PGP and S/MIME in the Age of Multi-Device Email Usage: "Admittedly, an outrageously confusing system."

Katharina Schiller
Hof University of Applied Sciences
katharina.schiller@hof-university.de

Andreas Herrmannsdörfer
Hof University of Applied Sciences
andreas.herrmannsdoerfer@hof-university.de

Zinaida Benenson
Friedrich-Alexander-Universität Erlangen-Nürnberg
zinaida.benenson@fau.de

Florian Adamsky
Hof University of Applied Sciences
florian.adamsky@hof-university.de

***Abstract*—The usability of OpenPGP and Secure/Multipurpose Internet Mail Extensions (S/MIME) has been extensively studied for decades, with persistent usability challenges identified as one of the factors behind their low adoption rates. However, the majority of users read and write their emails on multiple devices. Yet, little is known about the challenges this multi-device setup poses for the usability of email encryption. Concurrently, the widespread use of encrypted instant messaging apps has raised user expectations for seamless, secure, and intuitive communication tools across devices. In this paper, we present a qualitative think-aloud study with $N = 34$ participants to evaluate the usability of popular email clients supporting OpenPGP and S/MIME in a multi-device scenario. We show that the majority of participants experience considerable difficulty integrating key material, configuring encryption, and transferring it to mobile devices. Participants were generally able to detect invalid signatures, leading a few to report greater vigilance in managing emails. While the initial setup presented challenges, participants found that once they configured encryption and signing correctly, the process required minimal effort and was easier than anticipated.**

***Index Terms*—email security, end-to-end encryption, usability study, OpenPGP, S/MIME**

I. Introduction

While chat systems, instant messaging, and collaboration tools are popular, email remains the most widely used system for text-based communication. By the end of 2024 [1], every minute, 251.1 million emails are sent over the Internet. In comparison, there were 18.8 million text messages and 1.03 million Slack messages sent. However, since email is over 50 years old, it was designed without security, so every email is like a postcard—easily read and modified en route. The lack of security allows nation-state actors to collect our emails [2], and, with exposed email servers [3], [4], our email communications could become public. To address this issue, two End-to-End Encryption (E2EE) standards for email have been developed: OpenPGP and S/MIME. Even though PGP was introduced in 1991 [5] and S/MIME in 1998, hardly any emails are encrypted and signed. Stransky et al. [6] analyzed 27 years of email data at a large university and found that only $5.45\,\%$ used OpenPGP or S/MIME, leading to $0.05\,\%$ encrypted emails and $2.8\,\%$ signed emails. The research community has been investigating the reasons for that for decades, starting with Whitten and Tygar [7], who investigated PGP and its key challenges for end users.

However, a lot has changed since the first usability studies of PGP and S/MIME. Increasingly, people are using their computers and mobile devices, such as tablets and smartphones, to read and write emails. In our survey with 184 participants, see Section IV-A, $90.8\,\%$ reported using either a computer and smartphone, or a computer and tablet, or all three devices to read and write emails. Additionally, usability continues to evolve, and the use of instant messaging tools, e. g., WhatsApp, Signal, and Threema, that have built-in encryption and signatures, influences app design.

In this paper, we present a qualitative think-aloud [8] usability study of popular email clients with PGP and S/MIME support in a multi-device scenario. We started with a survey with $N = 184$ to find participants for the usability test and ended up with 34 participants. We divided participants into groups based on email clients: Outlook (11 with S/MIME), Apple Mail (11 with S/MIME), and Thunderbird (12 with OpenPGP). During the experiments, participants were tasked with performing several security operations: encrypting and decrypting an email, signing an email, transferring cryptographic key material to a mobile device, and verifying the authenticity of a received email. After the tasks, we conducted a follow-up interview and evaluated the usability of the desktop and mobile email clients used with the System Usability Scale (SUS) [9] and Single Ease Question (SEQ) [10].

Research Questions (RQs). We address the following RQs:

- RQ1 How usable is email encryption in popular email clients, including both desktop and mobile versions?
- RQ2 What challenges affect the usability of email encryption when used across multiple devices?

RQ3 How does the display of an email signature influence users' perception of the authenticity of the email?

Contributions. The contributions are as follows:

- We conducted a usability study with 34 participants, each interacting with popular email clients (Outlook, Apple Mail or Thunderbird) that support end-to-end email encryption, to identify usability challenges.
- We show that multi-device use further increases the difficulty of email encryption; 61.8 % of participants, regardless of the tools used, encountered major problems completing related tasks.
- We present the results of a task in which participants had to distinguish between correctly signed emails and fake ones; when the signature status was visually indicated by the system, 81.8 % accurately identified the correct email.
- We show that, after the setup challenges, 64.7 % participants were able to send a signed, encrypted email on desktop, and 88.2 % on mobile.

II. Background

Two open E2EE standards for email encryption and digital signatures are OpenPGP and S/MIME. S/MIME is more commonly used in corporate and governmental settings [11] due to its integration with Public Key Infrastructure (PKI), while OpenPGP is favored in the technical community [12].

In **OpenPGP**, a client generates an asymmetric cryptographic key pair, consisting of a private and public key, and publishes the public key, typically via a keyserver. To encrypt an email, the sender encrypts the email body with a random symmetric session key and then encrypts the symmetric key with the recipient's public key and attaches it to the email. To sign an email, the sender hashes the email body and then encrypts the hash with their private key. The recipient then uses the sender's public key to decrypt the hash, thereby verifying the integrity and authenticity of the email. OpenPGP relies on a *Web of Trust* where participants verify and sign each other's keys, eliminating the need for a central trust entity.

S/MIME uses trusted Certification Authoritys (CAs) to establish the authenticity and integrity of emails. A client generates an asymmetric cryptographic key pair, and a CA signs the public key and issues an X.509 certificate. Encryption and signing work similarly to OpenPGP at the cryptographic level. The email body is encrypted with a randomly generated symmetric key, which is then encrypted with the recipient's public key from their X.509 certificate. S/MIME signing works in a similar way; however, the signature format differs, as PKCS#7/Cryptographic Message Syntax (CMS) is used.

III. Related Work

The usability of email encryption has been studied for decades. Early studies [7], [13] demonstrated that users often struggle to correctly configure email encryption or understand its meaning. Unclear user interfaces, lack of feedback, and insufficient mental models of the underlying mechanisms lead to encryption functions being misapplied or misunderstood.

A major problem highlighted by previous studies [7], [13], [14], [15], [16], [17] is key handling. Garfinkel and Miller [13] automated key management processes through their approach known as Key Continuity Management (KCM). Compared with manual key management, this solved some usability problems. They demonstrate that users recognize digital signatures and that visual cues enhance usability, but the social engineering attacks on email authentication are still successful. Likewise, Ruoti et al. [18], [19] demonstrate that while automated key management and security processes reduce user errors, they also create opacity, as users have difficulty comprehending the cryptographic states and mechanisms. Extending this research, studies [20], [21], [22] on key management approaches show that users favor convenience over stronger security, and that automated or server-managed key management improves email encryption usability. The demand for usable key management extends beyond email to any E2EE service where credentials must be synchronized across multiple devices [23].

