When SIGNAL hits the Fan: On the Usability and Security of State-of-the-Art Secure Mobile Messaging

Svenja Schröder
University of Vienna
Email: svenja.schroeder@univie.ac.at

Markus Huber, David Wind, Christoph Rottermanner
St. Pölten University of Applied Sciences
Email: {markus.huber, is121030, is121023}@fhstp.ac.at

***Abstract*—In this paper we analyze the security and usability of the state-of-the-art secure mobile messenger SIGNAL. In the first part of this paper we discuss the threat model current secure mobile messengers face. In the following, we conduct a user study to examine the usability of SIGNAL's security features. Specifically, our study assesses if users are able to detect and deter man-in-the-middle attacks on the SIGNAL protocol. Our results show that the majority of users failed to correctly compare keys with their conversation partner for verification purposes due to usability problems and incomplete mental models. Hence users are very likely to fall for attacks on the essential infrastructure of today's secure messaging apps: the central services to exchange cryptographic keys. We expect that our findings foster research into the unique usability and security challenges of state-of-the-art secure mobile messengers and thus ultimately result in strong protection measures for the average user.**

I. INTRODUCTION

Tools to securely communicate over the Internet, using end-to-end (e2e) encryption, have been available for decades. End-to-end encryption ensures that sensitive encryption keys never leave users' devices, and communication providers are therefore unable to read exchanged messages. The first generation of end-to-end encryption tools, such as PGP, however lacks widespread adoption due to their bad usability [1], [2], [3], [4]. Since the first release of PGP three decades ago, two important aspects of secure messaging changed: everyday communication via mobile devices continued to grow as smartphones replace PCs [5] and the general awareness for privacy and security increased.

The trend of communication via mobile devices and the growing awareness for online privacy led to a number of new secure mobile messengers. The Electronic Frontier Foundation (EFF) provides an overview on the security properties of current mobile messengers [6]. From a security perspective, state-of-the-art mobile messengers can be split into two categories: messengers that provide client to server encryption and messengers with end-to-end encryption. The first category of messengers allows service providers to read exchanged messages, while the second group ensures that messages can not be read by service providers. State-of-the-art end-to-end encrypted mobile messengers only require users to authenticate via their mobile number; the generation and exchange of cryptographic keys is handled transparently by the applications. The transparent end-to-end encryption of messages makes strong encryption accessible to the masses but also creates new security challenges. As compared to PGP, state-of-the-art secure mobile messenger applications rely on centralized services to provide the cryptographic identities of its users. This modus operandi results in the following security challenge: if the key-exchange service is tampered with, either willingly or by an attacker, the overall security of systems is subverted. In order to account for the compromise of the key-exchange service, mobile messaging apps therefore offer the possibility to verify the cryptographic identities of other users ultimately to establish the trust of exchanged encryption keys.

To the best of our knowledge we are the first to study the unique usability challenges of mobile end-to-end encrypted messengers. Specifically, we perform a user study on SIGNAL for Android [7]. SIGNAL originated from two separate mobile applications [8] — TextSecure (encrypted instant messaging) and RedPhone (encrypted phone calls). Due to its strong encryption protocols and the availability of its source code under an open source license, SIGNAL has become an important tool for users who face surveillance [9]. In April 2016, the currently most popular messenger app WHATSAPP [10], rolled out end-to-end encrypted messaging, based on SIGNAL's protocol, to more than one billion users [11]. SIGNAL's encryption protocol thus became the de facto standard for end-to-end encrypted mobile messaging. In this paper we present a usability study of the messaging app SIGNAL including an exploration of the users' abilities to notice, handle and mitigate man-in-the-middle (MITM) attacks during usage. Our MITM attack simulates a compromised key-exchange service to ultimately evaluate the usability of SIGNAL regarding the detection and mitigation of such attacks. Our paper makes the following main contributions:

- We performed a user study with 28 participants on the usability of SIGNAL's security features, the state-of-the-art application for secure mobile messaging.
- Our results showed that 21 of 28 participants failed to

EuroUSEC '16, 18 July 2016, Darmstadt, Germany

http://dx.doi.org/10.14722/eurousec.2016.23012

compare encryption keys to verify the identity of other users. The majority of these users however believed they succeeded while in reality they failed.

- Finally, we suggest improvements for the usability of SIGNAL to better counter attacks on SIGNAL.

