All Our Messages Are Belong to Us: Usable Confidentiality in Social Networks

Marian Harbach, Sascha Fahl, Thomas Muders and Matthew Smith
Distributed Computing & Security Group
Leibniz University of Hannover
Hannover, Germany
{harbach,fahl,muders,smith}@dcsec.uni-hannover.de

ABSTRACT

Current online social networking (OSN) sites pose severe risks to their users' privacy. Facebook in particular is capturing more and more of a user's past activities, sometimes starting from the day of birth. Instead of transiently passing on information between friends, a user's data is stored persistently and therefore subject to the risk of undesired disclosure. Traditionally, a regular user of a social network has little awareness of her privacy needs in the Web or is not ready to invest a considerable effort in securing her online activities. Furthermore, the centralised nature of proprietary social networking platforms simply does not cater for end-to-end privacy protection mechanisms.

In this paper, we present a non-disruptive and lightweight integration of a confidentiality mechanism into OSNs. Additionally, direct integration of visual security indicators into the OSN UI raise the awareness for (un)protected content and thus their own privacy. We present a fully-working prototype for Facebook and an initial usability study, showing that, on average, untrained users can be ready to use the service in three minutes.

Categories and Subject Descriptors

H.3.5 [**Information Storage and Retrieval**]: Online Information Services—*Data Sharing*; E.3 [**Data**]: Data Encryption

General Terms

Security, Human Factors

Keywords

Confidentiality, Privacy, Social Networks, Usability

Introduction

Online Social Networks (OSNs) currently play an important role in many people's daily lives. The amount of social interaction taking place on the Internet is growing rapidly and benefits from new technology. However, this also poses a risk to a user's information in terms of privacy and trust. Studies have shown that users value their privacy in general, but act the opposite in social networks [1]. While the privacy implications of publicly sharing information is slowly finding its way into the users' minds, the problem of giving the OSN provider all their (possibly highly private) information without any means of control has not been properly recognised by the general public yet.

While the idea of cryptographically securing messages and other OSN content is no novelty, previous approaches often suffer from a lack of usability by impeding the user's regular workflow. Especially because privacy in OSNs is often outweighed by the perceived utility, users are not willing to invest considerable effort into additional measures. Therefore, we propose a novel and above all user-friendly approach to add confidentiality and integrity to private messaging in social networks. By utilising existing infrastructure, this approach will allow us to conserve the existing user experience and scale well to the sizes of current online social networks. We provide an easy-to-use prototype for Facebook that integrates seamlessly with the normal workflow and raises awareness for (un)protected messages.

Usable Confidentiality for OSNs

Catering for the users' habits, we derive the following usability requirement for a usable confidentiality and integrity mechanism in OSN messaging:

Usability Reqirement.
Confidentiality and integrity for OSN messages must be unobtrusively integrated into the users' common workflows. Both usage patterns as well as user interfaces must be supported smoothly, causing minimal extra effort.

To achieve the above requirement, we integrate a lightweight privacy service, operated by a third party. The service does not store any information, but cryptographically protects a user's messages without ever learning anything about their content. We use AES as a symmetric encryption algorithm to protect the users' messages from eavesdroppers. Furthermore, all the user has to do in order to enrol in this service is to create another user account in addition to the one that was created with Facebook. Authentication for the service is lightweight and based on email-based identification and authentication using a regular password. This is a standard procedure and proved to be easily accepted during our user study (cf. below).

After a successful binding of the Facebook identity, we use a Greasemonkey[1] script to be able to encrypt and decrypt messages within the regular Facebook UI. Greasemonkey is a Firefox extension for client-side, third-party JavaScript ex-

[1] `http://www.greasespot.net/`

WWW 2012 Companion, April 16–20, 2012, Lyon, France.
ACM 978-1-4503-1230-1/12/04.

tensions, that can manipulate a displayed website while protecting information in a sandboxed environment.