Atwater et al. [24] show that making the encryption more visible does not necessarily increase user trust, and that users tend to trust local, offline tools more than online services. Ruoti et al. [17] compare different systems for email encryption and show that tutorials and instructions are important for users and help them learn how to use the system more quickly. Users tend to prefer integrated solutions and are not inclined to use separate tools, although there are many new developments in this area, i. e., Tutanota, ProtonMail, or Confidante [25].

Beyond usability, another barrier to adoption of encrypted emails is users' understanding of secure communication, as many are unaware that emails are not secure by default [26], doubt that encrypted emails are safe [27], or tend to avoid encryption when they perceive it as excessive or inappropriate [28]. Although users wish for secure communication [29], the use of PGP and S/MIME remains low [6], reflecting ongoing challenges in key management and usability [30].

As mobile devices become central to digital communication, research has expanded to their context. Schiller and Adamsky [31] found that OpenPGP- and S/MIME-based email solutions on smartphones introduce additional usability challenges, including constrained user interfaces and limited support for key and certificate management. Multi-device use presents additional challenges, even without E2EE [32].

To our best knowledge, no study yet focused on the transition of cryptographic keys and the identification of correctly signed emails on both desktop and mobile.

IV. Methodology

Figure 1 shows our study design. The study was conducted at a German university. First, we conducted a survey to identify participants for the usability study. Of the 184 who filled out the survey, 34 participated in the usability study. We divided 34 participants into three groups: 11 for Outlook with S/MIME, 11 for Apple Mail with S/MIME, and 12 for Thunderbird with OpenPGP. We explain the selection of software in Section IV-B. Participants used the same application ecosystem on

both desktop and smartphone (e.g., Outlook on both devices); mixed setups (e.g., Thunderbird on desktop and Outlook on mobile) were not tested. The usability test contained multiple tasks, which we describe in more detail in Section IV-C. After the usability test, we conducted interviews with the participants and evaluated the gathered data.

A. Survey

The main objective of the survey was to recruit participants for the usability study. We were particularly interested in their email clients and the operating systems on their devices. Based on this information, we assigned participants to one of three email clients to ensure they were generally familiar with both the operating system and the email client. However, this could not be realized in all cases, as we tried to keep group sizes similar for comparability, and some applications had been used by only a small number of participants prior to the study. The survey was anonymous unless participants provided their email address and indicated interest in participating in the usability study. Independently, two researchers translated the survey into English and discussed their translations, which ultimately led to the final version. Participants could change the language on the survey's first page. Both versions of the survey, German and English, were then tested by four other researchers and their feedback was integrated. For the survey see the Extended Version [33].

B. Email Client Selection

We investigated 10 email clients (see Table 1 in the Extended Version [33]) to assess their suitability for our usability study. Email clients were required to support S/MIME or OpenPGP natively (no plugins), enable cross-device configuration, be widely available, be free of charge, and allow integration with existing email accounts via Internet Message Access Protocol (IMAP). We considered metrics, such as download counts and app store ratings, to ensure the inclusion of widely adopted, actively maintained clients. Also, we wanted a setup for desktop (Windows and Mac) and mobile (Android and iPhone) to reach a widespread population. Therefore, we chose Apple Mail (Desktop V16.0, iOS V18.6) for S/MIME, Mozilla Thunderbird (Desktop V145.0–146.0, Mobile V14.0) for OpenPGP, and Microsoft Outlook (Desktop V2510, Mobile Build v5.2545.0) for S/MIME. The other clients either had a technical issue or did not meet our requirements.

We conducted the study using an Android Xiaomi 11T (Android 14), an iPhone 11 (iOS 18.6), a MacBook Pro (M2, 2022; macOS Sonoma 14.5), and a Windows 11 system (version 25H2). All devices provided native email clients. Android devices were preconfigured with Thunderbird and OpenKeychain; iPhones with the respective Outlook version. OpenKeychain was required because Thunderbird Mobile does not provide built-in OpenPGP key management functionality, unlike the desktop version. Certain device–application combinations were not included due to technical incompatibilities or because some setups were beyond the scope of this study, e. g., Thunderbird Mobile is not available for iOS. In addition, the Thunderbird mobile app does not support S/MIME, therefore we did not include a Thunderbird group for S/MIME.

C. Scenario

The participants slip into the role of Sam Peters communicating with a friend, Robin Winter. Both are gender-neutral names to help participants emerge into the situation. From the start, Sam Peters has an initial email in the inbox that reflects the content of the corresponding task card. Robin Winter has discovered S/MIME or OpenPGP (depending on the test group) and is now planning a birthday party for another friend. With this in mind, Robin Winter asks Sam Peters whether he/she would like to join the party and set up email encryption for further communication. In our fictional scenario, Robin Winter shares the fingerprint of their public key through a secure communication channel (Signal Messenger). We printed the fingerprint on the task card, and participants can verify it before importing Robin's public key. After that, the participant should accept this invitation on behalf of Sam Peters. Robin Winter then sends a PayPal link to collect money for a gift. In the end, participants should write another encrypted and digitally signed email on their smartphones and indicate in the reply how they can tell it is encrypted.

a) Fake Emails: For RQ3, we aim to investigate whether users perceive and understand how different email clients communicate cryptographic signatures, and whether they can verify their validity. Therefore, we include two fake emails, simulated as sent by an adversary and injected into the inbox by the study coordinator (see Fig. 4 in Extended Version [33]. This setup allows us to evaluate signature perception in a controlled and ethically safe manner, while keeping email content and difficulty identical. Contrary to Garfinkel and Miller [13], we used emails with identical content to examine the signature in isolation. Recall that Robin Winter collects money for a birthday present by sending a link to a PayPal pool. Three nearly identical emails appeared in the participant's inbox, each with a different PayPal link. The PayPal link itself does not indicate validity. The legitimate email is sent by the study coordinator from Robin Winter's email account and is correctly signed. The first fake email had no digital signature, but we added a lock icon at the top of the email body with the following text next to it: "*This email is digitally signed by robin.winter@[domain].de*". In the second email, we changed the PayPal link in the email body, which invalidated the signature.

D. Test Environment

We conducted the usability tests in the department's office. The participants attended in one-hour time slots. Between each participant, we scheduled a 30-minute break to reorganize the setup and reset all devices and files to the initial settings of our study. That includes deleting all keys and certificates from the email clients and the devices' certificate stores, clearing the browser history, deleting all emails from the inbox, and resending the first email, adjusting the timestamps of the fake emails. We provided participants with separate test devices,

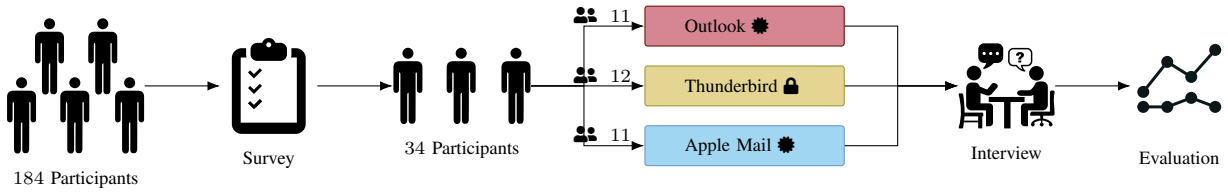


Fig. 1: Study design: We surveyed 184 participants to understand email usage better and identify candidates for our usability study. Then we divided the 34 into three groups: 11 for Outlook using S/MIME, 11 for Apple Mail using S/MIME, and 12 for Thunderbird using OpenPGP. After the usability test, we did a final interview and evaluated the data.

depending on the test group: either a MacBook, a Windows computer, an iPhone, or an Android smartphone, along with a USB cable for those who preferred wired data transfer. At the beginning, we presented each participant with the consent form and asked them to read it carefully and sign if they agreed. We reminded them of the legal requirements and that we would be recording the screen and audio, for which we used OBS.[1] For recording smartphone screens, we used the device's built-in screen recording feature. During the usability tests, a study coordinator sat next to the participants to monitor the screens. The second study coordinator sent Robin Winter's emails and verified that the participants' emails were correctly encrypted and signed. Similarly, the second coordinator uploads fake emails to the participant's inbox during the usability tests.