II. BACKGROUND

SIGNAL offers forward secrecy at the same time as asynchronous message exchange. As such SIGNAL combines the PGP-like asynchronous messaging with the security properties of the OTR protocol [12]. Figure 1 shows a simplified description of the SIGNAL protocol, which is divided into three phases (registration, session setup, and message exchange). We point the interested reader to Frosch et al. [13] for a detailed analysis of SIGNAL's protocol.

Alice and *Bob* want to use SIGNAL to exchange end-to-end encrypted messages. ❶ Alice installs SIGNAL and verifies her mobile number at the SIGNAL Server with either a verification text message (SMS) or a voice call. Once verified, Alice creates different sets of keys: a longtime asymmetric key-pair called `Identity Key Pair`, 100 ephemeral key pairs called `One-Time Pre Keys` as well as one `Signed Pre Key` which is signed with the `Identity Key`. SIGNAL automatically uploads Alice's `Signed Pre Key` as well as the 100 `One-Time Pre Keys` to its server. ❷ Alice attempts to establish a session with Bob and therefore requests a `Pre Key Bundle` for Bob and Bob's `Identity Key` from SIGNAL's central service. The `Pre Key Bundle` consists of a single public `One-Time Pre Key` and the `Signed Pre Key` of Bob. Based on the `One-Time Pre Key` and the `Signed Pre Key`, Alice derives a symmetric `Master Key` for future communication, and stores Bob's `Identity Key`. ❸ Based on the `Pre Key Bundles` of each other, both Alice and Bob derive the same `Master Key`, which is used to create ephemeral `Message Keys` for the actual message exchange.

The unique long-term Identity Key pair remains the same as long as the user does not delete it by for example re-installing the SIGNAL application. These Identity Keys are essential to verify the identity of communication partners. The SIGNAL application therefore stores the Identity Keys of other users as soon as a secure session has been successfully established. SIGNAL allows users to view this Identity Key within the application. In order to make sure that communicating parties received the correct Identity Keys, both parties have to verify the public Identity Keys via an out-of-bound channel (e.g. meet in person or via phone). This can be done by comparing the hexadecimal representation of the key byte per byte or by scanning the QR code of each other's Identity Keys in person.

A. Threat Model

Our threat model accounts for the compromise of SIGNAL's central services. This compromise can be the result of targeted attacks on SIGNAL's service infrastructure or assistance of SIGNAL's team to a subpoena request. The compromise of SIGNAL's key server results in two different possible attacks:

❶ **Attacks on the first session setup** do not result in direct user feedback. This attack can only be detected by **manually verifying** e.g. over the phone or face-to-face via scanning the QR codes. Consider Bob wants to initialize a secure session with Alice, and Bob receives the attacker's Identity Key (Mallory's Identity Key) instead of Alice's Identity Key which is then stored by SIGNAL as Alice's identity.

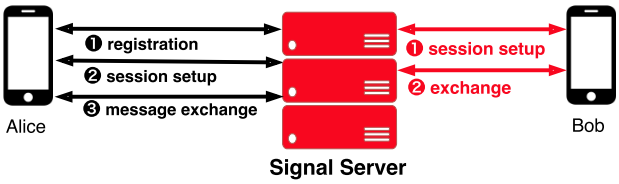

Fig. 1. Exchange of encrypted message with SIGNAL: a central service is used to exchange the public encryption keys — this service is critical for SIGNAL's security.

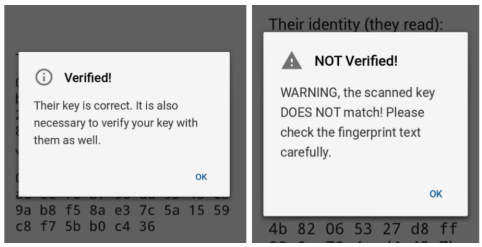

Fig. 2. Verification of Identity Keys by scanning the each other's QR codes. On the left: a successful verification. On the right: Warning because Identity Keys did not match.

❷ **Attacks on established sessions** where Bob has previously established a secure session with Alice and stored Alice's correct Identity Key. An attacker (Mallory) could force both parties to re-negotiate a new communication session. In this scenario the compromised SIGNAL server would respond with the attacker's Pre Key Bundle including the Signed Pre Key of the attacker, and thus establishes a man-in-the-middle attack.

