indicated they felt safe or very safe when using a chat system. We further tried to assess respondents' inclination toward verifying the identity of the person with whom they are chatting beyond the facilities provided by the chat provider. Only 5% of the respondents did not feel confident in the identity of the opposite party. This suggests a possible entry point of attack where someone malicious could assume the identity of another trusted person.

We then asked two questions to ascertain the respondents' awareness and concern about what chat providers do with their conversations after having sent and stored them. Fifty-seven percent of respondents showed concern that chat providers may mine the text of their chat conversations to provide better targeted advertising. The remaining 43% were either indifferent or not concerned. Forty-nine percent of respondents showed concern that chat providers permanently store their messages while 51% were indifferent or not concerned.

The final question attempted to assess how responsive users might be if their standard way of chatting was found to be vulnerable. We specifically asked about the chat client. Fifty-five percent answered affirmatively that they would be inclined to move to another client, and 29% preferred not to move but rather that the client be fixed. The remaining 16% were indifferent.

III. Design and Implementation

A. Design Goals

The primary design goal is ease of use. Usability issues have been a barrier in previous secure communication systems and have prevented wide spread adoption [3], [4]. PFC extends Facebook Chat so that users can continue to use their existing chat system that they are already familiar with instead of switching to a new secure chat system. Users will read and compose messages with the added ability to designate when a chat session should be secure. The interface will clearly illustrate the difference between a standard chat session and a secure chat session. The low-level details of how the secure chat is implemented (e.g., key management, encryption algorithms) are handled transparently.

PFC is designed to be adopted in a grass roots fashion. The system spreads incrementally as users engage in secure chat sessions. In order to facilitate a simple installation procedure, the PFC client is implemented as a bookmarklet. A bookmarklet is a browser bookmark that runs JavaScript in the browser window rather than navigating to a webpage. Because the bookmarklet is just JavaScript, we can reach a greater audience because all major browsers support JavaScript. In addition, the user does not have to be an administrator to install a bookmarklet since it is just a bookmark. It is also easy and highly usable because the "install" is the same as adding any other bookmark or favorite in the browser.

The threat model that PFC is designed to address is to prevent an eavesdropper from accessing Facebook chat messages. PFC uses end-to-end encryption to prevent network eavesdroppers and Facebook itself from from reading chat messages. PFC uses a key escrow system to handle all key management so that users don't need manage their own keys. The key server uses Facebook's authentication mechanism to reliably hand out keys to the proper Facebook user. This approach means that users don't need another account password in order to use PFC.

B. Prototype Implementation

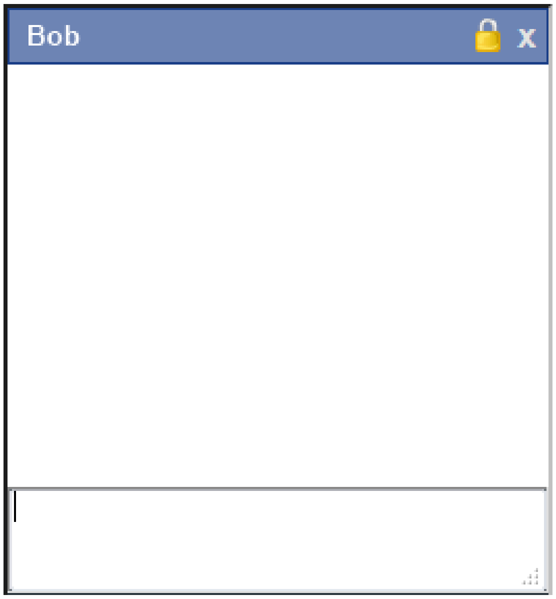
We created a prototype implementation of PFC. This section describes the significant innovations of the prototype and describes the client interface that users experience while using PFC.

1) Security Overlays: PFC uses security overlays. An overlay is a frame that rests directly on top of another part of the page. Its purpose is to hide parts of the original page to prevent the user from interacting with it. The user interacts with the overlay while the overlay interacts with the obscured parts of the original page on behalf of the user. An overlay provides security features that the original page does not. Since the content in the secure overlay is served from a domain that differs from the original page, the browser's same origin security policies prevents the original page from accessing content in the overlay. PFC overlays Facebook Chat windows with an overlay frame where users read and compose chat messages while the normal Facebook Chat window only sees encrypted content.