E. Participants and Recruitment

We emailed various university mailing lists (which we are permitted to under GDPR) to recruit participants. In the email, we included a link to the survey and a note that, after completing the survey, participants could enter their email address to register for the usability test. Participants in the usability test received a goodie bag and joined a raffle for one out of three Amazon gift cards worth EUR 25. In addition, students received bonus points in our classes for participating in the study. After two weeks, we sent out a reminder email. Furthermore, we promoted our usability tests in onboarding training courses for employees and in lectures.

A total of 184 participants completed the survey, of whom 121 expressed interest in participating in the usability test. We invited participants with details about the usability tests, including location, duration, compensation, and a booking link. One day prior to the booked time slot, everyone received a reminder email. In the end, 40 participants took part in the usability tests, of whom 34 were included in the evaluation. We had to exclude 6 participants due to technical issues related to study setup, such as improper device resets and leftover keys from previous tests.

The majority of participants are male (28). Although the majority of participants were up to 30 years old (28), the sample also included one participant over 60 years old and several in their 30s and 40s. The majority stated that they had a background in IT (30) and a high level of IT affinity (25). The interviews revealed that four participants regularly use encryption, two of them for personal communications (OpenPGP) and the other two at work (S/MIME). Another three participants tested it at least once, e. g., as part of a class, successfully or unsuccessfully. Most participants had heard of email encryption but had not yet used it (18), some participants said they had never heard of it or were unsure (9). See Tab. 2 in the Extended Version [33] for an overview of the participants. We discuss the limitations of the sample in Section VI-C.

F. Usability Metrics

We used SEQ and SUS as usability metrics in the study. The SEQ is a short usability measure [10] used right after a finished task. Participants can rate the difficulty of the task with a scale from "*very difficult (1)*" to "*very easy (7)*", which can be seen in the Extended Version [33]. The advantage of this metric is that it does not interrupt the study too much. We used SEQ for setting up OpenPGP and S/MIME.

The SUS is a questionnaire [9] for assessing the perceived usability of a system which should be applied to tasks intended to be repeated. It contains 10 questions rated on a 5-point Likert scale, ranging from "*strongly disagree*" to "*strongly agree*" (see the Extended Version [33]). We asked participants to complete the SUS questionnaire after completing the main tasks, keeping in mind that it focused on everyday use.

G. Main Tasks of the Usability Study

Figure 2 shows all tasks each participant should complete. We denote task i with T_i; for a specific task in OpenPGP, we use 🔒_i; and for S/MIME, we use ✹_i. Participants assigned to Apple Mail or Outlook completed the S/MIME (✹_i) tasks, whereas participants assigned to Mozilla Thunderbird completed the OpenPGP (🔒_i) tasks. Participants were allowed to utilize additional resources, including search engines and AI tools. The study coordinator read the task card aloud, and participants could ask questions if something was unclear. They received hints *only* when they were stuck and could choose to attempt the task again. At times, more support was necessary, and we highlight these instances in Section V-C. The other study coordinator set up things in the background, e. g., sending the emails and uploading the fake emails. All instructions and tasks for the usability tests are in the Extended Version [33].

1) Desktop Application Tasks (T_1–T_5): The first task ✹_1 was to obtain a certificate. Participants should search online for guidelines on obtaining an S/MIME certificate for personal

[1] https://obsproject.com/

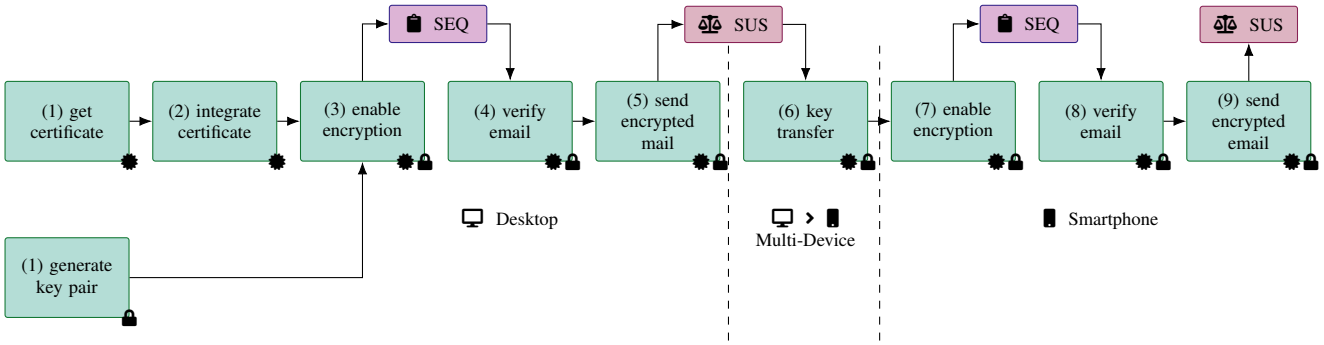


Fig. 2: We denoted the different tasks for the usability tests in general with T_i and the corresponding number, for OpenPGP with $\blacksquare_i$ and for S/MIME with $\bullet_i$. Tasks T_1–T_5 were on the desktop pc, task T_7 contains our multi-device usage, and tasks T_8–T_{10} were on the smartphone. Tasks $\bullet_1$–$\bullet_2$ were only for S/MIME and task $\blacksquare_1$ was only for OpenPGP. Starting with tasks T_3–T_9, these tasks had to be done for both OpenPGP and S/MIME. Between task T_3 and T_4 and between T_7 and T_8, we had a SEQ. Between task T_5 and T_6 and after the final task T_9 we had a SUS.

use. We set a rough time frame of 10 min, whereby participants could let us know if they finished earlier. After that, we showed that we prepared a certificate for the experiment. The next task was $\bullet_2$, integrating this certificate for Sam Peters' account, followed by $\bullet_3$, enabling encryption in the desktop application. The first task $\blacksquare_1$ was to generate a new key pair, followed by $\blacksquare_3$ and enabling encryption in the email client. After these initial tasks, participants received the SEQ to evaluate their experience with the setup process.

