The Security Blanket of the Chat World: An Analytic Evaluation and a User Study of Telegram

Ruba Abu-Salma[1,*], Kat Krol[2,*,‡], Simon Parkin[1], Victoria Koh[1], Kevin Kwan[1], Jazib Mahboob[1],
Zahra Traboulsi[1], and M. Angela Sasse[1]

[1] University College London (UCL), {ruba.abu-salma.13, s.parkin, victoria.koh.13, kevin.kwan.13,
jazib.mahboob.13, zahra.traboulsi.13, a.sasse}@ucl.ac.uk
[2] University of Cambridge, kat.krol@cl.cam.ac.uk

***Abstract*—The computer security community has advocated widespread adoption of secure communication tools to protect personal privacy. Several popular communication tools have adopted end-to-end encryption (e.g., WhatsApp, iMessage), or promoted security features as selling points (e.g., Telegram, Signal). However, previous studies have shown that users may not understand the security features of the tools they are using, and may not be using them correctly. In this paper, we present a study of Telegram using two complementary methods: (1) a lab-based user study (11 novices and 11 Telegram users), and (2) a hybrid analytical approach combining cognitive walk-through and heuristic evaluation to analyse Telegram's user interface. Participants who use Telegram feel secure because they feel they are using a secure tool, but in reality Telegram offers limited security benefits to most of its users. Most participants develop a habit of using the less secure default chat mode at all times. We also uncover several user interface design issues that impact security, including technical jargon, inconsistent use of terminology, and making some security features clear and others not. For instance, use of the end-to-end-encrypted *Secret Chat* mode requires both the sender and recipient be online at the same time, and *Secret Chat* does not support group conversations.**

I. Introduction

Recent events have seen developers offering messaging tools with greater security to support a diverse range of user motivations. These include revelations about mass surveillance and the potential for user tracking in communication tools (e.g., Facebook's tentative plans to use WhatsApp user data [30]). End-to-end (E2E) encryption has been adopted in several messaging tools (e.g., WhatsApp, iMessage), whereas other tools have positioned security as a key selling point (e.g., Telegram, Signal). Security-related features may differ in how much they involve the user, whereas differences in the visibility of security features can create problems and impact user trust in a messaging tool [52], [53]. Telegram[1] is unique in offering separate modes of communication with differing levels of security. However, it may be difficult for users to distinguish between these modes and make effective use of them [31]. Users may explore the functionality of a messaging tool, or identify features that satisfy specific goals (which may or may not relate to security, such as sharing sensitive information with others). Users new to a security tool may also use it in ways that are not anticipated by developers [46].

Here, we explore the motivations and security behaviours of using a messaging tool that claims to be secure, specifically those who have not used Telegram before and those who are familiar with the tool. We combine two research techniques: (1) a novel lab-based user study with 11 novices and 11 participants with prior experience of using Telegram, and (2) a usability inspection bringing together cognitive walk-through and heuristic evaluation, focusing on Telegram's UI. This approach has been applied before in the area of usable security, most notably by Whitten and Tygar [62] to evaluate PGP 5.0. Here, we have planned a lab-based study that uses a set of tasks to elicit user perceptions of Telegram. The usability inspection complements this by allowing us to look at issues not touched upon by those tasks or not reported by our participants.

Prior work has focused on novices, with the admirable goal of identifying barriers to adoption [52], [62]. Studies of secure communication tools have rarely involved non-novices, where these users can identify the motivations for adopting and using security features in practice. Participants brought their mobile devices to the lab. Novices installed Telegram to explore its features by way of a 'sensitive payment information' messaging scenario. Prior users of Telegram were similarly involved in the task, but as an opportunity to see how they have used the tool and the role of Telegram's various security features in these practices, such as the *Secure Chat* mode. In both cases, scenario tasks were used to promote discussion as part of semi-structured interviews. Use of a System Usability Scale (SUS) questionnaire further explored the usability of the tool for novices and users alike. We found that both groups

[*] Authors contributed equally.

[‡] The study was conducted while the author was at University College London (UCL).

[1] https://telegram.org/

EuroUSEC '17, April 29, 2017, Paris, France
Copyright 2017 Internet Society, ISBN 1-891562-48-7
http://dx.doi.org/10.14722/eurousec.2017.23006

associated high SUS scores with Telegram, even where prior users had been found to use a few features – including security features – in practice.

Comparing the initial impressions of novices with the experiences of non-novices illustrated that in practice there may be at most *partial adoption* of Telegram and its security features by regular users. The participants with prior experience of Telegram tended to abandon the tool despite believing it was usable, adopting a communication tool which was more popular with the people they wanted to remain in contact with. Similarly, participants would benchmark Telegram and its features against a favoured communication tool (e.g., WhatsApp). For both groups, when participants wanted to exchange sensitive information, they would use a mix of communication channels (such as calling a person's phone or meeting in person), rather than exclusively using even a trusted tool, such as Telegram (and indeed other messaging tools).

The consistency of information for users was also a source of issues for both novices and non-novices. Participants did not generally understand that Telegram's *Secret Chat* mode does not support group conversations, nor that the communicating parties are required to be online at the same time. This leads to recommendations to practitioners, foremost that usability engineering principles (such as those advocated by Nielsen and Molich [45]) must be followed consistently, or they risk leading users into using tools in such a way that the security properties become unclear.

II. Related work

Secure communication tools became widely available with the release of PGP in 1991 [63], designed for asynchronous, high-latency email communications. This led to a large ecosystem of PGP tools [1], [2], [5]. The release of OTR [3] in 2004 in turn addressed low-latency messaging environments, such as chat clients, and introduced additional security features, such as forward secrecy and deniability. The emergence of OTR has also led to a range of secure communication tools [7], [10], [11], [27], [41], [57], including the Signal protocol [4], which has recently gained popularity [4].

The use of self-destructing messages was popularised by Snapchat, released in 2011. This was seen by users as addressing their security and privacy needs, though motivated attackers remain a problem, as does secure data deletion in messaging [26], [47], [49]. There is also uncertainty as to whether tool providers are able to uphold assurances around security in the face of, for example, government requests for data [51]. We refer the reader to Unger et al. [60] for a detailed review of the literature on secure communication tools.

The main user interface (UI) challenge for E2E-encrypted messaging tools is seen as being *trust establishment* [60], which is often reduced to verifying ownership of cryptographic keys in some fashion. In traditional PKI, this assurance is delivered in the form of a signed certificate from a trusted authority [33]. There are, however, many issues with this certificate management, including key storage, distribution and revocation [28]. The most widely-deployed E2E-encrypted tools (e.g., iMessage, WhatsApp, Signal) shield users from key management, by using a trusted server that vouches for the authentic public keys of other users. Recent proposals attempt to limit the trust in these servers using transparency logs [42], [54].

The smartphone era has seen an explosion of new communication tools to support the sharing of messages. Many of these tools may claim to be secure, but lack specific security guarantees or adequate grounding in existing cryptographic literature [16], [60]. Responses have emerged in the community, such as the Electronic Frontier Foundation (EFF) Secure Messaging Scorecard, aiming to provide objective information about secure communication tools and their features. User comprehension and adoption of security-related features persist as barriers to secure communications.

Lack of usability can hamper both the adoption of secure communication tools and their effectiveness during use. In their seminal paper [62], Whitten and Tygar assessed whether PGP 5.0 could be used effectively by non-specialist users to secure their email. Analysis was informed by a hybrid analytical approach combining cognitive walk-through [39] and heuristic evaluation [45], alongside a lab-based user study. Security-related problems were identified in the UI (e.g., irreversible errors, lack of consistency and feedback). Only one-third of participants were able to correctly sign and encrypt their email messages during a 90-minute session; it was concluded that to be usable, security needed domain-specific UI design principles and techniques.