2) Usage Scenario: Suppose Alice wants to chat securely with Bob and already has the PFC bookmarklet installed in her browser. She opens a Facebook Chat dialog to Bob and clicks on the bookmarklet to enable the secure chat feature. This executes the Javascript associated with the bookmarklet and activates PFC for the duration of her Facebook session.

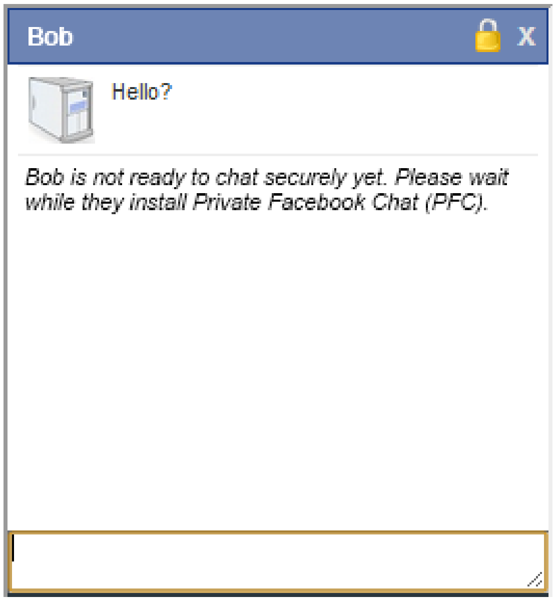
PFC overlays the Facebook Chat dialog with a security overlay and displays a lock icon to inform Alice that the chat session is secure (see Figure 5).

Fig. 5. Secure chat window with closed lock



Alice now waits for Bob to enter secure chat before she can send a message to Bob. If Alice tries to send a message before Bob is ready, the overlay system will inform her that Bob is not yet ready to receive secure messages (see Figure 6).

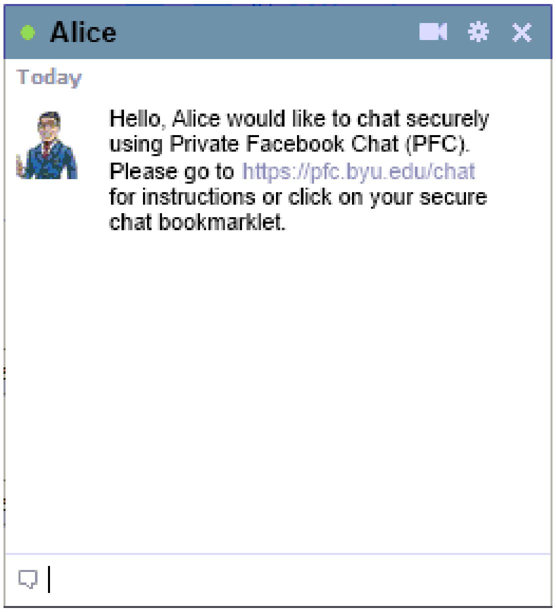
Fig. 6. Pending setup notification



Hello?

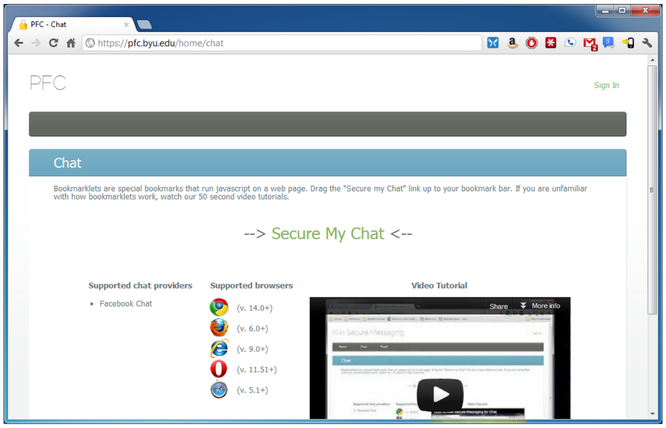
Bob is not ready to chat securely yet. Please wait while they install Private Facebook Chat (PFC).

Fig. 7. Standard greeting



Hello, Alice would like to chat securely using Private Facebook Chat (PFC). Please go to https://pfc.byu.edu/chat for instructions or click on your secure chat bookmarklet.

Fig. 8. PFC website



Bookmarklets are special bookmarks that run javascript on a web page. Drag the "Secure my Chat" link up to your bookmark bar. If you are unfamiliar with how bookmarklets work, watch our 50 second video tutorials.