Task T_4 was to open the email from Robin Winter and realize that it is digitally signed. For task $\blacksquare_4$, the public key was attached to this email. We added the public key's fingerprint to the participants' task card. The study coordinator mentioned that Robin Winter shared the fingerprint through another secure channel. Participants should, in task T_4, verify that the email was indeed sent by Robin Winter, thereby confirming its authenticity. If participants did not check the fingerprint on purpose, the study coordinator would intervene *only* if they were confused. After that, participants should accept the invitation in task T_5 by replying to Robin Winter's email with a digitally signed and encrypted message. For task $\blacksquare_5$, the study coordinator monitored the participant's actions to ensure that Robin Winter received the public key. Thunderbird automatically attaches the public key, so nothing is needed there. After completing the tasks, we asked participants to fill out the SUS for the desktop applications, focusing on everyday use of encryption and digital signatures.

2) Transfer from Desktop to Smartphone (T_6): In task T_6, participants were asked to set up email encryption on the smartphone using a key pair ($\blacksquare_6$) or certificate ($\bullet_6$) in the email app. To complete task $\blacksquare_6$ in Thunderbird, participants needed to export the key pair and then transfer it to the smartphone. They also needed to add the email account to the smartphone app. Regarding task $\bullet_6$ in Outlook and Apple Mail, the participants could transfer the certificate directly. Outlook offers two options: either send the public key via email or send it automatically via the M365 cloud. The latter option was not available in our setup. In Apple Mail, participants could use standard file transfer options (AirDrop or cable).

3) Smartphone App Tasks (T_7–T_9): After the participant had transferred the key pair or certificate, the next task T_7 was to enable encryption on the smartphone. Again, all participants were asked to answer the SEQ regarding the transmission of the key pair or certificate to the smartphone and the setup process for email encryption on the smartphone. Then the study coordinator sent the two fake emails (see the Extended Version [33]) and the real one in random order from Robin Winters' account. In task T_8, participants were asked to identify which emails were legitimate and which were fake, and to justify their decisions. The relevant information came from the digital signature, which was corrupted in one email, mimicked using HTML in another, and was correct in a third. Finally, the participants should send a new, encrypted, digitally signed email to Robin Winter in task T_9 from the smartphone. After that, they received a final reply from Robin Winter. Participants were asked how they could tell whether an email is encrypted. Lastly, participants again completed the SUS with a focus on encryption and digital signatures on the smartphone.

4) Pre-Study: To ensure that the instructions and tasks were appropriate, we conducted a preliminary study with five lab members. We assessed the clarity of the instructions, the time required, the workflow efficiency, and any inaccuracies in the information. Based on the results, we revised the instructions, e. g., clarifying that the transferred certificate or key pair must match the one on the desktop device.

H. Post-Study Interviews

Following the usability tests, we conducted semi-structured interviews that covered several key aspects of participants' experiences. These included the usability and comprehensibility of the encryption tools, their perceptions of trust and security, and their understanding of authenticity and key management processes. For OpenPGP, we discussed the role and functionality of the key server, while for S/MIME, we focused on the

use and handling of certificates. We also asked participants whether they had prior experience with other forms of email encryption and invited them to compare those solutions with the ones they tested. In addition, we addressed observations from the usability tests, allowing participants to revisit the applications and provide their feedback directly within the user interfaces. This allowed us to clarify any open questions or uncertainties that had arisen during the earlier testing phase.

I. Ethical Considerations

Following the Menlo Report [34], we rated the study as minimal risk for all stakeholders (participants, researchers, and email providers). Participation in the survey was anonymous. We used the survey tool LamaPoll[2], which ensures GDPR-compliant data processing and stores data on servers located in Germany. To participate in the survey, participants had to be at least 18 years old. They could discontinue the survey at any time or withdraw their responses. At the end of the survey, participants could voluntarily provide an email address for invitations to the usability tests. We only used the email addresses for communication and the raffle. All participants were assigned an ID that does not allow any conclusions about the person, but only about the tested application.

Before the lab study, participants were informed about data protection requirements, the recordings of screens and audio, and signed a consent form. Participants could withdraw at any time or decline to participate. For the usability tests, we provided devices and test accounts, so usage of personal devices or accounts was not needed. The audio and screen recordings were recorded and stored locally and shared between the researchers via the university's internal NextCloud. For transcription, we used an offline version of OpenAI's Whisper and manually refined it. The fake emails were not sent but only added to the inbox, and did not link to any fraudulent website. Our study did not require ethics approval, as it is only necessary for medical research at our university.

J. Data Analysis

We analyzed the quantitative data from the survey and usability questionnaires using Python. The screen recordings from the desktop device and smartphone were edited side by side so that they could be watched simultaneously. We documented any observed results that were not expressed through the think-aloud method [8] or that contradicted participants' verbalizations in the transcripts in brackets. We used qualitative content analysis by Mayring [35] to evaluate the interviews and usability tests, and MaxQDA for coding. First, two researchers independently coded the same three transcripts and created an initial codebook. Both then discussed the codebook and created a revised codebook. During the coding process, the researchers repeatedly discussed and adjusted the codebook. Subsequently, one of the researchers coded all transcripts using the final codebook, see the Extended Version [33].