SIGNAL accounts for both of the attack scenarios of our threat model. First, SIGNAL provides a feature to manually verify established Identity Keys, outlined in Figure 2. Second, SIGNAL warns users when it detects that long-term keys of users change, see Figure 3. In our paper we study exactly how usable and effective these two countermeasures of SIGNAL are.

III. EXPERIMENTAL DESIGN

We conducted a user study in a laboratory setting in order to explore the usability of SIGNAL regarding its security features. Our study consisted of two parts: a usability study of the SIGNAL app with focus on SIGNAL's instant messaging and security features, and the execution of an actual MITM attack with a subsequent assessment of the users' reactions. To gain insights into the participants' motivations, strategies and goals they were asked to constantly comment aloud on their actions with the Think Aloud method [14], which facilitated to understand the users' mental models. User interaction and voice were recorded with a camcorder. Participants had to fill out a consent form before the start of the study, as well as a short questionnaire including demographics and general attitude towards privacy and security regarding smartphones and especially messaging apps. The study took place in the usability lab of the COSY Research Group at the University

of Vienna, which provides two lab rooms for usability experiments and an operator room. Two tests were conducted in parallel, thus four operators (two in the operator room and two in the respective test rooms) had to be present to conduct the study in parallel.

A. Study Design

At the beginning of the study, participants received a set of instructions including all tasks and questionnaires, as well as an Android device with SIGNAL pre-installed. Each phone (Alice) had a contact entry for the conversation partner (Bob), handled by an operator. The detailed technical set-up is described in the next subsection. In the following we describe the tasks participants had to complete as part of our study. The **first part** of the study focused on SIGNAL's general usability related to messaging and security features. In the first task users had to participate in a brief chat conversation with Bob. Bob was simulated by an operator in the operator room. In a second task, participants had to create a password and subsequently export and import a backup of their messages from the first task. With this task we aimed at covering another basic security feature of SIGNAL. In-between the two study parts the MITM attack was initiated by the operator. In the **second part**, participants again had to exchange a few more messages with Bob. Due to the MITM attack of our simulated compromised SIGNAL server, this triggered an error message about Bob's mismatching key (see Figure 3). The task description also asked users to verify Bob's identity, after the message exchange. Our instructions informed participants that they could ask their chat partner Bob into the room at any time. Bob (simulated by an operator) was instructed to play a completely passive role and not to reveal any information on the verification task. Following the verification task, the participants had to fill-in a debriefing questionnaire aimed at assessing the users' mental model of the MITM attack, as well as possible mitigation strategies, by using quantitative and qualitative questions.

B. Technical Set-Up

In order to conduct our study with two persons in parallel, two identical setups were used which were each administered by one operator. One working setup consists of three smartphones and one computer which was responsible for intercepting the traffic and for creating a WLAN hotspot for the smartphone's internet connectivity. All smartphones were rooted and had Cydia Substrate [15] and SSLTrustKiller [16] installed in order to eliminate the SSL certificate pinning protection of SIGNAL. For traffic interception and manipulation we used mitmproxy [17] in combination with a custom script to automatically intercept SIGNAL messages. Two client smartphones (Android 4.4.4) and one attacker smartphone (Android 4.4.4) were used. The attacker smartphone (Mallory) was preloaded with a modified version of SIGNAL to handle intercepted messages and to forward intercepted messages to the original recipient. The two client smartphones had the latest version of SIGNAL installed (3.15.2). One client smartphone was given to the study participant (Alice), the other client smartphone was used by the operator (Bob) in the operator room. Finally, because all smartphones shared the same network, the smartphones connected to our attack proxy via a ProxyDroid [18] configuration. For each study participant the devices were reset and re-registered with SIGNAL.

C. Pilot Study

We conducted a pilot study with six participants from the authors' research groups to refine our study design before the actual study. In our pilot study we asked users to "verify" their communication partner. This request led to confusion as our participants never reached SIGNAL's verification features and had widely diverging understandings of the term "verification". Thus no user successfully managed to compare keys. Based on our results of the pilot study we included a brief explanation of SIGNAL, to point participants towards SIGNAL's technical verification features. Furthermore, we decided to include a "hint": the instructions told the participants that they could ask for their communication partner (Bob) to enter the room at any time. Since participants of the pre-study were unsure whether Bob is a real person or a pre-scripted Bot, this information was crucial to include.

IV. RESULTS

A. Participants and general Usability Results

Overall, 28 participants took part in our study (7 female, 21 male), which lasted about 30-45 minutes. All of the participants were computer science students at the University of Vienna, the majority of whom were enrolled in an HCI course and recruited over that course. The only requirement for participation in the study was experience with the Android operating system. The students got a reward in the form of extra points for the HCI course.