Similarly, Garfinkel and Miller studied CoPilot, a prototype email client based on Key Continuity Management (KCM) [24]. KCM attempts to reduce user effort and potential for error by automating key generation, key management, and message signing. Garfinkel and Miller concluded that KCM is a workable model for improving email security, and that the UI of CoPilot enables users to send protected emails easily because, for example, it visually distinguishes encrypted emails from unencrypted ones. Ruoti et al. studied Private Webmail (Pwm) and Message Protector (MP) [53]. They found that users trusted MP more than Pwm because with MP they "*could see the ciphertext*" after encryption had taken place, equating this with protection.

More recently, Ruoti et al. tasked pairs of novices to send encrypted emails using a range of email tools [52]. It was found that trust in the system was reduced when security properties were hidden from users. The authors concluded that integrated encryption solutions increase usability, but that complete transparency is counterproductive. The need for visible feedback matches the findings of Whitten and Tygar [62] as well as the "*visibility of system status*" usability engineering principle encouraged by Nielsen and Molich in 1990 [45].

Bai et al. investigated whether non-expert users can evaluate the security trade-offs between a traditional key-exchange model (analogous to PGP) and a registration model (analogous to iMessage) [8]. Participants preferred (and also trusted) the apparently more usable, but less secure, registration model for "everyday communications". Bai et al. concluded that designers ought to explain the security properties of their encryption tools.

Gaw et al. explored the social context behind users' decisions about whether and when to encrypt emails [25]. They interviewed members of an activist organisation under the presumption that the organisation's employees would have a strong incentive to encrypt emails. They found that the perception of encryption behaviour by others (e.g., use of encryption for protecting secrets is seen as "justified", for general communications as "paranoid") influenced participants' decision to adopt encrypted email.

Renaud et al. proposed seven possible explanations for the non-adoption of E2E encryption in email [50]. They identified key factors for non-adoption, which included usability issues, incomplete threat models, misaligned incentives, and lack of understanding of the email architecture. They concluded that security researchers should focus on building "*comprehensive mental models of email security*".

Dourish et al. studied how users experience and practice security as an "*everyday, practical problem*" [21]. We here aim to understand users' security practices and needs, and the background against which they decide to use or stop using a messaging tool that claims to be secure.

III. Telegram

Telegram is an IM tool launched in 2013. Client applications exist for both desktop (e.g., OS X, Windows) and mobile systems (e.g., Android, iPhone). Telegram supports the exchange of messages, photos, videos and files between individual users or groups of users. In February 2016, Telegram announced that they had more than 100 million monthly active users, generating 15 billion messages daily.

Telegram accounts are tied to phone numbers, verified via an SMS message or a phone call. Users can create an alias (public username) to limit exposure of their phone number. A passcode can be used to lock an account, and multiple devices can be linked to a single account. An optional "two-step verification" feature requires a password in order to access an account from a new device (in addition to the verification code). Users can delete their accounts manually. Otherwise, an account is automatically deleted after a period of inactivity.

Telegram has been launched with security as a key selling point. Telegram offers both a default chat mode and a *Secret Chat* mode. Messages sent within the default mode are cloud-based and stored on Telegram's servers, supporting synchronisation of messages across a user's connected devices; in this mode, messages are encrypted in transit using client-server/server-client encryption. In contrast, messages exchanged within the *Secret Chat* mode are E2E-encrypted, and can only be accessed from the originating device (i.e., not cloud-based). When a *Secret Chat* session is initiated, participating devices exchange a long-term encryption key using Diffie-Hellman key exchange [20]. After an E2E-encrypted session has been established, communicating devices use this key to exchange encrypted messages using a symmetric encryption protocol, called MTProto.

When a *Secret Chat* session has been established, both an image and a textual visualisation are generated on both the sender and recipient's devices, visualising their public key fingerprints. Communicating parties would verify each other's identities by comparing both representations through a trusted channel (e.g., in person). If the representations match, the session is secure, and a man-in-the-middle attack has not occurred. Users can initiate different *Secret Chat* sessions with the same contact. Telegram claims that secret chats can be deleted at any time, and can, in principle, self-destruct, i.e., disappear from both the sender and recipient's devices after a period of time, without storing a copy of exchanged messages on Telegram's servers. Users have the option to set the "self-destruct timer" to the desired time limit. Secret chats cannot be forwarded to other users, and, according to Telegram, leave no traces on the servers.

Telegram's client-side code is open-source, whereas its server-side code is closed-sourced and proprietary. The EFF Secure Messaging Scorecard provides information to non-expert users about the security features offered by various messaging tools [22]. Currently, the EFF Scorecard gives Telegram's default chat mode a rating of four out of seven points, and the *Secret Chat* mode seven out of seven. However, more recent audits have revealed that Telegram has a wide range of security issues that might compromise its integrity as a secure messaging tool [34]. Telegram uses its own "home-brewed" encryption protocol, MTProto, rather than well-studied, provably-secure encryption schemes that achieve strong definitions of security (and that are at least as efficient). The tool also leaks meta-data, which may allow an attacker to determine when a user is online and offline, when they use the tool, and where they are located [58].

IV. Study aims

The aim of our study was to evaluate the user experience of Telegram and its security features. We were interested in two main aspects, specifically its usability and the factors for adoption. We chose Telegram because it was launched with security as a key selling point, and has recently gained significant popularity. That the Telegram architecture distinguishes between two distinct chat modes further facilitated exploration of user perceptions and motivations around secure messaging: messages sent within the default chat mode are encrypted in transit, whereas within the *Secret Chat* mode are E2E-encrypted. Abu-Salma et al. [6] found that participants who reported using Telegram to send sensitive information had been doing so using the less secure default chat, rather than the *Secret Chat*, mode.

Studies summarised in Section II have further motivated questions around the uptake of secure communication tools and their security-related features. Hence, we studied both novices and people with prior experience in using Telegram to contrast their experiences. We conducted a two-part study to map the factors impacting adoption and effective use of Telegram. The interview results showed that participants were not making regular use of the features of the *Secret Chat* mode. Hence, we conducted a usability inspection to understand if the UI was a potential source of confusion for users.

V. User study

In this section, we give an overview of our user study, detail participant demographics, and describe the task-based scenario, complementary study questionnaires, and post-study interview plan.

A. Method

1) Participants: We recruited our participants via UCL's Psychology Subject Pool. We asked prospective participants to complete a short online pre-screening questionnaire, providing basic demographic information (e.g., gender, age) and contact details (e.g., name, email address). We also provided participants with a list of different communication tools, asking them to (1) indicate which tools they currently or previously used, and (2) name any other tools they had used which were not on the list – Telegram was among the choices.

Overall, 210 individuals completed the pre-screening questionnaire, 27 of whom reported having used Telegram. We divided the pool of eligible participants into sub-groups based upon these demographics, with the aim of achieving a balance of age and gender in both groups: novices (first-time users) and users who have used the tool. We selected 11 novices and 11 users of Telegram to bound a focused, exploratory study.

Overall, 22 participants took part in our study, 12 females (six novices and six users) and 10 male participants (five novices and five users). Mean age was 31 (range: 19–75). Four completed high-school education, 14 had an undergraduate degree (e.g., BA, BSc), and four had a higher degree (e.g., MA, MSc, PhD). Our participants used several communication tools on three computing platforms: Android (14), iOS (7) and Windows (1).

2) Task-based scenario: We designed a face-to-face laboratory session consisting of both a task-based scenario using Telegram and a post-study interview (lasting for approximately one hour). Upon arrival, a participant would be told the general objective of the study, which was the assessment of the user experience of Telegram. The individual's rights as a participant were also outlined, supported by a printed information sheet and a consent form. The examination of perceptions around security was explicitly mentioned in a debrief as part of the post-study interview to invite participants to further reflect upon the UI and functionality of Telegram. Each participant received a £10 Amazon voucher for their participation.

We asked novices to install Telegram on their own personal mobile devices, rather than using a device which would be unfamiliar and could hinder use of the tool. We maintained back-up mobile devices in the event of technical problems (Nexus 6, Android 6.0; iPhone 6, iOS 10.1.1). Four participants used these; two Android users and two iOS users. After installation, novices were given time to explore the features of the tool.