--> Secure My Chat <--

Supported chat providers

- Facebook Chat

Supported browsers

(v. 14.0+)
(v. 6.0+)
(v. 9.0+)
(v. 11.51+)
(v. 5.1+)

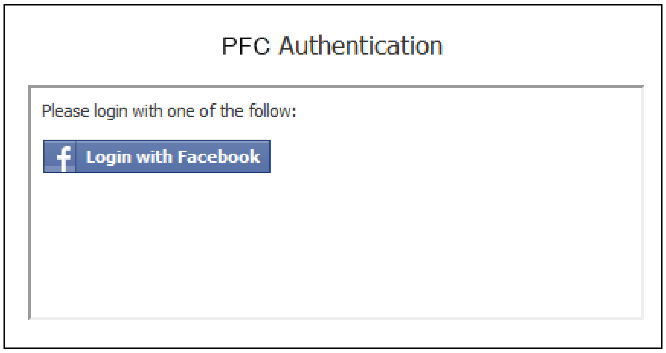
The PFC client on Alice's machine sends Bob a chat message from Alice stating that she would like to chat securely (see Figure 7).

The message directs Bob to a website containing instructions on how he can install the bookmarklet and chat securely with Alice (see Figure 8).

The website contains instructions and a short PFC video tutorial that directs Bob to install the secure chat bookmarklet by dragging it to his browser's bookmark bar. He then goes back to Facebook where the chat session with Alice is pending and clicks on the bookmarklet to secure the chat session in his browser. Alice and Bob are then both presented with the Facebook authentication dialog (Figure 9) that requests they authenticate with Facebook.

When they click on the *Login with Facebook* button, they are presented with a dialog from Facebook asking them if it is okay that the PFC Secure Messaging Facebook application access basic user information from their Facebook user profile (see Figure 10).

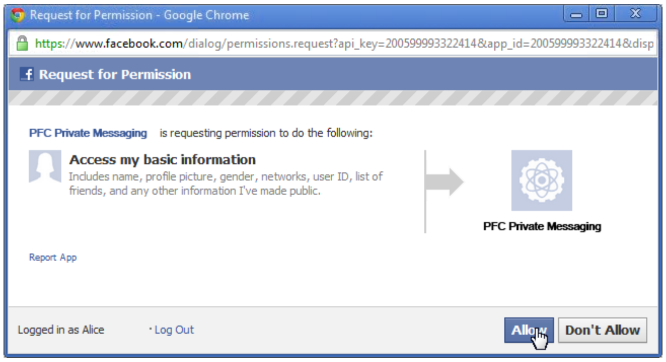
Fig. 9. Authentication window



PFC Authentication

Please login with one of the follow:

Fig. 10. Service provider authentication



Request for Permission - Google Chrome
https://www.facebook.com/dialog/permissions.request?api_key=200599993322414&app_id=200599993322414&disp
Request for Permission
PFC Private Messaging
Report App
Log Out
Allow
Don't Allow

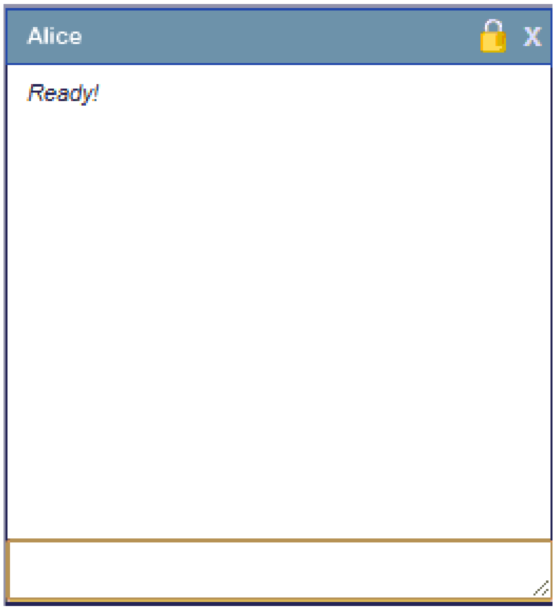
PFC Private Messaging is requesting permission to do the following:

Access my basic information

Includes name, profile picture, gender, networks, user ID, list of friends, and any other information I've made public.

Logged in as Alice

Fig. 11. Secure chat window after bookmarklet is used



Ready!

Upon choosing *Allow*, both Alice and Bob will receive a ready signal. Alice and Bob can now begin chatting securely via Facebook Chat (See Figure 11).