Two of the participants were 26-35 years old, the remaining people were in the age between 18 and 25.

Nearly all of the participants actively use text messaging/SMS (27) and WHATSAPP (26) as instant messaging apps, followed by TELEGRAM (18), VIBER (8), FACEBOOK MESSENGER (4) and KAKAOTALK (2). LINE, ANDCHAT, SKYPE, SIGNAL, THREEMA and TANGO were used by one participant each. Regarding self-assessment of computer security knowledge, most of the participants said they had no or some knowledge about privacy and security mechanisms (7 respectively 17), while 4 stated to have a lot of knowledge. None of the participants claimed to be an expert in computer security.

Privacy and security on smartphone apps are of importance to the participants, and they care about third parties reading their messages. Confidentiality of text messages and active security / privacy measures were weighted to be of average importance. Regarding the first usability task (in which participants were asked to exchange a few messages with Bob and send a picture of the lab room), six participants were only partially able to complete the task, since SIGNAL's interface did not indicate whether the image had been send or not. Those pictures were only sent at a later point. All of the other participants were successful. In the second usability task participants were asked to set a passphrase for the app and import/export a backup of the app's data. While setting the passphrase seemed easy, six of the participants were unable to find the backup option. Most of the participants who failed

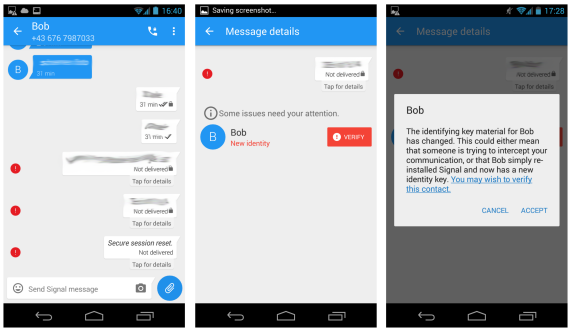

Fig. 3. Message delivery failure (1), notification about Bob's new identity (2) and new identity dialogue (3)

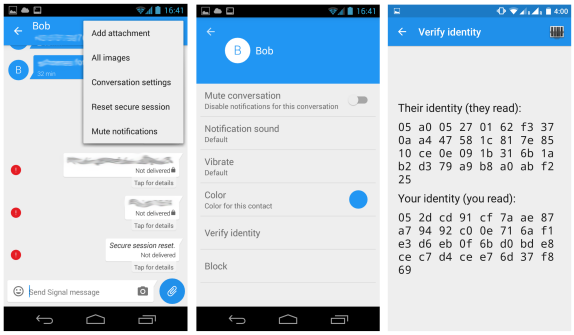

Fig. 4. "Verify identity" option in the conversation settings (1 & 2). Key comparison page displaying Bob's key at the top and Alice's resp. the user's key at the bottom (3).

in this task searched for a backup list item in the preferences section, with the wanted item being located in SIGNAL's main menu.

B. Users' Reactions to the Attack

Shortly before the third task the MITM attack was launched. After the launch of the MITM attack, messages sent through SIGNAL were not delivered since SIGNAL's protocol needs mutual keys to send messages. In consequence all of the users noticed the attack because of an error notification next to the undelivered message (see Figure 3), and clicked on the notification icon to open the error dialogue.

At this point the error dialogue already confronted the users with the task of verifying Bob. While 24 out of 28 users read the text in the subsequent dialogue, the remaining 4 directly chose the "Accept" option whilst skipping the text. These participants seemed to follow "the flow" of the dialogue to quickly reestablish messaging functionality.

Even if the participants were able to access the key comparison page, whether from the error dialogue or later in the task (8 users never did), the key verification page of SIGNAL's Android application did not provide any instructions on how to perform the actual verification. As Figure 4 shows (picture on the right), SIGNAL displays the Identity Keys of both communication partners, but no further instructions are provided. The participants of our study therefore faced problems on how to use the displayed keys. One participant e.g. stated: *"...ok, those are keys, but what am I gonna do with them?"*.

In total 13 users asked Bob into the room during this task for verification, however less than half of those users managed to successfully match keys with Bob (seven users). When keys were correctly compared, a message about verification failure was raised due to the MITM attack (see Figure 2). The error message, however, did not provide any information on consequences, further mitigation strategies or strategy changes. One participant thus said: *"Well great, and now what?"*, while another participant told us: *"To be honest...I have no idea what to do now."*.