Participants were introduced to a researcher with whom they would communicate during a structured scenario. In the scenario, the researcher acted as a trustworthy colleague, with whom they ran a (fictional) small business. They were required to send 'sensitive' financial details using Telegram in order to complete a mock purchase. Participants referred to a paper booklet containing a randomly generated credit card number, card validation value (CVV), and card expiry date. We asked participants to treat these details in the same way they would treat their own sensitive information. Participants were free to choose which chat mode (default or *Secret Chat*) to send the sensitive information with no prompting from researchers, and were encouraged to think aloud throughout the activity.

We then asked participants to send a greeting to the 'colleague', who would then prompt them to start a *Secret Chat* conversation. Upon switching to the *Secret Chat* mode, participants were asked to give first impressions of the summary of security features displayed to them (as shown in Figure 3), and to test the self-destruct timer.

3) SUS: Upon completing the study tasks, our participants were asked to complete the SUS questionnaire, a tool commonly used to assess the usability of technologies [13]. The SUS is a 10-item questionnaire for capturing subjective assessments of usability. Each question has five possible responses on a Likert scale, ranging from "Strongly agree" to "Strongly disagree". SUS is generally used after use of a system or product, but before any debrief. Many studies have considered SUS to be a good indicator of perceived usability [59], and it has been used, for instance, to assess secure email systems [52], [53].

4) Post-study interview: After completing the SUS questionnaire, participants were asked about their general impressions of Telegram. The interviewer would also explore participants' opinions of other tools that they had used, and how they compared to Telegram. This was also an opportunity to explore perceptions of security, privacy and information sensitivity. Notable events during the session would be revisited during the post-study interview (e.g., whether the use of one chat mode or the other was incidental or deliberate). This may in turn be influenced by real-life experiences and approaches to sharing sensitive information.

5) Pilot study: We conducted a pilot study to calibrate the study design and protocol (e.g., duration, response to technology failures). We used the common practice of convenience sampling [29], selecting six colleagues and friends (three novices and three users of Telegram).

6) Data analysis: The study sessions were recorded and the recordings were transcribed. All transcripts were coded by two researchers using Thematic Analysis [12]. First, three researchers coded five transcripts and created their codebooks independently. Then, they met to discuss the themes and the most suitable codes to capture them. One researcher then unified the codebooks into one. The unified codebook was used by two researchers to code the 22 interviews in batches, meeting weekly to discuss the meaning of participant statements, and revise the codes by adding, merging and removing codes. The results presented in the following sections of the paper reflect the agreed themes.

7) Research ethics: The study was reviewed and approved by UCL's Research Ethics Committee (project no.: 3615/008). At the start of each session, we asked our participants to read an information sheet and sign a consent form outlining the study. We emphasised that all data collected was treated as strictly confidential, and handled in accordance with the provisions of the UK Data Protection Act (registration no.: Z6364106/2016/02/71). Participants had the option to withdraw at any point during the study without needing to provide a reason. The SIM card used by the experimenter was solely purchased for the purpose of the study and destroyed after the last study session.

B. Results

First, we report the SUS scores provided by our participants to assess the usability of Telegram. Second, we present the key themes that emerged across both the 'think aloud' task scenario and post-task interviews. We report how many participants mentioned each theme to give an indication of the frequency and distribution of key points. Discussion was participant-led, meaning that the frequency of themes across the participant group may be lower than the real frequency; a participant not mentioning a belief does not mean that they do not hold that belief. We use the following prefixes: 'PN' for novices, and 'PU' for users.

1) SUS: We Asked participants to complete the SUS questionnaire, facilitating a quantitative comparison of usability across participants, especially given that we interviewed both novices and users of Telegram. SUS scores range from 0 to 100. Table I in the Appendix shows the SUS scores for each group (novices and users).

Bangor et al. analysed 2,324 SUS surveys from 206 usability tests over a ten-year period, and derived a set of acceptability ranges ('not acceptable', 'marginal', and 'acceptable') to generally describe the usability perceived by users [9]. Later, a letter-grade scale (A, B, C, D and F) was used to interpret SUS scores. Bangor et al. also associated specific ranges of SUS scores with adjective descriptions (e.g., excellent, good, poor). In our study, Telegram's SUS scores of 83.4 (novices) and 81.8 (users) indicate that Telegram has "excellent" usability (letter grade: B). The difference between the scores of the two groups of participants was not statistically significant (unpaired t-test, $t = 0.36$, $p = 0.72$).

2) Adoption and abandonment of Telegram: We asked the prior users of Telegram to tell us the story of how they adopted the tool. For seven of them, the tool was suggested to them by one of their contacts. Eight participants said that they explicitly chose it for reasons relating to security. For example, PU06 used Telegram when taking part in a protest: *"people at the time downloaded Telegram to send messages to talk about situations at their sites."*

Three participants stated that they tried Telegram out of curiosity. Nine participants associated Telegram with low popularity, and saw this as an obstacle to continued use. PU04 explained: *"I personally would prefer to move to Telegram 'cause I don't like Facebook as a company but so few people use it that I'm still stuck using WhatsApp for most things."*

Out of 11 users of Telegram, seven stated that they used Telegram for casual conversations rather than sensitive content, and seven mentioned that they used the tool to keep in touch with some of their contacts who were using Telegram exclusively. Six users of Telegram mentioned that they stopped using the tool because it does not support voice calls, as opposed to WhatsApp and Viber.

Upon arrival to the lab, we asked participants in the user group if they still had Telegram installed on their phone. Out of 11 participants, six stated they had uninstalled it, whereas four said that they still had the tool on their phone due to having sufficient storage space to be able to keep it there.

3) Secret Chat: Eight of 11 novices did not encounter the *Secret Chat* mode during the initial exploration, despite encouragement to explore Telegram and assurance that they could take as much time as they felt was necessary to do so. The other four novices believed that tapping on a contact's name, to begin messaging them, would enter the *Secret Chat* mode, not the default chat mode (as also uncovered during the usability inspection described in Section VI-B).

Five participants (three novices and two users) associated *Secret Chat* with one-to-one chat; i.e., a private chat mode to message only one person rather than multiple users. Telegram's default mode supports group conversations, whereas *Secret Chat* does not. PN05 questioned whether *"the recipient uses the Secret Chat mode"* if the sender initiates a *Secret Chat* session, viewing the modes as options that senders and recipients can choose from independently. PU16 believed they had always been using the *Secret Chat* mode for group chats, despite this not being supported.

Starting a *Secret Chat* session with a contact prompts a list of the mode's security features, specifically: "use E2E encryption", "leave no trace on our servers", "have a self-destruct timer", and "do not allow forwarding" (Fig. 3). Below, we review participants' perceptions of these features. We also discuss participants' views on verification fingerprints.

Using E2E encryption. Out of 22 participants, 20 (11 novices and nine users) told us that they had heard of the term "encryption". Two participants (PU16 and PU17) stated they did not know what encryption meant.

When asked about encryption, six participants (three novices and three users) provided explanations relating to security and safety. These included *"an extra barrier of security"*, *"more time is needed to know the content of the message"*, and *"making chats safe from hacking until they get deleted from the servers."*

Regarding the barrier analogy, some participants variously referred to encryption as *"a security blanket around the message"*, *"a password to unlock the message"*, "*hiding the message in a box"*, *"a message protected by a series of programs"*, *"covering the IP address of devices"*, *"a message cannot be intercepted by other people"*, and *"a thread or tunnel is used to exchange messages through."* The capacity to see what is in a message was also alluded to by a number of participants, such as *"writing a message in a way only the sender and recipients can understand"*, *"sending a message in a different format"*, *"using all sorts of numbers to jumble up the message"*, *"looking like gobbledygook to people outside"*, *"seeing a bunch of signs that do not make sense when reading an encrypted message"*, and *"turning the language into something else for reading the other device."*

Overall, 17 participants (nine novices and eight users) offered explanations for what the 'ends' were in E2E-encrypted communications, including: sender and recipient; sender and recipient's phones; sender and Telegram's server; the start and end of a message body.