[2]https://www.lamapoll.de/

V. Results

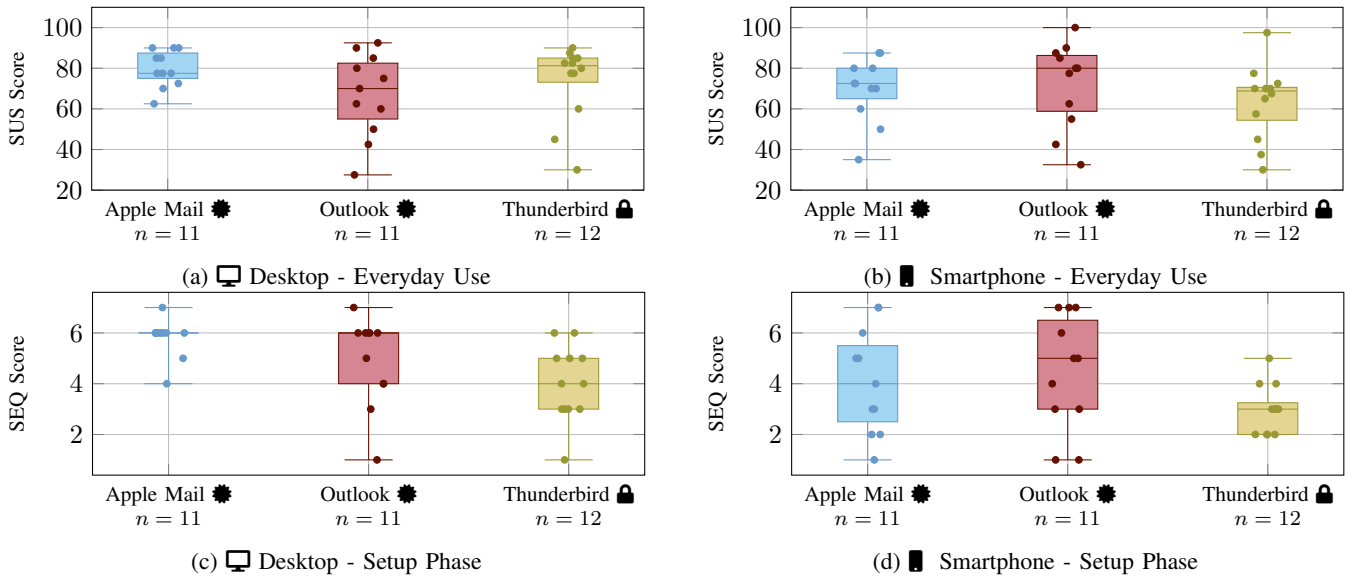
A. Survey Results

We analyze the answers from 184 participants. Only 17 participants (9.2 %) reported using email on a single device, whereas 167 participants (90.8 %) used multiple devices. The most common combination was smartphone and laptop ($n = 106$), followed by smartphone, laptop, and tablet ($n = 60$). Nearly half of the participants ($n = 83$, 45.1 %) reported using email encryption before, with 29.9 % stating that they use it occasionally and 15.2 % stating that they use it regularly. Of these participants, most (55.4 %) had a positive experience, followed by mixed (38.6 %) and negative experiences (6.0 %). Participants wrote about mixed or negative experiences in the free-text field, mainly due to the cumbersome process and technical problems. Most participants stated that it is rather important (48.4 %) or very important (31.0 %) for them to keep their emails protected from third parties. A total of 61.4 % of participants indicated they would be willing to put in additional effort for this purpose. Among these participants, as well as those who were undecided, the majority (69.6 %) preferred to invest less than 15 min.

B. Questionnaire Results

Using the SEQ, participants evaluated the tasks for integrating certificates and setting up the encryption functions, including the transfer of the certificate or key pair. Figure 3 depicts the results for all tested email clients for desktop and smartphones. Overall, the desktop versions achieved better results, with S/MIME performing better than OpenPGP (Apple Mail: SEQ=5.8, SD=0.8, Outlook: SEQ=4.9, SD=1.8, Thunderbird: SEQ=4.0, SD=1.5). These results show that participants found setting up S/MIME to be very easy or easy, whereas setting up OpenPGP was considered difficult, as reported in [36]. Transferring key material to the smartphone and setting up encryption options is generally considered more difficult. All applications received a difficult rating (Apple Mail: SEQ=4.1, SD=2.1, Outlook: SEQ=4.5, SD=2.3), whereby Thunderbird is rated notably more difficult (SEQ=3.0, SD=1.0). The observational results show that these difficulties stemmed primarily from users' inability to export the private key, as described in more detail in Section V-C7.

With the SUS questionnaire, we encouraged participants to focus on everyday use, particularly on encryption and signatures. Figure 3 depicts the results for all tested email clients for desktop and smartphones. Outlook Desktop received a mean SUS score of 66.8 (SD=20.6), while Outlook Mobile showed marginally better results, with a SUS score of 72.0 (SD=21.2). On the Bangor's Scale [37], that would be a rating between *OK* and *Good* in both cases. Thunderbird received a SUS of 73.5 (SD=18.7) on the desktop app and a lower score in the mobile version (SUS=63.3, SD=18.5) While the desktop version ranks between *Good* and *Excellent* on the Bangor's Scale, the mobile version only ranks between *OK* and *Good*. Apple Mail achieved the best results in the desktop version with SUS=79.8 (SD=9.1, between *Good* and *Excellent*). The



(a) Desktop - Everyday Use (b) Smartphone - Everyday Use

(c) Desktop - Setup Phase (d) Smartphone - Setup Phase

Fig. 3: SUS (a, b) results focus on the everyday use of encryption and signatures, and SEQ (c, d) results focus on the setup phase. Higher numbers are better. All subfigures represent boxplots for the three evaluated email applications on desktop (a, c) and smartphone (b, d). The boxes indicate the Interquartile Range (IQR), the center line marks the median, and whiskers represent the minimum and maximum values. The number of participants per condition is shown below each label.

score is lower in the mobile version (SUS=69.5, SD=16.0, between *OK* and *Good*).

C. Behavioral and Interview Findings

In this section, we present the usability test results and relate them to the interview findings. Participants are identified by the first letter of their email client—(O)utlook, (A)pple Mail, or (T)hunderbird—followed by a number.

1) Obtaining Certificates ($\bullet_1$): Most participants chose one of the top Google search results. Price played a major role: participants would pay for a certificate, but 16 participants preferred a lower-priced option, and some tried to find a free certificate, without success. This may be linked to the involvement of many students. Most participants were unfamiliar with S/MIME certificates, making it difficult for them to assess their cost-benefit. Participant O02 chose one that is free of charge in the first year and justified his decision as follows: "*[...] I would have given it a try first. And after a year, I could have decided whether it was worth it to me or not.*" Higher prices were seen as a sign of greater security.

Only five participants considered different validation levels (e. g., mailbox-validated, individual-validated). Seven participants would pay attention to ratings on comparison websites or on Google Search. Several participants preferred German or European providers (4), or providers they were already familiar with and/or whose website they found trustworthy (9). Four participants initially selected a Transport Layer Security (TLS) certificate rather than an S/MIME certificate, one of them through sponsored ads. Six participants described selecting and acquiring a high-quality certificate as challenging, particularly due to the additional steps and responsibilities involved, as A09 explained: "*I would have had to create an account, then download the certificate, and afterwards remember to cancel it at some point so I wouldn't be charged. That would have been harder because I am very forgetful.*"

2) Integrate Certificates & Enable Encryption ($\bullet_{2,3}$): In Apple Mail, participants easily installed and set up the certificate because it was automatically assigned to the correct mail account. However, this led to uncertainty among seven participants. Participants who found older instructions requiring manual assignment and activation, or AI-generated results that included those instructions, felt confused because they could not locate the necessary settings. Participant A07 searched for a suitable setting to connect the certificate to the email account until we suggested that he could try writing an email. He discovered the encryption and signature icons and expressed his confusion: "*Okay, but why? I haven't done anything in Mail. Why is this already configured?*" The certificate failed to appear in the keychain in Apple Mail on four occasions. Participants thought they had made an error and kept trying, although a restart of the keychain app was required.

Participants encountered more difficulties with Outlook, as it requires more steps, relies on advanced settings, and often leads to missed steps. All participants searched the Internet or used AI for instructions. Despite this, seven participants needed advice or assistance. At the same time, the setup process lacked transparency in some instances, and participants were uncertain whether everything was set up correctly or if they had forgotten something, as participant O07 noted: "*I haven't received any confirmation yet. Has anything changed here? There's nothing here yet. I think it should be fine now. But I'm not sure...*" Overall, eight participants sent a test email to themselves or to Robin Winter to see if they could discover

the new features. In Outlook, S/MIME settings are not clearly visible and can only be accessed through other tabs, leading participants to mistakenly use different encryption methods (see Section V-C5).

3) Create Key Pairs & Enable Encryption ($\blacksquare_{1,3}$): In Thunderbird Desktop, there are many menus and settings for encrypting emails, which participants found difficult to navigate to the correct options to generate a key pair. Unfortunately, early in our testing, participants had difficulty finding the specific button in the settings that leads to the E2EE options because it did not stand out from the text. During testing, Thunderbird automatically installed security updates (145.0 $\rightarrow$ 146.0) that made the button more obvious. Five participants still used the older version, the rest used the newer one. Difficulties arose also because Thunderbird supports both OpenPGP and S/MIME, leading four participants to mix them up. Some participants were also uncertain about the technical parameters required to generate the keys. In total, six of the participants needed advice or help.