C. Mental Models of the Attack

Ideally, Alice and Bob compare their keys in person for verification purposes to confirm their mutual identity. If Mallory launched a MITM attack on their conversation, Alice and Bob ideally recognize this type of attack, stop communicating over SIGNAL and uninstall the app. As previously stated, successful MITM attacks on SIGNAL result from their central key exchange services being compromised, Alice and Bob thus need to stop using SIGNAL. In consequence, successful verification of Bob with matching keys was at no point possible in our setup due to the MITM attack. However, 13 participants assumed that they had successfully verified Bob in the final questionnaire, while they failed to correctly compare keys with Bob. They therefore accepted Bob's new identity and would likely have continued to communicate over an insecure connection since they assumed it to be secure. Those users had different (false) verification strategies, which we discuss in subsection IV-C1. Seven users successfully matched keys with Bob. Only three of those assumed some sort of attack, but did not mention MITM in particular. Two of those users assumed they were not chatting with Bob, but with the attacker Mallory. Three other users thought that the app simply malfunctioned. Thus matching of the keys did not necessarily lead to the correct assumptions. We discuss our participants assumptions below. The rest of the participants (eight users) did not manage to compare keys with Bob and were unsure about having verified Bob or knew they had not. Five of those participants explicitly assumed a MITM attack took place. Subsequently, not all users picked correct mitigation strategies. An overview over strategies users would have chosen is outlined below.

1) Verification Strategies: Out of the 13 participants who thought to have verified Bob, but did not manage to do so by comparing the keys, 12 came up with different verification strategies. 6 assumed that accepting Bob's new key in the error dialogue following the attack successfully verified Bob. 4 "verified" Bob by either meeting him in person or by asking him questions about messages he received and his identity via chat or via phone calls. One person assumed that the presence of the keys on the key comparison page proves the authenticity of Bob's identity, while another person attempted to verify the authenticity of the chat by asking Bob whether he thought the chat was secure.

2) Assumptions about the Attack: In order to assess the users' assumptions about the attack we included an open question about the "unexpected events" in the final questionnaire. Spoken remarks in the Think Aloud protocol were also taken into account. Overall, 14 participants made remarks about possible explanations for the unforeseen events (multiple mentions

could be made). 7 participants speculated or stated that a MITM attack could have taken place, although only one of those participants compared keys correctly. As already stated not all the participants who successfully compared keys made the right assumptions about the events during the MITM attack. Several other incorrect assumptions were drawn: 4 participants stated that an attacker made an attempt to impersonate Bob, thus they assumed that they had compared keys with Mallory instead of Bob. Furthermore, 3 participants speculated that Bob could have reinstalled SIGNAL as suggested in the error message. Another 3 users assumed that the app was simply malfunctioning. 2 participants finally stated that an attack could have happened, but did not specify the type of attack.

3) Mitigation Strategies: The final questionnaire contained another open question about participants' possible mitigation strategies after the unexpected events. The type of attack was deliberately not revealed so as not to bias answers. Also the users' actions and remarks during the last study task were considered. Several possible mitigation strategies (not necessarily referring to MITM attacks in particular) arose from the answers: 11 participants would simply uninstall the app (the only valid mitigation strategy against compromise of the server), although it was not clear whether they wanted to avoid further hassle and would simply use another messaging app, or whether they knew it was the recommended mitigation strategy. Other strategies aimed at gathering more information, such as contacting Bob on another channel via other apps, phone or face-to-face meetings (8 participants), searching for information on the Internet (6 participants) or asking friends (4 participants). 3 participants would inform the developers or read license agreements and policies (3 resp. 1 participants). Another branch of strategies involved problem solving: restarting the app (2 participants), disconnecting the phone from the Internet (2 participants) or a virus scan (1 participant).