In terms of the role of encryption in protecting messages, participants variously mentioned that a person who does not have the phone number cannot read the message, that a special piece of software (not Telegram) is necessary to decrypt a message, and that encryption is ineffective against people sitting next to the sender/recipient. We found that 20 participants believed that "someone" could access and read E2E-encrypted messages. These entities included government agencies, Telegram's service providers (sometimes referred to as "staff"), competing companies (e.g., Apple, Google), and tech-savvy people (sometimes referred to as "hackers" by our participants).

Leaving no trace on servers. Out of the 22 participants, one novice (PN07) linked "no traces" to protecting the meta-data of communications, stating that Telegram claims to protect meta-data including users' online/offline status, their location, the identities of the communicating parties, and how many messages have been exchanged. The 21 remaining participants believed that leaving no traces on servers meant that exchanged messages would not be stored on Telegram's servers, confusing it with the functionality of the self-destruct timer (which is discussed next).

Having a self-destruct timer. Telegram claims that messages sent via *Secret Chat* can (in principle) self-destruct, that is, disappear from both the sender and recipient's devices after a period of time with no copy of exchanged messages persisting on Telegram's servers. 12 participants (seven novices and five users) said that *Secret Chat*'s self-destruct timer is useful for sending sensitive information, but not for daily conversations (where 'daily' was regularly equated to the conversation being trivial). Similarly, the capacity to keep a log of exchanged messages (the 'chat history') was seen as important. Several participants believed that message expiry was defeated by the capacity to take a picture of their phone's screen (including timed messages). All prior users of Telegram explicitly mentioned that they had not set up the timer for sending a message. Some participants (three novices and seven users) also doubted how effective the feature is because, as PN05 put it, *"the practical thing is that nothing in electronic media is ever destroyed."*

The understanding of what the self-destruct timer did varied across participants: some thought that all their previous messages would be destroyed once a timer had been set up, whereas others were unsure if the messages would also be destroyed in the recipient's phone (as likened to Facebook Messenger). Similarly, there was confusion as to whether both the sender and recipient would have to forcibly or bilaterally set up the timer to the same amount of time, and whether the timer is applicable to group conversations. PN11 believed that the default chat mode should have a self-destruct timer, instead of the timer being exclusive to *Secret Chat*, judging sensitivity on a message-by-message basis: *"first of all if it's encrypted for all messages, which would be safer and more useful and then if the timer is made available on a regular chat and if you need to use it, you use it; otherwise you don't."*

No forwarding. 17 participants (six novices and 11 users) considered the "no forwarding" feature as not useful because messages could still be copied and pasted, and taking screenshots is possible (where Telegram v3.16.1 detects the taking of screenshots, but does not prevent it).

Verification fingerprints. When the user taps on a contact's name within the *Secret Chat* mode, a screen entitled "Encryption Key" is presented, as shown in Fig. 4. Public-key fingerprints are used for verifying the identities of communicating parties. All 22 participants were unsure of the usefulness of fingerprints, where all 11 users of Telegram had not used this feature despite their experience with the tool. They variously speculated that the image is the encrypted message sent to the recipient, the key used to encrypt/decrypt, or a sign that messages sent via the *Secret Chat* mode are E2E-encrypted.

4) Comparison of chat modes: When we asked participants to send fabricated payment card numbers to their mock colleague (as per the task scenario described in Section V-A2), seven novices sent it via *Secret Chat*, and four used the default chat. In the group of Telegram users, two participants used the *Secret Chat* while nine used the default mode.

We asked participants to tell us if they saw any differences between the two chat modes in Telegram. Overall, 17 participants said that the *Secret Chat* mode was more secure than the default chat mode, and nine pointed at the four features that the *Secret Chat* advertises to its users. 12 participants speculated that the default chat mode offered no encryption of messages, whereas two participants expressed hope – rather than pointing to particular listed features – that the default chat had some form of encryption. Furthermore, three participants stated that the default chat mode had some encryption, but that it was weaker than that of the *Secret Chat*. 11 participants stated that only the timer was the distinguishing feature that made the *Secret Chat* more secure. Five participants stressed that the default chat mode was equivalent to other messaging tools, such as WhatsApp.

Eight users of Telegram stated that they had never used *Secret Chat*, and five stressed that they did not feel they communicated information that was sensitive enough to warrant using it. Seven participants also mentioned that *Secret Chat* is not useful because it does not support group conversations.

5) Sharing of sensitive information: After the scenario tasks, we asked participants if they had ever used a communication tool to share their own payment card numbers – or any other information that they regarded as sensitive – with others in real life. Most participants had experience with sharing payment card numbers with others, where participants preferred in-person sharing. 11 of the 22 participants stated that they had shared sensitive information over the phone. Eight of them argued that phone calls were more secure than text messages, and three mentioned calls as being more secure than email messages. Otherwise, three participants reported splitting up a number into segments and sending it by different communication methods to the same recipient, or by the same method to several different recipients (e.g., family members).

C. Secure communications

The discussions around the two chat modes and encryption often led to more general questions of what "secure communications" mean to our participants. Out of 22 participants, 13 equated security with confidentiality. For PU2, "security" meant *"the fact you can send something to someone without somebody else, even if they know you."* Specifically, three participants took security to mean making sure the right person is receiving their messages.

1) Comparison with other messaging tools: Overall, 21 participants made some comparison between Telegram and other messaging tools during the session. 16 participants (eight novices and eight users) stated that it was similar to WhatsApp, where 10 of these found the UIs to be very similar. PN09 wondered if Telegram and WhatsApp were developed by the same company, whereas PU06 stated: "*I basically think Telegram is a fake version of WhatsApp. I don't know, everything is similar. With some… other functions that are not relevant to me.*" 15 participants (nine novices and six users) believed that Telegram had more options than WhatsApp. Four participants described Telegram as faster, and three of them mentioned photo downloads being smoother. In terms of security, 12 participants (eight novices and four users) felt that Telegram was more secure than WhatsApp.

2) Security and privacy settings: Telegram has several privacy and security settings, some analogous to features found in other popular messaging tools, such as WhatsApp and Facebook Messenger. These include the ability to block contacts, a visible "last seen" status for contacts, and active sessions. Here, we explore participants' perceptions of three security-specific functions: passcode, two-step verification and account deletion.

All but one participant (PN09) believed that a passcode was used to lock their Telegram account, whereas PN09 associated passcodes with specific chat sessions. Eight participants, four in each group, felt that a passcode would provide "extra security", however, two novices and three users saw the feature as lacking usefulness because the participant could not know who the sender is, and what the content of the messages received is. Forgetting a passcode was seen as a risk, resulting in the loss of prior messages and a need to reinstall Telegram. Eight novices implied that they would set a passcode, whereas nine users of Telegram had not done so, seeing the phone lock of their handset as sufficient (i.e., Apple's Touch ID or passcode, and Android's pattern/PIN lock).

Telegram has an optional two-step verification feature to further verify users when logging into their accounts from a new device (in addition to the verification code). However, only two novices and one user understood the two-step verification feature after reading the description of the feature on the screen. 17 participants (eight novices and nine users) equated this feature with two-factor authentication, seeing it as analogous to having an additional password for online banking and email accounts. Based on this belief, participants said that setting two passwords is *"a hassle"* (PU18), and *"having one password is enough"* (PU20). PU21 believed that the feature is used to reset a password, which may be attributed to them not having used the feature before. None of the 11 users of Telegram had used this security feature before.

Telegram allows users to delete their accounts, and accounts are automatically deleted after a period of inactivity. Although five novices and one user perceived the feature as useful, many participants (six novices and ten users) would prefer to receive a notification that their account would be deleted. Several participants believed that uninstalling a tool would also remove their account; five participants expressed a wish to keep a copy of their chat history.

VI. Usability inspection

A. Method

Usability inspection is an approach used to find usability problems in UI design [44]. Evaluating a design without users can identify problems that may not necessarily be revealed by an evaluation with users [19], [36], [40], [44]. It is important to bring users into the design process, but evaluating the design when no users are present can also provide benefits [36]. There are several different usability inspection methods, where we use a hybrid approach combining two methods: cognitive walkthrough and heuristic evaluation. Both methods are actively used in human-computer interaction (HCI) research [32].