4) Verifying Email (T_4): Overall, 33 participants out of 34 correctly identified the signature of Robin Winter. Most participants emphasized that the way the signature is displayed on the desktop via a ribbon icon and label (Outlook), only via an icon (Thunderbird), or via a check mark icon and label (Apple Mail) is clear and unambiguous. In Apple Mail, problems arose when participants wanted to check the signature more closely, as only the check mark icon was clickable, not the entire label. In this issue, five participants had difficulty or needed help hitting the icon precisely. On the iPhone Apple Mail App, it is easier to access the corresponding menu because the user can tap on the entire sender name.

5) Send Encrypted Email (T_5):

a) Signing: We encountered a problem that participants mistook the textual email signature for the S/MIME signature. That was especially a problem with Outlook Desktop (7 participants), but also with Thunderbird Mobile (T11), and Apple Mail Desktop (A02). Some only hesitated briefly and recognized the same terminology themselves, while others were in doubt until the end. Participant O01 searched extensively both within the Outlook, using help icons, and on the official Microsoft support page before creating a new textual email signature and entering the text "I'm real", after not finding alternative solutions.

b) Encryption: Three participants said they would first send a test email to themselves to ensure everything is set up correctly, before sending sensitive data. About half of the participants verified after sending a message in the *Sent Items* folder whether there were any indicators that the email was actually encrypted and signed. For A04, it indicated working encryption, as he had not yet transferred the certificate to his smartphone and could not access the email in his inbox.

In our usability test, 22 participants successfully sent a correctly encrypted and signed email without assistance on the desktop device; however, most had received assistance in preceding tasks. Only eight participants completed the entire desktop workflow without assistance. The remaining 12 needed guidance and, in some cases, up to three attempts. With Apple Mail, all except one person succeeded right away without any major problems. Most of the problems arose with the Outlook desktop application, where eight participants needed help. Out of these, five participants first noticed the integrated Microsoft Purview Message Encryption (MPE) and activated it. Participant O09 noticed after sending a test email that this might be a different type of encryption, and wondered: "*But now the question is, is it really encrypted with the certificate, or is it just one of those Microsoft in-house encryption methods?*" We informed the other participants who had not noticed this, and that they should try again. The main problem we found was that the settings for S/MIME encryption are hidden in "*huge submenus*" (O02), while the settings for MPE were relatively easy to find. With Thunderbird Desktop, only a handful of participants encountered difficulties, i. e., two participants did not select the newly generated key pair for their email accounts, which turned off the encryption option.

6) Key Management ($\blacksquare_{5,9}$): Some participants were confused about the key management process, particularly why they could not encrypt the email before entering the sender's address. In general, key management is performed automatically by the applications. Thunderbird still requires the most user action when importing keys, but provides guidance, such as small pop-ups. Although participants were tasked with ensuring that the recipient could also reply in encrypted form, only participant T05 attached their own public key to the email. Since Thunderbird automatically adds the sender's public key by default, this oversight did not cause any functional problems. In Thunderbird, it also became apparent that participants were confused by the public keys in the attachment. Although T12 personally did not consider it dangerous, he noted that others might think it is: "*[...] if someone isn't technically savvy, those numbers and the attached file mean nothing to them. They just think it's some kind of malware.*"

7) Multi-Device Usage (T_6): Plenty of participants (20) found it difficult to transfer the key pair or certificate to their smartphones and integrate it, often viewing it as the hardest step. Initially, seven participants forgot they still had to transfer the certificate or key pair and first searched for it on their smartphones.

a) Exporting the Key Pair: Regarding task $\blacksquare_5$, eleven participants overlooked the option *Create backup of private key* in Thunderbird, because they expected different wording or the option was too hidden. Ten participants quickly discovered the *export public key* option and expected similar terminology for the export of the private key like T03 explained: "*[...] especially on the phone when the issue of importing came up, [...], you automatically think of the opposite as exporting [...]*" Participant T11 spent a long time searching for the export function, but guessed why it was so hidden: "*Particularly if you are not very familiar with these two things [public and private key], you won't quickly get access to the private key to share it in any form. Because I believe that if you are not aware of how encryption works, you can very quickly end up publishing your private keys.*" Because it was an

obvious feature, three participants tried the *Send public keys by email* function, but could not figure out how to use it on the smartphone. One participant published the public key on a keyserver and thought she could access it on her smartphone. After participant T09 received assistance from us and discovered the correct functions, he replied: "*Admittedly, an outrageously confusing system.*"

b) Transfer to Smartphone.: A total of 18 participants considered sending the certificate or key pair to themselves by email. Seven of these considered this a secure option, having now successfully set up encryption, and wanted to send the certificate in an encrypted email explicitly. (T05: "*Because if I already have encryption via OpenPGP here [in the desktop application], then it should probably work.*" Others were less concerned about channel security and saw it as the easiest way. However, four participants successfully enabled the option to encrypt all emails by default, they still sent themselves an encrypted email that also contained the certificate or private key in encrypted form as an attachment. On the smartphone, they tried to open the encrypted email but realized it was not possible. Some participants directly noticed their misstep and took it with humor, as participant T05 did when he read the notification about the missing key: "*Import the key... [laughs] Of course, it's a bad idea to send an encrypted message to yourself and then not have the key.*"

Regarding Outlook, participants had to send the certificate to themselves. Three participants expressed surprise that this was the case, as they did not consider it to be a particularly secure option. O02: "*Wow, usually that's not encrypted. And sending a key like that over an unencrypted channel is actually a bad idea.*" Although Microsoft recommends setting a secure password for transferring the certificate, we found that this can cause practical difficulties. The password must be entered on the smartphone, and, as participant O11 reported, it cannot be displayed, which caused him problems when entering it. O11: "*I don't know if I typed that correctly, because I typed the whole thing rather sloppily, since I can't even make the password visible.*" Apple Mail caused the fewest difficulties. Participants preferred Airdrop or iCloud for transferring data.

8) Enable Encryption on Mobile (T_7): In the Outlook group, after participants sent the certificate to themselves, five did not tap the attachment directly; instead, they opened the context menu to choose where to save the file. They correctly saved the certificate on the iPhone and installed it, so it appears in the iPhone's certificate storage. Because Outlook Mobile cannot access the device's certificate storage, the app cannot detect the certificate, leaving participants confused about why they cannot select it in the Outlook S/MIME settings. In Thunderbird Mobile, participants reported almost no problems with this step. In Apple Mail Mobile, difficulties occurred upon installation of the certificate. A new option in the settings links directly to the appropriate section, but about two-thirds of users overlooked it and spent a long time searching. About half of the participants forgot to confirm the S/MIME settings because the button was on the previous interface, leading to confusion about why it was not working despite activation.