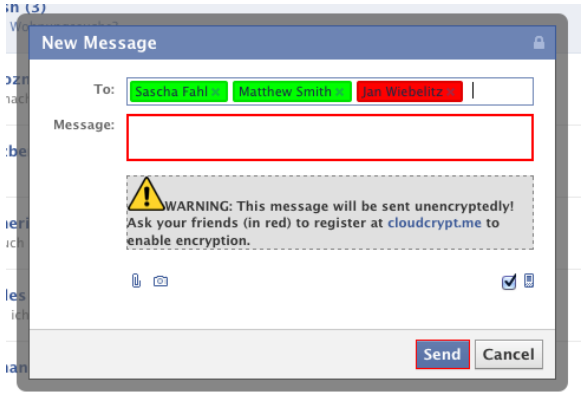

Figure 1: The modified Facebook message composer.

The client-side script currently supports the sending and receiving of encrypted messages through the regular Facebook user interface without requiring any understanding of cryptographic artefacts at the end-user's side. Figures 1 and 2 show the modified Facebook UI for sending and receiving messages.

Figure 2: A comparison of the message with and without decryption.

To help users to intuitively understand that message protection is in place, protected content is annotated with visual security indicators. Studies found that the timing and placement of privacy indicators play a central role in the user's perception and need to be close to the elements of interest (e. g. [2]).

Thus, in our prototype, a red border marks a piece of information that is potentially in need of protection or is unprotected, while a green border intuitively indicates successful protection. We believe that the visual indicators will also raise the user's awareness for private information in need of protection, and will therefore eventually increase the user's privacy perception and actual privacy.

Evaluation

We conducted a user study with 20 undergrad students and found that registering for the service and binding a Facebook account took 3:08 minutes on average (ranging from 90 seconds to 6 minutes). Additionally, we collected performance measurements to estimate the delay a user experiences when using enhanced social networks. On average, it took between 33ms and 154ms to encrypt or decrypt a message of 2 to 8000 characters (cf. Fig. 3), including transport. The tests were run on a 2.66 GHz Core2 Duo machine with 8 GB of RAM against a 3 GHz Pentium D dual core server having 4GB of RAM.

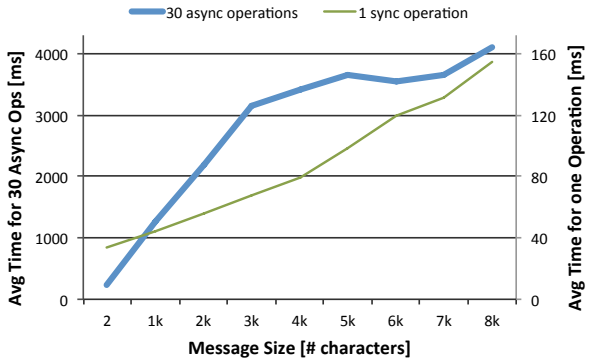

Figure 3: Response time for one crypto-operation (thin line) and 30 asynchronous decode operations (bold line) on variable message length.

Using the proposed mechanism, encrypting a message upon sending or decrypting upon reception is only barely noticeable by the user. Asynchronously decrypting the entire message history (typically the past 30 messages) takes on average as long as it takes to load the page (between 222ms and 4101ms depending on the message size) and can begin while the rest of the page is still loading. We therefore believe that our confidentiality plugin does not disturb the normal Facebook experience.

Conclusion

In this paper, we presented an approach for user-friendly, non-obtrusive confidential messaging in online social networks. We enhance the existing and intuitive messaging workflow and UI of OSNs, which allows users to gain an awareness of privacy as well as an ability to directly see which content is protected and which isn't. A user study showed that the enrolment procedure for our service is quick and easy and requires no specialised knowledge. Our Facebook plugin demonstrates that privacy mechanisms can be transparently integrated into existing user interfaces without interruption of the user's regular workflow.

REFERENCES

[1] D. Boyd. *Taken Out of Context: American Teen Sociality in Networked Publics*. PhD thesis, University of California-Berkeley, School of Information, 2008.

[2] S. Egelman, J. Tsai, L. F. Cranor, and A. Acquisti. Timing is Everything?: The Effects of Timing and Placement of Online Privacy Indicators. In *Proceedings of the 27th International Conference on Human Factors in Computing Systems*, pages 319–328. ACM, 2009.