1) Cognitive walkthrough: Cognitive walkthrough focuses on evaluating a design for its exploratory learnability, a key aspect of usability testing [56] based on a cognitive model of learning and use [39], [48], [61]. Users may prefer to learn a system by exploring it, rather than investing time in comprehensive formal training [14], [23]. A cognitive walkthrough identifies problems users can have as they approach an interface for the first time, as well as mismatches between users' and designers' conceptualisation of a task and the assumptions made by designers about users' knowledge of a specific task (which can for example impact the labelling of buttons).

Cognitive walkthrough is task-specific, studying one or more specific user tasks. The process comprises a preparatory phase and an analysis phase. In the preparatory phase, reviewers decide and agree on the input to the cognitive walkthrough process: (1) a detailed description of the UI, (2) its likely user population and context of use, (3) a task scenario, and (4) a sequence of actions that users should accurately perform to successfully complete the designated task. In the analysis phase, analysts examine each of the user actions needed to accomplish the task. The evaluation procedure follows a structured series of questions, derived from a theory of exploratory learning, to evaluate each step (or action) in the task workflow. A detailed overview of the cognitive walkthrough procedure can be found in [61].

2) Heuristic evaluation: Heuristic evaluation involves having usability evaluators judge dialogue elements in an interface against established usability principles (or "heuristics") [44], [45]. Ten principles, derived by Nielsen and Molich in 1990, can be found in [45]. The use of a complete and detailed list of usability heuristics as a checklist is considered to add formalism. Jeffries et al. found that heuristic evaluation uncovered more issues than any other evaluation methods, whereas empirical usability testing (user testing) revealed more severe, recurring and global problems that are likely to impede users [35].

3) Hybrid approach: To avoid biases inherent in either of the inspection methods, we used a hybrid approach combining the two methods. Combining both task scenarios and heuristics was recommended by Nielsen [43] and Sears [55]. The procedure is as follows:

1) Provide a detailed description of the UI.
2) Define the users and their goals.
3) Define the tasks the users would attempt (e.g., sending a message).

4) Break each task into a sequence of sub-tasks (or actions).
5) Walk through each task workflow step-by-step through the lens of the user (e.g., what terms they use, the things they would look for, the likely paths they would take).
6) For each action, look for and identify problems based on a set of heuristics (or guidelines).
7) Specify where in the UI the problem is, how severe it is, and possible design fixes.

Two researchers inspected the UI of Telegram using the hybrid approach, doing so independently before discussing the results. A third researcher read both evaluation reports, identifying usability issues in each report and aggregating all the uncovered usability problems in a larger set.

4) Privacy by design: Apart from assessing Telegram using usability inspection methods, we also examined the tool using the seven foundational principles of *Privacy by Design*, as defined by Cavoukian [15]. The principles advocate that systems should be designed with the preservation of privacy as a requirement, and that settings that ensure security and privacy should be the default.

B. Results

We used a hybrid approach combining cognitive walkthrough and heuristic evaluation to assess the usability of Telegram based on a set of tasks. In any messaging tool, users perform two core tasks: sending and receiving messages [60]. In secure communication tools, users would normally need to perform additional security tasks, such as deciding whether to manually encrypt/digitally sign messages. However, Telegram secures all messages by default without user interaction. Hence, our focus is on evaluating the two core tasks of sending and receiving messages, and the related security and privacy settings offered by Telegram. We evaluated Telegram v3.16.1 on both iPhone 6 (iOS 10.1.1) and Nexus 6 (Android 6.0.1). We did not find any meaningful differences for the tasks studied between these two devices.

As shown in Fig. 1, the user is presented with three ways of sending a message: "New Group", "New Secret Chat", and "New Channel". However, tapping on a contact's name in the list underneath (here: Kat) directs the user to the default chat mode, without giving them the option to choose between either the default chat mode or Telegram's *Secret Chat* mode. The default chat mode is not displayed as an option, but is activated once the user taps on the contact's name. This violates Nielsen's criterion of *visibility of system status*, which stresses that the system should always keep users informed about what is happening [32].

When starting a *Secret Chat* session with a contact (here: Kat), Telegram provides a list of the security features offered by this mode (Fig. 3). The four bullet points state that *Secret Chats* "use end-to-end encryption", "leave no trace on our servers", "have a self-destruct timer", and "do not allow forwarding". The terms assume a familiarity with technical terminology, and there is no direct link to further explanation. This violates Nielsen's heuristic of *match between system and the real world*, which states that the system should speak the users' language, with words, phrases and concepts familiar to users, rather than system-oriented terms [32] – as seen in Section V-B3, participants differed in how they perceive and articulate security terminology.

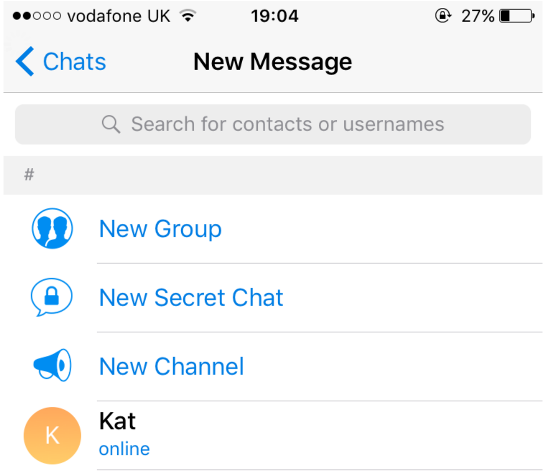


Fig. 1. Options to send a message in Telegram.

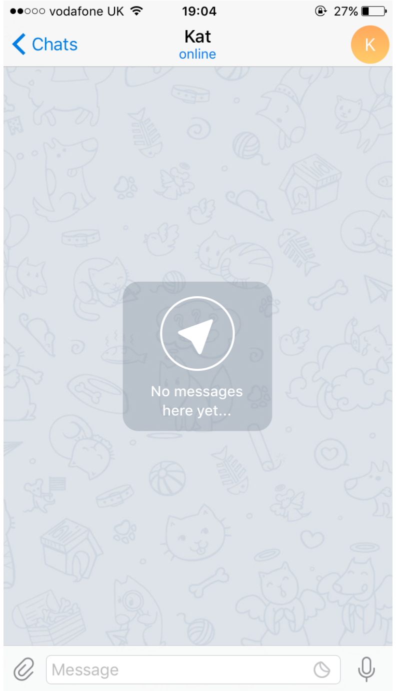
The security properties of the default chat mode are also not clear (Fig. 2). Once users start exchanging messages, the visible difference between being in a default chat and a *Secret Chat* is minimal. As shown in Figures 2 and 3, there is a small padlock symbol to the left of the contact's name in a *Secret Chat* session. This also violates Nielsen's criterion of *visibility of system status* [32], as described above.

To start a *Secret Chat* with someone, the contact has to be online for the two participating devices to exchange keys (Fig. 5, Appendix). This might push the user to send messages within the default chat mode if their contact is not online. The *Secret Chat* mode also does not support creation of a group chat, which can further undermine the mode's usefulness.

When the user taps on the contact's name within the *Secret Chat* mode, a screen entitled "Encryption Key" is presented (Fig. 4). Displayed is a square image containing smaller squares of different shades of blue, and four lines of characters and numbers underneath. The explanation found below this states that the image and text were derived from the encryption key for the *Secret Chat* with 'Kat'. In this case, if the representations are the same on the sender's and Kat's devices, "end-to-end encryption is guaranteed".