9) Verify Email on Mobile (T_8): Overall, 18 participants (8 Outlook, 10 Apple Mail) correctly identified legitimate emails among fake emails. However, particularly with Thunderbird mobile, participants had difficulties, since the app's default setting displays signatures only for encrypted emails. Many participants were confused because the legitimate email and the email with the broken signature looked identical. They then tried to find other clues, e. g., considered the missing attachment to be critical. We pointed out that there is an option in the settings that could help here, but only participant T08 found it right away. Others needed more support and countered this with disapproval, as in the case of participant T06: "*So why is that the default setting? It doesn't make any sense at all.*" Once the participants enabled this option, seven of them correctly classified the emails.

Both Outlook Mobile and Apple Mail on the iPhone clearly showed the differences between the broken and valid signatures for the participants. Overall, 8 participants (3 Outlook, 4 Thunderbird, 1 Apple Mail) did not recognize the HTML-generated signature in the email body. However, most participants explicitly stated that the signature was included in the email content and is not part of the User Interface (UI).

In general, we found evidence that participants noticed that something was wrong with the broken signature, but were unsure about its meaning, as O07 said: "*I don't know why it can't be verified. [...] That's why I said, okay, for me it's not wrong per se. It's just unknown. So I initially treated it with caution and wondered if it might be a fake.*" This is similar to the response of participant O09, who expressed the same view but would seek confirmation through a different channel: "*I don't know right now how the system handles that. I would just contact Robin directly and ask, 'Hey, did you really send this email? Could you please send it again with a different signature?' Or I would open it on my laptop so I could see it better.*" It appears that this made the participants more aware and more careful when handling the email, even if they were uncertain about its exact meaning. Nevertheless, a forged signature may succeed and lead users to believe the system is more secure than it actually is.

10) Send Encrypted Email on Mobile (T_9):

a) Signing.: In the Thunderbird mobile app, participants searched for a way to add a digital signature manually but found nothing, since it is applied automatically for encrypted emails. For this reason, T11 inserted the textual email signature, as she could not find anything else. Participants were also confused in Apple Mail Mobile about whether their own email was signed, as the signature icon disappeared when the email was encrypted.

b) Encryption: Participants had no major issues sending encrypted emails. They could digitally sign emails on their smartphones, either automatically if set up in advance or through a clearly visible option. However, some participants were confused about the meaning of the icons indicating encryption and signature, as they did not find a label next to them. For both Outlook Mobile and Thunderbird, there were two participants who forgot either the signature or encryption.

It is not entirely clear whether they just overlooked it in the task description or mistakenly did not activate it.

11) Use of Assistive Tools: All participants researched on the Internet. Some participants referred to AI results directly suggested by Google search results, while others used separate web-based LLM services. In some cases, participants searched the Internet before even looking at the application. Particularly when participants used the wrong instructions, for example, for iOS instead of Mac, or outdated instructions, this led to confusion and uncertainty. We found indications that some participants mindlessly followed the guides they encountered, without reflection or consideration of what they were doing. For example, participants were confused that new emails were signed and encrypted even though they enabled the *Always encrypt and sign* options during setup. Some attempted to export their imported certificate as instructed, despite it still being in the original folder.

D. User Reflections

We summarize participants' reflections from think-aloud comments and follow-up interviews not tied to specific tasks.

1) Demand for Encryption: One of the main reasons participants were not using email encryption was that they preferred simpler alternatives, such as messengers or cloud storage. Often, it depended on the email's content and trust level, or on the effort required, which was rated as too high. Participant O09 explained it as follows: "*Despite everything, it's always a question of cost-benefit. So, do I really want to encrypt my lunch orders to the Asian restaurant when I send them by email? Or what added value does that bring for everyone? It's still effort that I have to put in.*" Additionally, nine participants mentioned that recipients also need to use S/MIME or OpenPGP, and 17 participants indicated that handling might be (too) difficult for people who are not IT-savvy.

However, 17 participants reported that the procedures were easier than expected. Especially once the setup is complete, the remaining steps proceed almost automatically (15) or could become a routine step (3). Participant T03 said: "*Well, once it's set up and also on your smartphone, which you only do once or at most when you get a new phone. [...] It's not something you do every day. And now doing it [encrypting and signing] in the email was easy, for sure. And once you've practiced it three times, I think you'll be fine with it, it shouldn't be a problem.*" Participants also saw other advantages, such as a general simplification of everyday life (T03, A05), support in recognizing fraudulent emails (A06, T10), or feeling more secure (8).

2) Trust and Transparency: In general, participants mostly trusted encryption and signatures. They appreciated the ability to choose different encryption algorithms during setup to enhance security, though the technical terms caused some uncertainty. Participants particularly trusted Thunderbird as an email client, as it is widely known as open-source software (6). In total, 19 participants criticized a lack of transparency and feedback, which not only eroded their trust in the system but also left them uncertain whether everything worked as intended. Participant O06 criticized the limited transparency of encryption, but at the same time considered it favorable for the average user: "*I just clicked on "Encrypt" and then entered a password. [...] but since you can't see what's in the background, which is probably a good thing for most people before they break something or whatever, you just have to trust that what it says is correct.*" Overall, participants found the visual representation of encryption and digital signatures understandable, despite minor usability issues—mainly on smartphones (e. g., missing labels)—especially after reviewing their own emails. Remaining uncertainty concerned the application's background processes rather than the interfaces.

VI. Discussion

A. Addressing the Research Questions

Regarding RQ1, we found that email encryption continues to bring major usability problems. While difficulties occur across all stages of use, the most pronounced issues arise during system setup and configuration rather than during encryption, signature creation, or key management itself. According to our SUS questionnaire results from Section V-B, several scores fell within the *good* to *excellent* range, while even the lowest ratings indicate average usability after the initial setup. Our findings indicate that opaque background processes contributed to participants' uncertainty, similar to [18], [19]. The issue stems less from the lack of transparency in the processes themselves and more from the lack of feedback that engages and guides users. Overall, participants trusted the encryption processes but often assumed they had missed a step or searched for options that did not exist. This was mainly due to a lack of feedback, prompting participants to send test emails to verify their configurations.

Regarding RQ2, the greatest difficulties occurred during the transfer of the key material to another device and its configuration on that device, as seen in Figure 3d, particularly with Thunderbird and Apple Mail. Some participants, intentionally or unintentionally, sent themselves an encrypted email with the key material attached and thus could not access it on their smartphones. However, we suspect this issue arose from the participants' initial sense of success after successfully setting up encryption on desktop devices, where sending encrypted emails was considered relatively simple and automated. This sense of achievement likely motivated them to approach the task with an overly optimistic mindset, leading them to overlook potential pitfalls. Participants did not deliberately choose an unsafe path; rather, they chose the one that required the least effort, reflecting users' incomplete mental models of E2EE systems [26]. It also aligns with the fact that users often use email as a workaround to transfer data in multi-device scenarios [38]. Also, participants were unsure how to securely transfer their key material to another device, as the guidelines they followed lacked details on the transfer process and did not confirm the safety of cloud services.

Regarding RQ3, participants noticed digital signatures and distinguished them from tampered content in many cases. Some individuals failed to notice the forgery, which creates

misplaced trust and may introduce new risks, as users might unwittingly act on harmful content. They also detected broken signatures and recognized that something was wrong, which made them more attentive and prompted them to double-check, even if they did not fully understand the underlying details. On mobile devices, limited user interfaces failed to display all relevant elements, thereby concealing important information. Thunderbird mobile hides signatures in unencrypted emails, while Apple Mail omits UI elements for signatures in encrypted emails. Digital signatures verify an email's authenticity and integrity, helping users recognize fraudulent messages and enhancing secure communication without encryption.