IV. Usability Study

A usability study of PFC was conducted involving 17 experienced Facebook Chat users (59% male, 41% female). The participants included BYU students and employees from a local software company. Eight participants (47%) were students and had no technical background. Of the nine participants from the local software company, three of them (33%) had no technical background.

The study consisted of 5 tasks designed to exercise a specific system feature along with some survey questions. The users had no advance notice that the study would focus on the security of Facebook Chat. We created 4 dummy Facebook accounts for participants to use during the study to help segregate tasks and to pose no privacy risks to the users. Each account had a friend named Steve that the participants would chat with for testing purposes. The average completion for each user was approximately 25 minutes.

A. Task #1

The purpose of this task is to the determine the usability of the bootstrapping process — can a participant who receives a request to chat securely successfully obtain the software and launch a secure chat session. Each participant was instructed to log in to Facebook and wait for Steve to contact them and send them some passwords. We purposefully avoided mentioning explicitly that this information should be exchanged securely because we wanted to measure the effectiveness of our bootstrapping prompts.

Each user received the PFC standard greeting (see Figure 7) inviting them to the PFC website to learn how to chat securely. All the users went to the website except one that said they never click on links unless they are sent by a friend they know extremely well.

Overall, users were successful in completing the task. 14 of the 17 of the participants (82%) found the standard greeting and the PFC website helpful in getting them started. In addition, these 14 participants found the bookmarklet very easy to install. The three participants who struggled with installing the bookmarklet provided feedback on how to improve the bootstrapping process. This feedback included how to order the instructions on the PFC website, areas where additional feedback can be given to the user, and portions of the process that could be further simplified. The overall success of this task shows that most users can bootstrap into the system with just a short greeting, provided the link in the standard greeting is trusted.

The PFC website was generally successful in helping the participants learn to install and use the bookmarklet. We found that participants' attention was drawn to the bookmarklet link before they read the short instructions or watched the video tutorial. Slightly less than half of the participants tried installing or using the bookmarklet without instruction. 6 participants (35%) attempted to directly click the link to enable secure chat. Others tried dragging the link into the address bar. The term *bookmarklet* seemed to confuse a few participants. We observed them being unsure how to create the browser bookmark. Eventually these participants abandoned their efforts and proceeded to read the provided instructions and watch the video tutorial.

Since PFC uses Facebook authentication, we had to create a Facebook application. Users were prompted to allow the application access to their personal information. 9 participants (53%) were very wary of trusting the application. Some participants tried to continue the task without allowing that application. Most users reported that they normally do not allow any Facebook applications.

B. Task #2

For task 2, the users were instructed to securely send a checking account number to set up direct deposit (we provided a dummy account number) to Steve using Facebook Chat. The purpose of this task was to determine whether a user that had already installed the PFC bookmarklet would be able to easily initiate a new secure chat. 15 out of 17 participants (88%) completed the task and reported that they found the bookmarklet to be intuitive and easy to use.

Six participants (35%) said that clicking on the lock icon was an unnecessary step, that the bookmarklet should automatically default to a secure chat rather than require a two-click process. One participant reported they would never send a bank account number in chat, but would prefer to use the phone. Finally, 3 participants sent their checking numbers insecurely because they forgot the bookmarklet was there and needed reminding. This highlights a weakness of the bookmarklet approach since bookmarklets are unable to execute without direct interaction from the user. A browser plug-in approach would be able to better intervene and remind users of the need to use secure chat.

C. Task #3

The goal of this task is to get the reaction of the participants when they see cipher text in their chat histories and measure the usability of resuming a secure chat session. The task asks the user to re-assume the identity in task #1 and securely chat with Steve again. By resuming a conversation, the chat history in Facebook will contain the cipher text of the previous conversation. We posit that the cipher text might be a point of confusion for participants unfamiliar with it.

5 participants (29%) did not notice the cipher text during the task. Most of these participants did not complete the previous tasks that were needed for cipher text to appear in this task. Another 5 participants (29%) said that they understood the significance of the cipher text and it did not bother them. The remaining 7 participants (42%) who saw the cipher text had mixed reactions. One of the side effects of sending text that contains a valid URL is that Facebook converts them into links so that they are clickable. The ciphertext package contains a URL to the key server that provided keys to encrypt the message. Because of the link in the standard greeting, some participants, when they saw this new link in the cipher text, assumed they were supposed to click on it, which took them to an error page.