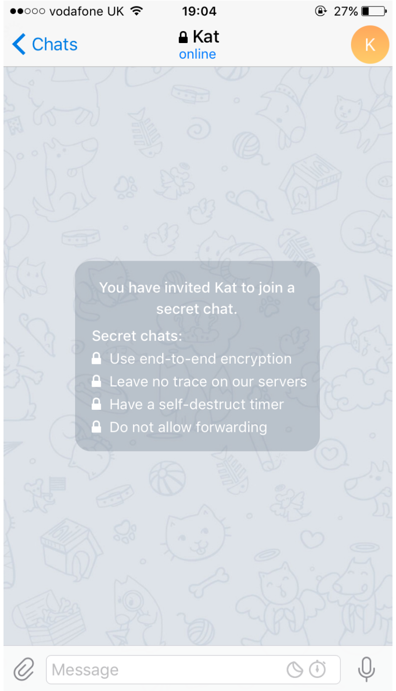
The tool uses the term "Encryption Key" to describe a verification fingerprint, which could lead users to think that the fingerprint is the encryption key used to encrypt messages. When a *Secret Chat* session with a contact is initiated, an image and a textual representation are generated on both the sender and recipient's devices, visualising their public-key fingerprints. The two communicating parties can verify each other's identities by comparing both representations through a trusted channel (e.g., in person). If the representations match, the session is secure, and no man-in-the-middle attack has occurred.

Finally, under the "Privacy and Security" settings, users can set up a passcode (4-digit PIN) to lock the application. However, we found that when Telegram is locked, users



No messages here yet...

Fig. 2. Default chat mode.

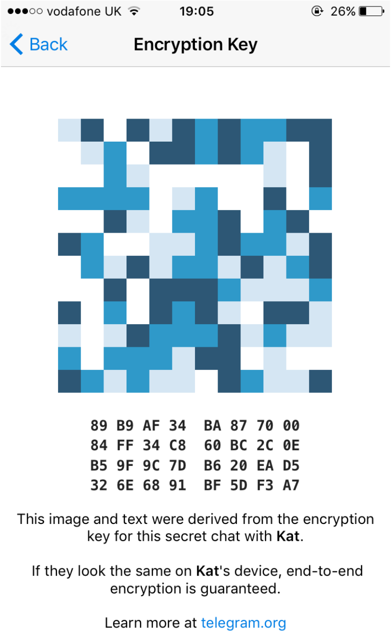


You have invited Kat to join a secret chat.

Secret chats:

- Use end-to-end encryption
- Leave no trace on our servers
- Have a self-destruct timer
- Do not allow forwarding

Fig. 3. *Secret Chat* mode.



```
89 B9 AF 34  BA 87 70 00
84 FF 34 C8  60 BC 2C 0E
B5 9F 9C 7D  B6 20 EA D5
32 6E 68 91  BF 5D F3 A7
```

This image and text were derived from the encryption key for this secret chat with **Kat**.

If they look the same on **Kat**'s device, end-to-end encryption is guaranteed.

Learn more at telegram.org

Fig. 4. A verification fingerprint.

can receive notifications about messages without the sender's name and text. Another usability issue related to locking the application is that users can either enable "auto-lock" after a specified period of inactivity, or *manually* lock the application from the screen, which is not explained to the user at the time of enabling the lock (Fig. 6 and 7).

We found Telegram was in violation of at least four of the principles of *privacy by design*. The second principle states that privacy should be the default setting, whereas in Telegram the user is guided towards the default chat mode, which is less secure and less private. This finding also violates the sixth principle, which stresses visibility and transparency. If the user chooses the *Secret Chat* mode, they will not enjoy full functionality (principle 4) because the other party has to be online, and group chats are not possible. The default chat mode does not offer "end-to-end protection" (principle 5) because it does not support E2E encryption.

VII. Discussion

Visible security can increase trust in a security system (e.g., [53]); our results indicate that making some security features visible, *while at the same time* leaving others as invisible, can create confusion for users. Many participants believed that both modes offered the same security properties, except for the self-destruct timer which was regarded as the most visible feature of the *Secret Chat* mode (and as such an indicator of that mode's level of security). Having two clearly distinct chat modes, and more so, the *less secure mode as the default*, can lead to confusion and error. Design principles, such as those proposed by Nielsen and Molich [45], are still not being followed in the design of communication tools.

Similarly, one model of (visible) security cannot be assumed to resonate with all users of a secure communication tool. Engagement with users exposed many different explanations and metaphors, even across our relatively small cohort of participants. Participants routinely compared Telegram to other communication tools they have used. This can have methodological implications, potentially requiring that assessing a tool happens alongside other tools that a user is familiar with or can contrast use against (e.g., side-by-side comparison has been seen to elicit different perspectives for CAPTCHA alternatives [37]).

The study of novices and users was invaluable: SUS scores indicated high usability across both groups, whereas engagement with prior users found that most had stopped using Telegram (because in most cases their contacts were using another messaging tool). In this sense, messaging is a naturally social practice, where peer influence can govern the adoption and use of encrypted communication tools [17], [18]. PU04 explained that the security of their communications also lied in the hands of other people: *"I worry about people being able to access it sort of semi-legitimately... the fact that someone I send a message to loses their phone on a drunken night out and someone gets access to it."*

A. Limitations

The design of the task-based scenario did not involve a real risk for participants, where an element of personal risk in the task may motivate participants to give consideration to more secure behaviours [38]. Here, the study tasks were considered more as an opportunity for participants to learn about the tool and promote further discussion of available features. Prior users may have made a choice between the default chat and the *Secret Chat* modes out of awareness or incidentally – in all cases, the choice was explored in the post-study interview. That Telegram's messaging modes could be referred to separately was used to support this discussion.

Results showed that participants described encryption in different ways. Researchers had to take care not to introduce jargon into the conversation with participants before they had the chance to frame concepts in their words (where participants' own perceptions of encryption proved interesting, as in Section V-B3).

VIII. Conclusions

We applied two evaluation methods to study Telegram; a lab-based user study (11 novices and 11 users of Telegram) and a hybrid analytical approach combining cognitive walkthrough and heuristic evaluation. Both evaluation methods found that the distinct chat modes, rather than promoting choice, have the potential to create confusion for users. Individuals likely choose a messaging tool based on how many of their regular contacts use it, rather than specific security features. Additionally, they may rarely or never share sensitive content with others, or do so through some other method (e.g., a phone call). Results demonstrated that people can have unique experiences and mental models of security and secure communications. Telegram is not necessarily a good fit; the tool may serve as a 'security blanket', creating an indistinct sense of security and limiting opportunity for individuals to grow as security-minded users. Future work could adapt the approach of recruiting novices and non-novices to study other communication tools and security-related features.

We make the following recommendations: (1) designers, developers, practitioners and researchers should apply usability inspection methods in a *consistent* fashion, following established design principles. Our findings would suggest that this is not being done; (2) secure communication capabilities ought to be developed in a way that does not limit utility or make implicit assumptions about how tools will be used (especially for the sharing of sensitive information or the preservation of privacy); (3) messaging tools that incorporate security features should follow a strategy for including users of varying levels of security expertise.

Acknowledgements

We would like to thank the reviewers for their useful comments and suggestions. We are also thankful to Jonathan M. Spring for his comments on an earlier draft of this paper. Ruba Abu-Salma's work was supported by a SUDS (Supporting Usability and Design in Security) Fellowship from the Open Technology Fund and Simply Secure. Kat Krol's work was supported by a Secure Usability Fellowship from the Open Technology Fund and Simply Secure. Participant incentives were funded by the Engineering and Physical Sciences Research Council (EPSRC) – grant EP/G037264/1. Interview transcription was funded through a Google Research Award.

References

[1] GPG4WiN. https://www.gpg4win.org/. Accessed on: 11.02.2017.

[2] GPGTools. https://gpgtools.org/. Accessed on: 11.02.2017.

[3] Off-the-Record Messaging. https://otr.cypherpunks.ca/. Accessed on: 11.02.2017.

[4] Open Whisper Systems: Signal. https://whispersystems.org/blog/signal/. Accessed on: 11.02.2017.

[5] The OpenPGP Alliance Home Page. http://www.openpgp.org/resources/downloads.shtml. Accessed on: 20.02.2017.