V. DISCUSSION

To the best of our knowledge we are the first to study the security, as well as usability, challenges of end-to-end-encrypted messengers. The central services used to exchange user keys pose the major security risk of today's end-to-end encrypted messengers. In our study we therefore simulate a compromised key service by performing an active MITM attack. Hence, we assess the usability of SIGNAL's security features in case of active attacks. However, like any user study, our work has some limitations:

First, the participants recruited for the study were homogeneous since all were students of computer science and shared the same age group. Similar experiments with different groups of participants might therefore lead to different outcomes. Second, we had to balance the extent of information we provided to participants on SIGNAL's encryption/verification features. We decided to explicitly ask users to verify each other in order to asses the usability of this core-security feature of SIGNAL. Our initial study design tested in our pilot study showed that none of the six participants used the verification feature in the face of our simulated attack. Similar experiments with participants without a computer science background and without a focus on a security subtask would likely result in even less successful key verifications.
Overall, we were surprised by the outcome of our study, especially given the fact that our participants had a computer science background. Our results suggest that the "verification" process and therefore the overall security of end-to-end encryption on mobile instant messaging faces serious usability obstacles, since 21 of our 28 participants failed to properly compare keys with their conversational partner. Especially surprising in our study was the high number of participants who thought they had successfully verified while in reality they failed to compare keys.

SIGNAL, as an easy-to-use end-to-end encryption enhanced app, should support struggling users to achieve security in the sense of increased usable security. Usability problems, in terms of missing support, can lead to serious security breaches, e.g. aborting the reestablishment of a secure connection after an attack. The gaps between self-assessment, mental models of differing correctness respectively level of detail as well as actual outcome (un/successful defense) could be explained in several ways: Either participants lacked the required knowledge, the app failed to support the users, they simply had a different understanding of what "verification" meant or the effort for successful defense was simply too high. During the MITM attack, SIGNAL was explicitly hinting at the fact that the connection could have been compromised. The fact that only 7 participants assumed the possibility of a MITM attack and only 3 thought that Bob reinstalled the app seem quite surprising. Either those users ignored, or did not read, the informational error message or simply excluded the possibility of an attack/reinstallation while remaining under the false illusion of security. The different strategies for verification and mitigation definitely hint at flawed mental models: users seem to lack an understanding of end-to-end encryption in general, possible attack scenarios and risk potentials. The findings from section IV-C1 also indicate a great trust by the users in the app to deal with security issues in the background, therefore assuming that the app's dialogues could be trusted.

A. Recommendations for SIGNAL

We think that SIGNAL can be improved in order to provide end-to-end encryption for the masses and further close the gap between insufficient knowledge on the users' side and possible support through the app. We suggest the following usability improvements to contribute towards an enhanced usable security experience for SIGNAL:

Awareness on security status of conversations: Conversations can only be assumed to be properly end-to-end encrypted once Alice's (the user's) and Bob's (the conversational partner's) Identity Keys were successfully verified. SIGNAL does not remember the verification status — only point-in-time verifications are possible and the user has to remember whom of his/her partners he/she already verified. SIGNAL thus lacks mechanisms to quickly assess the security status of a conversation. Such a security status should be directly visible in the corresponding conversation.

Comprehensible instructions for recommended actions: In order to avoid risky behavior, especially in the verification and attack mitigation process, users should be provided with clear instructions respectively suggestions for actions. On the key comparison page users with no exact knowledge of asymmetric encryption mechanisms failed to act on the

displayed information. In our opinion, a brief instructional message combined with optional further information would have led to a higher verification success rate (e.g. *"Please contact your partner outside the app to compare your Identity Keys. If the Identity Keys do not match, please consult the FAQ or contact the developers."*). We found that this issue is most pressing for the Android version of SIGNAL. The iOS version of SIGNAL provides brief information on how to verify users: *"Compare both fingerprints to verify your contact's identity and the integrity of the message"*. However, no information is provided on how to proceed in case of failure (fingerprint mismatch).

Clear risk communication: On the other hand SIGNAL should inform users of the possible consequences of their actions. E.g. during the process of accepting Bob's identity after the attack the denomination of the buttons ("Verify" and "Accept") was misleading. Under the false assumption that the mitigation process would lead to a verification of Bob, users failed to have a clear understanding of the risks.

Easily accessible verification: The verification options should be easily accessible in the menu. A suggestion would be to add a shortcut for the verification mechanism directly to the conversation in order to maximize visibility.

Based on our findings on the usability of SIGNAL's error handling of actual attacks, we found that these features led to more problems than to actual attack mitigations. Under these circumstances it is not surprising that WHATSAPP has disabled all encryption related error messages by default. If users want to get feedback on mismatching Identity Keys or alike, they explicitly have to enable the error messages in the preferences. Since reactions to non-comprehensible error messages (due to the interplay of potential missing information on the app's side and incomplete mental models on the user's side) range from uninstalling of the app, contacting the developers and/or a definitive feeling of insecurity in general, we assume the developers of WHATSAPP made a compromise between usability and security due to economical reasons. Since communication over WHATSAPP was only encrypted between the client and the server recently, messages on changed Identity Key might lead to confusion, ultimately angry users and eventually uninstallation.