Other participants tried clicking on the link in the standard greeting again. Some participants reported that they were confused when they saw it or thought that the secure chat system was broken. Overall, 14 participants (82%) eventually used the bookmarklet and successfully completed the task. These participants agreed that resuming a secure chat conversation was easy. More participants reported this task as easier than the previous task. We hypothesize that this is because the process of resuming a secure chat only requires a single click of the bookmarklet whereas the previous task also required the participants to click the lock icon.

D. Task #4

The goal of this task is to highlight the feature that helps a user maintain a secure chat session with the opposite party. The task instructs the participant to have a secure conversation with Steve, but that he is having problems with his computer. The instructions warn that Steve might jump in and out of secure communication. The participant is supposed to do all that they can to maintain a secure communication with Steve.

During this task, 15 of 17 participants (88%) agreed that it was easy to distinguish the difference between secure and insecure messages. These participants successfully completed the task. When interacting with the option to invite the opposite party to continue secure chat, some participants were expecting a *Ready!* notification when the opposite party rejoined. Overall, the participants were able to easily maintain a secure chat session with the other party.

E. Task #5

The goal of this task is to have participants switch in and out of a secure chat session during a conversation. The task instructs the participant to conduct a casual, insecure conversation with Steve. At some point during the conversation the participant must enter a secure chat session to send a bank account number. After sending that number, the participant must exit secure chat and finish the casual conversation they were having before. At this point in the study, the participants should be familiar with all aspects of PFC.

One of the features of the secure chat history is that messages are displayed as users would have received them. For example, if an insecure conversation is transitioned to a secure conversation, all previous insecure messages sent from the opposite party will be displayed as though they were received insecurely in secure mode. Eight participants (47%) clicked on the action items presented in the insecure messages. These past messages seemed to cause a moment's confusion. However, as soon as the other party started chatting securely, all participants ignored the previous messages and continued with the task.

Fifteen participants (88%) reported that distinguishing the difference between secure and non-secure mode was easy. These participants referenced the lock icon as their primary means of demarcation. Fifteen participants (88%) also said that using the lock icon to transition between secure and non-secure mode was easy and intuitive. Six participants (35%) commented that clicking the lock icon was an unnecessary step to enter secure mode because by clicking on the bookmarklet, they are already signaling their intention of chatting securely. Nine participants (53%) reported that once in secure chat they would continue in secure chat rather than leaving it, even if the topic of conversation becomes non-sensitive.

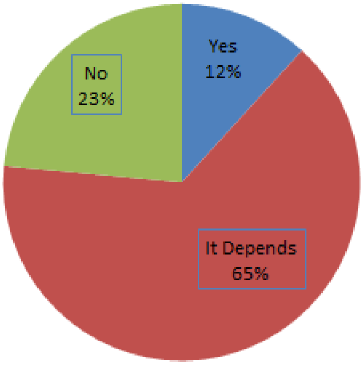
F. Survey

Following completion of the tasks, the participants answered some follow-up questions including several questions from the initial user survey. We also include questions that help us assess what the participants learned from the usability study. We try to assess the participants' inclination toward using PFC and their understanding of why using it is important.

Twelve participants (71%) said that after using PFC they were less inclined to send sensitive information over normal Facebook chat. The remaining 5 participants (29%) reported that they already do not send sensitive information through chat. Nine participants (53%) stated that they would be more inclined to send sensitive information via chat if PFC were available. Most of the other participants wanted more information about the underlying system before they would make a commitment to use it.

The success of the bootstrapping mechanism depends on users following the link sent in the standard greeting. We asked the participants if they trust links sent in chat, and Figure 12 shows their responses.

Fig. 12. Question: Do you trust links sent to you over chat?



Those who answered *It Depends*, stated that their trust of the link was dependent on their trust of the party who sent it. If trust of the standard greeting link is an issue for most users, the alternatives would be to change the standard greeting or rely on users to introduce the system to their contacts by giving them the link out-of-band.

Thirteen participants (76%) agreed they would be likely to start using this system with friends, family, or acquaintances if it were available. The remaining 4 participants (24%) were undecided. A previous task has shown that some users forgot the bookmarklet was present. This lack of decision could be due to the lack of need or reminders to use this kind of secure system. Most participants reported that if they did use this system in the real world, they would, at the very least, enable it for sensitive topics of conversation. Eight of the participants (47%) said their awareness about the security of chat systems changed as a result of their participation in this study.