[6] R. Abu-Salma, M. A. Sasse, J. Bonneau, A. Danilova, A. Naiakshina, and M. Smith. Obstacles to the Adoption of Secure Communication Tools. In *IEEE Symposium on Security and Privacy (S&P)*, 2017.

[7] C. Alexander and I. Goldberg. Improved User Authentication in Off-the-Record Messaging. In *ACM Workshop on Privacy in the Electronic Society (WPES)*, pages 41–47, 2007.

[8] W. Bai, D. Kim, M. Namara, Y. Qian, P. G. Kelley, and M. L. Mazurek. An Inconvenient Trust: User Attitudes toward Security and Usability Tradeoffs for Key-Directory Encryption Systems. In *Symposium on Usable Privacy and Security (SOUPS)*, pages 113–130, 2016.

[9] A. Bangor, P. T. Kortum, and J. T. Miller. An Empirical Evaluation of the System Usability Scale. In *International Journal of Human-Computer Interaction*, volume 24, pages 574–594. Taylor & Francis, 2008.

[10] J. Bian, R. Seker, and U. Topaloglu. Off-the-Record Instant Messaging for Group Conversation. In *IEEE International Conference on Information Reuse and Integration (IRI)*, pages 79–84, 2007.

[11] N. Borisov, I. Goldberg, and E. Brewer. Off-the-Record Communication, or, Why Not To Use PGP. In *ACM Workshop on Privacy in the Electronic Society (WPES)*, pages 77–84, 2004.

[12] V. Braun and V. Clarke. Using Thematic Analysis in Psychology. In *Qualitative Research in Psychology*, volume 3, pages 77–101. Taylor & Francis, 2006.

[13] J. Brooke. SUS – A Quick and Dirty Usability Scale. In *Usability Evaluation in Industry*, volume 189, pages 4–7, 1996.

[14] J. M. Carroll and M. B. Rosson. *Paradox of the Active User*. The MIT Press, 1987.

[15] A. Cavoukian. Privacy by Design. In *IEEE Technology and Society Magazine*, volume 31, pages 18–19, 2012.

[16] G. Cluley. WhatsApp Doesn't Properly Erase Your Deleted Messages, Researcher Reveals. https://www.hotforsecurity.com/blog/whatsapp-doesnt-properly-erase-your-deleted-messages-researcher-reveals-16169.html. Accessed on: 12.02.2017.

[17] S. Das, T. H.-J. Kim, L. A. Dabbish, and J. I. Hong. The Effect of Social Influence on Security Sensitivity. In *Symposium on Usable Privacy and Security (SOUPS)*, volume 14, 2014.

[18] A. De Luca, S. Das, M. Ortlieb, I. Ion, and B. Laurie. Expert and Non-Expert Attitudes towards (Secure) Instant Messaging. In *Symposium on Usable Privacy and Security (SOUPS)*, 2016.

[19] H. Desurvire, J. Kondziela, and M. E. Atwood. What Is Gained and Lost When Using Methods Other Than Empirical Testing. In *Conference on Human Factors and Computing Systems (CHI)*, pages 125–126, 1992.

[20] W. Diffie and M. Hellman. New Directions in Cryptography. In *IEEE Transactions on Information Theory*, volume 22, pages 644–654, 1976.

[21] P. Dourish, R. E. Grinter, J. D. De La Flor, and M. Joseph. Security in the Wild: User Strategies for Managing Security as an Everyday, Practical Problem. In *Personal and Ubiquitous Computing*, volume 8, pages 391–401, 2004.

[22] Electronic Frontier Foundation (EFF). Secure Messaging Scorecard. https://www.eff.org/secure-messaging-scorecard. Accessed on: 09.09.2015.

[23] G. Fischer. Supporting Learning on Demand with Design Environments. In *International Conference on the Learning Sciences*, volume 199, pages 165–172, 1991.

[24] S. L. Garfinkel and R. C. Miller. Johnny 2: A User Test of Key Continuity Management with S/MIME and Outlook Express. In *Symposium on Usable Privacy and Security (SOUPS)*, pages 13–24, 2005.

[25] S. Gaw, E. W. Felten, and P. Fernandez-Kelly. Secrecy, Flagging, and Paranoia: Adoption Criteria in Encrypted E-mail. In *ACM Conference on Human Factors in Computing Systems (CHI)*, pages 591–600, 2006.

[26] R. Geambasu, T. Kohno, A. A. Levy, and H. M. Levy. Vanish: Increasing Data Privacy with Self-Destructing Data. In *USENIX Security Symposium*, pages 299–316, 2009.

[27] I. Goldberg, B. Ustaoğlu, M. D. Van Gundy, and H. Chen. Multi-party Off-the-Record Messaging. In *ACM Conference on Computer and Communications Security (CCS)*, pages 358–368, 2009.

[28] P. Gutmann. PKI: It's Not Dead, Just Resting. *Computer*, 35(8):41–49, 2002.

[29] R. H. Bernard. *Non-probability Sampling: Social Research Methods: Qualitative and Quantitative Approaches*. SAGE, 2006.

[30] A. Hern. Facebook 'Pauses' WhatsApp Data Sharing after ICO Intervention. *The Guardian*, https://www.theguardian.com/technology/2016/nov/08/facebook-pauses-whatsapp-data-sharing-after-ico-intervention, 2016. Accessed on: 01.03.2017.

[31] A. Herzberg and H. Leibowitz. Can Johnny Finally Encrypt? Evaluating E2E Encryption in Popular IM Applications. In *ACM Workshop on Socio-Technical Aspects in Security and Trust (STAST)*, 2016.

[32] T. Hollingsed and D. G. Novick. Usability Inspection Methods after 15 Years of Research and Practice. In *ACM International Conference on Design of Communication*, pages 249–255, 2007.

[33] R. Housley, W. Polk, W. Ford, and D. Solo. Internet X. 509 Public-key Infrastructure Certificate and Certificate Revocation List (CRL) Profile. Technical report, 2002.

[34] J. Jakobsen and C. Orlandi. On the CCA (in)security of MTProto. In *ACM Workshop on Security and Privacy in Smartphones and Mobile Devices (SPSM)*, pages 113–116, 2016.

[35] R. Jeffries, J. R. Miller, C. Wharton, and K. Uyeda. User Interface Evaluation in the Real World: A Comparison of Four Techniques. In *Conference on Human Factors in Computing Systems (CHI)*, pages 119–124, 1991.

[36] C.-M. Karat, R. Campbell, and T. Fiegel. Comparison of Empirical Testing and Walkthrough Methods in User Interface Evaluation. In *Conference on Human Factors in Computing Systems (CHI)*, pages 397–404, 1992.

[37] K. Krol, S. Parkin, and M. A. Sasse. Better the Devil You Know: A User Study of Two CAPTCHAs and a Possible Replacement Technology. In *NDSS Workshop on Usable Security (USEC)*, 2016.

[38] K. Krol, J. M. Spring, S. Parkin, and M. A. Sasse. Towards Robust Experimental Design for User Studies in Security and Privacy. In *Workshop on Learning from Authoritative Security Experiment Results (LASER)*, 2016.

[39] C. Lewis, P. G. Polson, C. Wharton, and J. Rieman. Testing a Walkthrough Methodology for Theory-Based Design of Walk-Up-and-Use Interfaces. In *Conference on Human Factors in Computing Systems (CHI)*, pages 235–242, 1990.

[40] C. Lewis and J. Rieman. *Task-Centered User Interface Design: A Practical Introduction*. 1993.

[41] H. Liu, E. Y. Vasserman, and N. Hopper. Improved Group Off-the-Record Messaging. In *ACM Workshop on Privacy in the Electronic Society (WPES)*, pages 249–254, 2013.

[42] M. Melara, A. Blankstein, J. Bonneau, M. Freedman, and E. Felten. CONIKS: Bringing Key Transparency to End Users. In *USENIX Security Symposium*, 2015.

[43] J. Nielsen. *Usability Engineering*. Elsevier, 1994.

[44] J. Nielsen. Usability Inspection Methods. In *ACM Conference Companion on Human Factors in Computing Systems (CHI)*, pages 413–414, 1994.