VI. RELATED WORK

Usable security as relatively new field of research focuses on the development of secure systems including the people who actually use them [19]. Cranor e.g. argues that security failures often originate from unintentional mistakes by users of computer systems due to usability problems [20]. Previous work specifically on the usability of secure messaging focused to a large extend on PGP and S/MIME. A number of experiments showed that this first end-to-end encrypted messaging protocols were plagued with usability issues [1], [2], [3], [4]. These previous results might also explain why PGP and S/MIME have not, as yet, enjoyed widespread adoption. Assal at al. [21] explored mobile privacy through a survey and usability evaluation of three privacy-preserving mobile apps, including the Off-the-Record Messaging application ChatSecure [22]. They observed a high number of participants who thought their conversations were encrypted while they were not, mainly due to usability issues and incomplete mental models of privacy risks. The study of Assal et al. has a close relation to our work. However SIGNAL communication is encrypted by default and we focus on the unique usability challenges of SIGNAL.

Mental models as an internal representation of concepts have a great influence on cognition, reasoning and decision-making. Although being incomplete and inaccurate by nature, mental models are able to provide predictive and explanatory powers for understanding interaction [23], [24], [25]. Especially with security's complex problems and concepts, mental models of security or privacy mechanisms and possible threat scenarios play a major role in usable security research. Mental models mediate the processing of risk messages [26]. One possible threat scenario in consequence is for malicious software to take advantage of gaps in the users' mental models [27]. The same incomplete internal representations of concepts and threats proved to be the reason for low end-to-end encryption uptake, apart from the lack of usability [3]. Nevertheless mental models in usable security research can help to shed light on users' decisions in case of failure detection [28]. Our work extends research on the use of mental models in the area of usable security and proved helpful to better understand the usability issues our participants faced.

The most comprehensive work on secure messaging has been published by Unger at al. [29]. Their survey provides a current view on challenges for secure messaging, and as such provides additional context for our work especially regarding technical means to verify users and the mitigation of MITM attacks. Regarding the main focus of our work, SIGNAL, Frosch et al. [13] provide a detailed analysis of the underlying cryptographic protocol of SIGNAL. Schrittwieser at al. [30] discuss the different attack vectors like account hijacking, sender ID spoofing, enumeration and several other security issues of early mobile messengers. This study has been complemented by Rottermanner et al. [31], who focused on the unique privacy challenges posed by mobile messengers. With the exception of the work by Unger at al. [29], previous work on secure mobile messaging does not discuss usability issues of secure mobile messengers but rather focuses on purely technical issues.

VII. CONCLUSION

In this paper we presented a user study on the security and usability of SIGNAL for Android, a secure mobile messenger that provides a promising solution for widely adoptable end-to-end encrypted conversations. SIGNAL's protocol has recently been adopted by WHATSAPP, which means that over one billion users can now potentially exchange messages protected by strong encryption. We first discussed the unique security challenges and threats today's secure mobile messengers face. Second, we conducted a comprehensive user study on the usability of SIGNAL's security features. As part of our user study we simulated man-in-the-middle attacks and showed that the great majority of users failed to detect and deter such attacks. We finally proposed a number of improvements for SIGNAL to make the existing security features easier to use.

REFERENCES

[1] A. Whitten and J. D. Tygar, "Why johnny can't encrypt: A usability evaluation of pgp 5.0." in *Usenix Security*, vol. 1999, 1999.

[2] S. L. Garfinkel, D. Margrave, J. I. Schiller, E. Nordlander, and R. C. Miller, "How to make secure email easier to use," in *Proceedings of the SIGCHI conference on human factors in computing systems*. ACM, 2005, pp. 701–710.

[3] K. Renaud, M. Volkamer, and A. Renkema-Padmos, "Why doesn't jane protect her privacy?" in *Privacy Enhancing Technologies*. Springer, 2014, pp. 244–262.

[4] A. Fry, S. Chiasson, and A. Somayaji, "Not sealed but delivered: The (un) usability of s/mime today," in *Annual Symposium on Information Assurance and Secure Knowledge Management (ASIA'12), Albany, NY*, 2012.