G. Lessons Learned

Overall, PFC is usable. Everyone except two extremely novice users were able to successfully engage in secure chat sessions without training or human assistance. Two older, non-technical participants failed to complete any task. They use Facebook and Facebook Chat solely to stay in touch with family and close friends. They are only comfortable learning new technology with human guidance, so the current system was unable to meet their needs.

Users may be reluctant to bootstrap into the system by following the link contained in the standard greeting. In practice, this means a user wanting to initiate a secure chat session with someone unfamiliar with PFC may first have to chat insecurely and introduce them to the process.

The user study uncovered ways to improve the PFC bootstrapping web page to be more informative and easier to follow. During the study we observed that many participants were drawn immediately to the bookmarklet link before watching the video tutorial. As a result, many assumed they were supposed to click on the link, whereas if they had first watched the tutorial, they would have created a bookmark instead. Some users ignore the video and attempt to use the system immediately. A short list of steps included on the web page that summarizes the video content will be helpful. For instance, this information can alert the users that they will need to trust the associated Facebook application in order to use PFC.

Participants quickly picked up the bookmarklet during the course of the study and recognized a way to streamline the process by having the one-click of the bookmarklet automatically opt the user into a secure chat session. Finally, the user study illustrated that the chat history is confusing when it contains links. Some participants assumed they were supposed to click on them. This can be improved by presenting the actions items in a dialog instead so that the actions do not display in the chat history.

V. Threat Analysis

This section contains a threat analysis of the PFC prototype. First we list the passive attacks against the system, and then give several active attacks that can be attempted against PFC.

A. Passive Attacks

PFC thwarts passive eavesdropping by Facebook. It also thwarts other eavesdroppers on the network. PFC chat sessions are encrypted end-to-end, and protection from eavesdropping is effective even if some of the transmission links are not protected with HTTPS.

B. Key Compromise

One way to attack the system is to compromise the encryption keys. This could be done by attacking the key escrow server directly or by stealing encryption keys from the users. Attackers could attempt to steal secret keys or key material from the key server. This attack can be mitigated by storing these values in secure hardware. Alternatively, the attacker could attempt to compromise the key server to hand out encryption keys online, but this is much easier to detect and shut down. Another way to mitigate the risk of stolen encryption keys is to limit the lifetime of a key. Our implementation derives keys that are only valid for the month in which they were derived.

An important factor in the system design is that compromising the key server or a specific user's keys is not enough for an attacker to recover plaintext messages. If an attacker is unable to eavesdrop on encrypted content sent across a network, the attacker must compromise the Facebook accounts of either the sender or receiver to acquire the encrypted content. This second layer of defense helps to mitigate the threat of key compromise.

C. Impersonation

PFC relies on Facebook's OAuth system to authenticate a user and provide a unique Facebook identifier for use in deriving encryption keys. This means Facebook can impersonate a user to the key server to obtain that user's encryption keys. Users trust Facebook not to impersonate them, an attack that Facebook could already launch in it's Facebook Connect system. If Facebook were to launch such an attack in PFC, they would risk detection and a loss of reputation. The risk of this attack can be mitigated by adopting a multi-factor authentication mechanism at the key server, such as adding a user-specific password in addition to Facebook's authentication. This would prevent Facebook from easily impersonating users, at the expense of making the system less useable. Our initial PFC design favored usability over strong authentication. These properties can be adjusted to tailor the system to specific user's needs.

D. Social Engineering

During the user study, users were hesitant to click on the link provided in the standard greeting. However, once they accepted that the study required them to click on links, they assumed any link presented by PFC was trustworthy. In practice, an attacker can exploit users who trust PFC messages. For instance, a malicious service provider or an active attacker on an insecure link could inject a malicious link into the standard greeting (or any other PFC message) in order to exploit an unsuspecting user that already trusts PFC messages.

The system is designed so that users who already have a trust relationship in Facebook can chat privately. We don't make any assumptions about user's likelihood to click on PFC links because research has shown that participants in user studies may be more likely to trust experimental software in a user study compared to actual use [5].

VI. Related Work

Mannan and van Oorschot [6] surveyed the security features and threats to instant messaging protocols in an effort to spark future security improvements. They observe that the greatest threat is insecure connections, and almost a decade later this is still the case because Facebook uses HTTP by default. This threat was a primary motivation for developing PFC.