[45] J. Nielsen and R. Molich. Heuristic Evaluation of User Interfaces. In *ACM Conference on Human Factors in Computing Systems (CHI)*, pages 249–256, 1990.

[46] S. Parkin and K. Krol. Appropriation of Security Technologies in the Workplace. In *Workshop on Experiences of Technology Appropriation: Unanticipated Users, Usage, Circumstances, and Design*, 2015.

[47] R. Perlman. The Ephemerizer: Making Data Disappear. *Sun Microsystems*, 2005.

[48] P. G. Polson, C. Lewis, J. Rieman, and C. Wharton. Cognitive Walkthroughs: A Method for Theory-Based Evaluation of User Interfaces. In *International Journal of Man-Machine Studies*, volume 36, pages 741–773, 1992.

[49] J. Reardon, D. Basin, and S. Čapkun. SoK: Secure Data Deletion. In *IEEE Symposium on Security and Privacy (S&P)*, pages 301–315, 2013.

[50] K. Renaud, M. Volkamer, and A. Renkema-Padmos. Why Doesn't Jane Protect Her Privacy? In *Privacy Enhancing Technologies Symposium (PETs)*, pages 244–262, 2014.

[51] F. Roesner, B. T. Gill, and T. Kohno. Sex, Lies, or Kittens? Investigating the Use of Snapchat's Self-destructing Messages. In *Financial Cryptography and Data Security*, pages 64–76. 2014.

[52] S. Ruoti, J. Andersen, S. Heidbrink, M. O'Neill, E. Vaziripour, J. Wu, D. Zappala, and K. Seamons. "We're on the Same Page": A Usability Study of Secure Email Using Pairs of Novice Users. In *ACM Conference on Human Factors and Computing Systems (CCS)*, 2016.

[53] S. Ruoti, N. Kim, B. Burgon, T. van der Horst, and K. Seamons. Confused Johnny: When Automatic Encryption Leads to Confusion and Mistakes. In *Symposium on Usable Privacy and Security (SOUPS)*, page 5, 2013.

[54] M. D. Ryan. Enhanced Certificate Transparency and End-to-End Encrypted Mail. In *Network and Distributed System Security Symposium (NDSS)*, 2014.

[55] A. Sears. Heuristic Walkthroughs: Finding the Problems Without the Noise. In *International Journal of Human-Computer Interaction*, volume 9, pages 213–234, 1997.

[56] B. Shackel. Human Factors and Usability. In *Human-Computer Interaction*, pages 27–41, 1990.

[57] R. Stedman, K. Yoshida, and I. Goldberg. A User Study of Off-the-Record Messaging. In *Symposium on Usable Privacy and Security (SOUPS)*, pages 95–104, 2008.

[58] The Grugq. Underground Tradecraft. https://grugq.tumblr.com/post/133453305233/operational-telegram. Accessed on: 09.01.2017.

[59] T. S. Tullis and J. N. Stetson. A Comparison of Questionnaires for Assessing Website Usability. In *Usability Professional Association Conference*, pages 1–12, 2004.

[60] N. Unger, S. Dechand, J. Bonneau, S. Fahl, H. Perl, I. Goldberg, and M. Smith. SoK: Secure Messaging. In *IEEE Symposium on Security and Privacy (S&P)*, pages 232–249, 2015.

[61] C. Wharton, J. Rieman, C. Lewis, and P. Polson. The Cognitive Walkthrough Method: A Practitioner's Guide. In *Usability Inspection Methods*, pages 105–140, 1994.

[62] A. Whitten and J. D. Tygar. Why Johnny Can't Encrypt: A Usability Evaluation of PGP 5.0. In *USENIX Security Symposium*, 1999.

[63] P. R. Zimmermann. *The Official PGP User's Guide*. 1995.

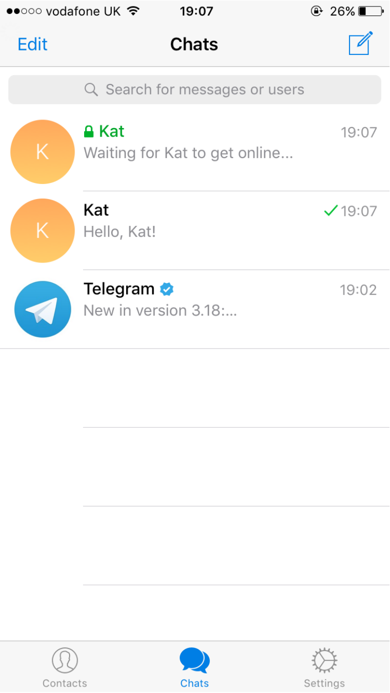
Appendix

A. *SUS scores*

Novices		Users	
Identifier	**SUS Score**	**Identifier**	**SUS Score**
PN01	82.5	PU01	92.5
PN02	97.5	PU02	100
PN03	75.0	PU03	92.5
PN04	95.0	PU04	75.0
PN05	70.0	PU05	87.5
PN06	77.5	PU06	67.5
PN07	70.0	PU07	77.5
PN08	82.5	PU08	67.5
PN09	97.5	PU09	80.0
PN10	82.5	PU10	85.0
PN11	87.5	PU11	75.0
Mean	83.4	**Mean**	81.8

TABLE I. SUS SCORES GIVEN BY NOVICES AND USERS OF TELEGRAM PARTICIPATING IN THE STUDY.

B. *Additional figures*


●●○○○ vodafone UK 19:07 26%

Chats

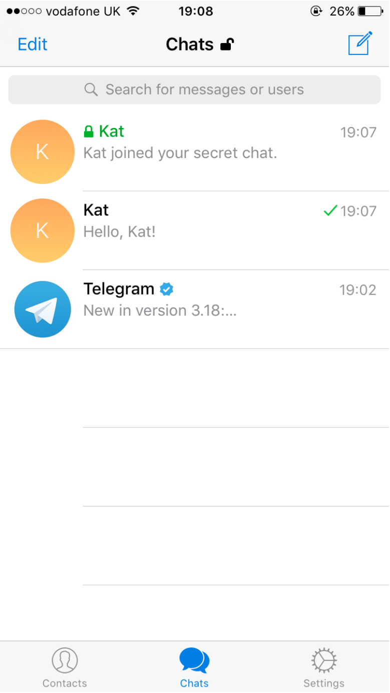
Search for messages or users

🔒 Kat 19:07
Waiting for Kat to get online...

Kat ✓19:07
Hello, Kat!

Telegram 19:02
New in version 3.18:...

Fig. 5. Starting a *Secret Chat* with a contact.


●●○○○ vodafone UK 19:08 26%

Chats

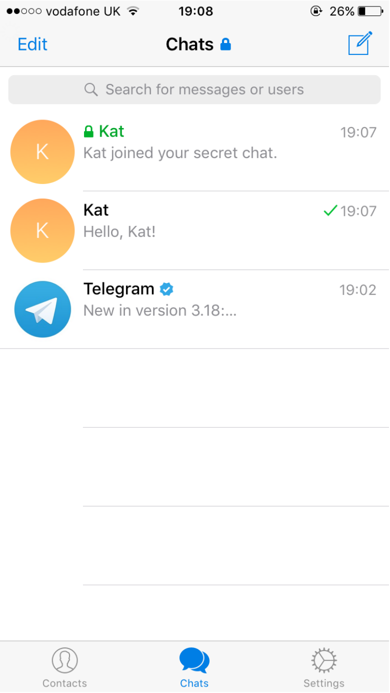
Search for messages or users

🔒 Kat 19:07
Kat joined your secret chat.

Kat ✓19:07
Hello, Kat!

Telegram 19:02
New in version 3.18:...

Fig. 6. Unlocked padlock.


●●○○○ vodafone UK 19:08 26%

Chats 🔒

Search for messages or users

🔒 Kat 19:07
Kat joined your secret chat.

Kat ✓19:07
Hello, Kat!

Telegram 19:02
New in version 3.18:...

Fig. 7. Locking Telegram by closing the padlock.