[5] Forbes, "Gartner survey showing declining pcs, increasing mobile devices through 2017," 2013, http://www.forbes.com/sites/chuckjones/2013/04/05/gartner-survey-showing-declining-pcs-increasing-mobile-devices-through-2017.

[6] EFF, "Secure messaging scorecard v 1.0," online, 2015, https://www.eff.org/node/82654.

[7] Open Whisper Systems, "Signal messenger," online, 2016, https://whispersystems.org.

[8] O. W. Systems, "Open whisper systems blog: Just signal," Nov. 2015. [Online]. Available: https://whispersystems.org/blog/just-signal/

[9] The Intercept, "With facebook no longer a secret weapon, egypt's protesters turn to signal," online, April 2016, https://theintercept.com/2016/04/26/facebook-no-longer-secret-weapon-egypts-protesters-turn-signal/.

[10] WhatsApp Inc., "Whatsapp," online, 2016, https://whatsapp.com.

[11] EFF, "Whatsapp rolls out end-to-end encryption to its over one billion users," online, April 2016, https://www.eff.org/deeplinks/2016/04/whatsapp-rolls-out-end-end-encryption-its-1bn-users.

[12] 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*. ACM, 2004, pp. 77–84.

[13] T. Frosch, C. Mainka, C. Bader, F. Bergsma, and T. Holz, "How secure is textsecure?" 2014.

[14] C. Lewis, *Using the" thinking-aloud" method in cognitive interface design*. IBM TJ Watson Research Center, 1982.

[15] L. SaurikIT, "Cydia substrate," 2016, http://www.cydiasubstrate.com.

[16] M. Blanchou, "Android-ssl-trustkiller," 2016, https://github.com/iSECPartners/Android-SSL-TrustKiller.

[17] A. Cortesi, "mitmproxy," 2016, https://mitmproxy.org/.

[18] M. Lv, "Proxydroid," 2016, https://github.com/madeye/proxydroid.

[19] L. F. Cranor and S. Garfinkel, *Security and usability: designing secure systems that people can use*. O'Reilly Media. Inc., 2005.

[20] L. F. Cranor, "A framework for reasoning about the human in the loop." *UPSEC*, vol. 8, pp. 1–15, 2008.

[21] H. Assal, S. Hurtado, A. Imran, and S. Chiasson, "What's the deal with privacy apps?: a comprehensive exploration of user perception and usability," in *Proceedings of the 14th International Conference on Mobile and Ubiquitous Multimedia*. ACM, 2015, pp. 25–36.

[22] C. Ballinger, "Chatsecure - encrypted messenger for ios and android," online, 2016, https://chatsecure.org.

[23] P. N. Johnson-Laird, *Mental models: Towards a cognitive science of language, inference, and consciousness*. Harvard University Press, 1983, no. 6.

[24] N. Staggers and A. F. Norcio, "Mental models: concepts for human-computer interaction research," *International Journal of Man-machine studies*, vol. 38, no. 4, pp. 587–605, 1993.

[25] N. Jones, H. Ross, T. Lynam, P. Perez, and A. Leitch, "Mental models: an interdisciplinary synthesis of theory and methods," 2011.

[26] L. J. Camp, "Mental models of privacy and security," *Available at SSRN 922735*, 2006.

[27] R. Wash, "Folk models of home computer security," in *Proceedings of the Sixth Symposium on Usable Privacy and Security*. ACM, 2010, p. 11.

[28] C. Bravo-Lillo, L. F. Cranor, J. Downs, and S. Komanduri, "Bridging the gap in computer security warnings: A mental model approach," *IEEE Security & Privacy*, no. 2, pp. 18–26, 2010.

[29] N. Unger, S. Dechand, J. Bonneau, S. Fahl, H. Perl, I. Goldberg, and M. Smith, "Sok: Secure messaging," in *Security and Privacy (SP), 2015 IEEE Symposium on*. IEEE, 2015, pp. 232–249.

[30] S. Schrittwieser, P. Frühwirt, P. Kieseberg, M. Leithner, M. Mulazzani, M. Huber, and E. R. Weippl, "Guess who's texting you? evaluating the security of smartphone messaging applications." in *NDSS*, 2012.

[31] C. Rottermanner, P. Kieseberg, M. Huber, M. Schmiedecker, and S. Schrittwieser, "Privacy and data protection in smartphone messengers," 2015.