Jennings et al. [7] discuss three popular chat protocols from 2006: AOL Instant Messenger, Yahoo! Messenger, and Microsoft Messenger. Although these systems provide modest improvements beyond plaintext passwords, their security features are limited and make no effort to provide confidentiality.

Significant research has been focused on instant messaging protocols with advanced security features. Off-the-Record Messaging (OTR) allows private conversations over instant messaging by providing encryption, authentication, deniability, and perfect forward secrecy. Borisov et al. [8] introduces an Off-the-Record Messaging protocol for secure instant messaging. The protocol uses the Diffie-Hellman key exchange protocol to establish short-term keys that are impossible to rederive from the long-term key material. These keys are then discarded after a period of use, making any past messages permanently unrecoverable. The messages in this protocol are not digitally signed. It is thus impossible to prove who sent a message. Because of the frequent key exchanges necessary for secure communication, it is vulnerable to replay attacks that allow an attacker to impersonate the sender to any other party in the system. Now that PFC has proven its usability, researchers can explore ways to incorporate stronger security properties while still maintaining usability.

Raimondo et al. [9] analyzes the key features presented in Borisov et al. [8] and examines security vulnerabilities. They propose a series of change recommendations for Off-the-Record Messaging in an attempt to fix the vulnerabilities. Their recommendations include replacing the authenticated key exchange protocol with stronger exchange protocols.

Other research in Goldberg [10] and Jiang [11] addresses Off-the-Record group conversations, such as public chat rooms or other multi-party scenarios. Alexander [12] applies the socialist millionaire's problem to OTR to improve user authentication. OTR has also become publicly available as a Pidgin plug-in. Stedman et al. [13] conducted a usability study of this Pidgin plug-in to determine if it is easy to use and successful at hiding computer security details from the user. They discuss flaws in the user interface that cause confusion and decreased security. They also discuss possible solutions to these errors.

Kikuchi et al. [14] present a secure chat protocol that extends the Diffie-Hellman key exchange in order to thwart a malicious administrator. PFC could also be implemented using a Diffie-Hellman key exchange. We chose a key server approach instead to prevent an active man-in-the-middle attack on the basic Diffie-Hellman protocol.

Mannan and van Oorschot [15] present the Instant Message Key Exchange (IMKE) protocol, which is a password authentication and key exchange protocol. IMKE allows for strong authentication and secure communication in IM systems. It provides authentication (via memorable passwords), confidentiality, and message integrity with repudiation. IMKE cannot be layered on top of existing IM systems without modifications to both client and server technologies. We designed PFC so that it could be deployed without any server involvement in order to enhance the security of a popular chat platform.

In 2005, Google launched its Google Talk platform. Google Talk has an OTR mode that promises to not store the contents of an IM session. Users must trust Google to follow through on this promise since there is no cryptographic assurance that the policy is enforced. PFC could be adopted to work with Google Talk. An OTR mode could be supported by having PFC turn on the OTR flag in Google Talk so that the encrypted PFC messages are not retained by Google.

VII. Conclusions

We conducted a survey of 65 users to assess their awareness and attitudes concerning the privacy of instant messaging. 59% of the users surveyed feel safe or very safe while communicating via instant messaging, and 15% of that group admit to sending highly sensitive information in a chat session. Users are more likely to share sensitive information with a close friend or family member. Some users trust email or the phone more than they trust chat. A number of users are wary of chat service providers; 57% are concerned that providers may scan their messages for directed advertising purposes, and 49% have concerns with their messages being stored permanently.

This paper presents PFC (Private Facebook Chat), a system that provides end-to-end encryption for Facebook Chat sessions so that eavesdroppers (including Facebook itself) cannot access chat messages. The system is designed with good-enough security to thwart eavesdroppers. Security overlays provide a distinct interface on top of the existing Facebook interface so that the plaintext of a chat conversation is not available to Facebook or anyone who could modify a Facebook page during transmission. The primary design focus is on making the system easy to use.

We report on a user study that demonstrates the system is usable by current Facebook users except for the most novice computer user. The user study revealed several issues with the system. Including a link in the bootstrapping messages in order to install the system may be problematic because some users do not trust links in a chat message. To overcome this issue, the person initiating the secure chat may need to provide preliminary information that encourages the recipient being willing to click on the link. Users reported a increased awareness of the privacy issues of using Facebook Chat as a result of participating in the user study.

Our future work includes exploring ways to support secure messaging in other areas of Facebook where the communication is asynchronous, such as wall postings and status updates. We would also like to explore ways to make the ciphertext less intrusive when displayed to unauthorized parties. For instance, can we adapt steganography so that the encrypted message is stored in an image that is visible to unauthorized users, while the actual message is displayed to those authorized to see it. We also plan to propose a standardized API that service providers could support to make it easier for third parties to offer security services based on security overlays.

Acknowledgment

The authors would like to thank Ben Burgon, Trevor Florence, Kimball Germane, Scott Robertson, and Ryan Segeberg for their work in the Internet Security Research Lab (ISRL) at BYU to develop ideas and software related to this work.

References

[1] Wikipedia, “Firesheep,” 2011, [Online; accessed 18-May-2012]. [Online]. Available: http://en.wikipedia.org/wiki/Firesheep

[2] ——, “Phreaking,” 2011, [Online; accessed 12-Nov-2011]. [Online]. Available: http://en.wikipedia.org/wiki/Phreaking

[3] S. Sheng, L. Broderick, J. Hyland, and C. Koranda, “Why johnny still can’t encrypt: Evaluating the usability of email encryption software,” in *Symposium On Usable Privacy and Security*, 2006.

[4] A. Whitten and J. Tygar, “Why johnny can’t encrypt: A usability evaluation of pgp 5.0,” in *Proceedings of the 8th USENIX Security Symposium*, vol. 99, 1999.

[5] A. Sotirakopoulos, K. Hawkey, and K. Beznosov, “I did it because i trusted you: Challenges with the study environment biasing participant behaviours,” in *SOUPS Usable Security Experiment Reports (USER) Workshop*, 2010.

[6] M. Mannan and P. C. V. Oorschot, “Secure public instant messaging: A survey,” in *Proceedings of the 2nd Annual Conference on Privacy, Security and Trust*, 2004, pp. 69–77.

[7] R. B. Jennings, E. M. Nahum, D. P. Olshefski, D. Saha, S. Zon-Yin, and C. Waters, “A study of internet instant messaging and chat protocols,” *Network, IEEE*, vol. 20, no. 4, pp. 16–21, 2006.

[8] N. Borisov, I. Goldberg, and E. Brewer, “Off-the-record communication, or, why not to use pgp,” in *Proceedings of the 2004 ACM Workshop on Privacy in the Electronic Society*, ser. WPES ’04. New York, NY, USA: ACM, 2004, pp. 77–84. [Online]. Available: http://doi.acm.org/10.1145/1029179.1029200

[9] M. Di Raimondo, R. Gennaro, and H. Krawczyk, “Secure off-the-record messaging,” in *Proceedings of the 2005 ACM Workshop on Privacy in the Electronic Society*, ser. WPES ’05. New York, NY, USA: ACM, 2005, pp. 81–89. [Online]. Available: http://doi.acm.org/10.1145/1102199.1102216

[10] I. Goldberg, B. Ustaoğlu, M. D. Van Gundy, and H. Chen, “Multi-party off-the-record messaging,” in *Proceedings of the 16th ACM Conference on Computer and Communications Security*, ser. CCS ’09. New York, NY, USA: ACM, 2009, pp. 358–368. [Online]. Available: http://doi.acm.org/10.1145/1653662.1653705

[11] B. Jiang, R. Seker, and U. Topaloglu, “Off-the-record instant messaging for group conversation,” in *IEEE International Conference on Information Reuse and Integration*, 2007, pp. 79–84.

[12] C. Alexander and I. Goldberg, “Improved user authentication in off-the-record messaging,” in *Proceedings of the 2007 ACM Workshop on Privacy in Electronic Society*, ser. WPES ’07. New York, NY, USA: ACM, 2007, pp. 41–47. [Online]. Available: http://doi.acm.org/10.1145/1314333.1314340

[13] R. Stedman, K. Yoshida, and I. Goldberg, “A user study of off-the-record messaging,” in *Symposium on Usable Privacy and Security*, 2008, pp. 95–104. [Online]. Available: http://doi.acm.org/10.1145/1408664.1408678

[14] H. Kikuchi, M. Tada, and S. Nakanishi, “Secure instant messaging protocol preserving confidentiality against administrator,” in *18th International Conference on Advanced Information Networking and Applications*, vol. 2, 2004, pp. 27–30.

[15] M. Mannan and P. van Oorschot, “A protocol for secure public instant messaging,” *Financial Cryptography and Data Security*, pp. 20–35, 2006.