B. Recommendations

We recommend steps to make email encryption and digital signatures more accessible to the public.

Cross-Application Recommendations:

- Automate technical processes in the background, but provide clear feedback to keep users informed.
- Guidelines should be comprehensive and cover cross-application workflows, such as key transfer.
- Clarify terms with multiple meanings (e. g., "signatures") to prevent confusion.
- Use recognizable icons for encryption and signatures, especially on smartphones, and support them with labels and color-coded visualizations, while ensuring accessibility through non-color-dependent cues.
- In organizations, let IT specialists handle complex setup and key transfer, allowing users to engage only in streamlined, automated processes.

Outlook:

- Make encryption settings easier to find and access.
- Allow other options to transfer the certificate.

Apple Mail:

- Provide more feedback during certificate installation on desktop.
- Automate certificate integration and encryption setup on mobile devices, similar to the desktop version but with feedback.

Thunderbird:

- Make private key export easier to find and protect it with a confirmation dialog from accidental disclosure.
- Make digital signatures visible for unencrypted emails on the mobile app.

C. Limitations

A limitation of our study is the simplified gender categorization used in the survey, which deviates from recommended survey practice [39] and may reduce inclusivity. We conducted a controlled usability test in a lab environment. Therefore, participants may have behaved differently than they would in real life. Most of our participants were students or university staff, who might be more tech-savvy than the general population. Some of the participants were students from our courses and received bonus points for their contribution, which in turn may have influenced their behavior. When participants were stuck, we helped them and generally provided feedback on whether they had completed a task correctly. This may have influenced the overall assessment. Furthermore, the scenario had some limitations, as it did not include the full certificate acquisition process, used fake emails with manipulated signatures, and only tested one-time synchronisation of keys. We were unable to test the synchronization of the S/MIME certificate with M365 in Outlook, as it would have required an organization-wide settings change.

VII. Conclusion

In this paper, we conducted a think-aloud usability study with $N = 34$ participants using popular email clients such as Thunderbird, Apple Mail, and Outlook, in which participants needed to encrypt, decrypt, sign, transfer cryptographic key material to a mobile device, and verify the authenticity of a received email. We show that multi-device use further increases the difficulty of email encryption. In our study, we found that many participants encountered considerable challenges when trying to integrate key material, set up encryption, and transfer it to their mobile devices, even with popular email clients. Our results indicate that participants were generally successful at detecting invalid signatures, leading some to report increased vigilance in managing their emails. Although the initial setup posed challenges, participants discovered that once they correctly configured encryption and signing, the process required minimal effort and was easier than expected.

As future work, we need more research that is not conducted in laboratory conditions but instead in real-world settings, with more heterogeneous participants and over a longer period of time. Additionally, the renewal or exchange of certificates and PGP keys should be tested when they have expired. It would also be valuable to gain a deeper understanding of the participants' knowledge of encryption and, for example, to explore whether they are aware of the types of attacks it protects against. For future studies, we recommend implementing automated procedures to reset the devices in order to prevent problems when restoring them to their pre-experiment state.

Acknowledgments

We used Grammarly for grammar and language revision. This work was funded by the European Union under grant number ROF-SG20-3066-3-2-2.

References

[1] "Data Never Sleeps 12.0," *domo.com*, [Online]. Available: https://www.domo.com/learn/infographic/data-never-sleeps-12. [Accessed: Feb. 6, 2026].

[2] G. Greenwald and S. Ackerman. "The Guardian: NSA collected US email records in bulk for more than two years under Obama," *theguardian.com*, [Online]. Available: https://www.theguardian.com/world/2013/jun/27/nsa-data-mining-authorised-obama. [Accessed: Feb. 16, 2026].

[3] B. Krebs. "At Least 30,000 U.S. Organizations Newly Hacked Via Holes in Microsoft's Email Software – Krebs on Security," [Online]. Available: https://krebsonsecurity.com/2021/03/at-least-30000-u-s-organizations-newly-hacked-via-holes-in-microsofts-email-software/. [Accessed: Feb. 7, 2026].

[4] K. Weise. "Microsoft Executives' Emails Hacked by Group Tied to Russian Intelligence," [Online]. Available: https://www.nytimes.com/2024/01/19/technology/microsoft-executive-emails-hacked.html. [Accessed: Feb. 7, 2026].

[5] P. Zimmerman. "Why I Wrote PGP," [Online]. Available: https://www.philzimmermann.com/EN/essays/WhyIWrotePGP.html. [Accessed: Feb. 6, 2026].

[6] C. Stransky, O. Wiese, V. Roth, Y. Acar, and S. Fahl, "27 Years and 81 Million Opportunities Later: Investigating the Use of Email Encryption for an Entire University," in *IEEE Symposium on Security and Privacy (S&P)*, 2022, p. 16. DOI: 10.1109/SP46214.2022.9833755.

[7] A. Whitten and J. D. Tygar, "Why Johnny Can't Encrypt: A Usability Evaluation of PGP 5.0," in *8th USENIX Security Symposium (USENIX Security 99)*, 1999.

[8] C. Lewis, "Using the "Thinking-aloud" Method in Cognitive Interface Design," IBM Thomas J. Watson Research Center, Tech. Rep., Feb. 17, 1982.

[9] J. Brooke, "SUS: A "quick and dirty" usability scale," in *Usability Evaluation in Industry*, P. W. Jordan, B. Thomas, B. A. Weerdmeester, and I. L. McClelland, Eds., Taylor & Francis, 1996, pp. 189–194.

[10] J. Sauro and J. S. Dumas, "Comparison of Three One-Question, Post-Task Usability Questionnaires," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, Apr. 4, 2009, pp. 1599–1608. DOI: 10.1145/1518701.1518946.

[11] rmoritz. "List of Large Germany Companies and Organizations Using S/MIME," [Online]. Available: https://gist.github.com/rmoriz/5945400#file-2_smime-md. [Accessed: Feb. 7, 2026].

[12] Electronic Frontier Foundation (EFF). "How to: Use PGP for Windows," *Electronic Frontier Foundation (EFF)*, [Online]. Available: https://ssd.eff.org/module/how-use-pgp-windows. [Accessed: Feb. 19, 2026].

[13] S. L. Garfinkel and R. C. Miller, "Johnny 2: A User Test of Key Continuity Management with S/MIME and Outlook Express," in *Proceedings of the 2005 Symposium on Usable Privacy and Security (SOUPS)*, 2005. DOI: 10.1145/1073001.1073003.

[14] S. Sheng, L. Broderick, C. A. Koranda, and J. J. Hyland, "Why Johnny Still Can't Encrypt: Evaluating the Usability of Email Encryption Software," in *Symposium on Usable Privacy and Security (SOUPS '06)*, Poster Abstract, ACM, 2006, pp. 3–4.

[15] S. Ruoti, J. Andersen, T. Hendershot, D. Zappala, and K. Seamons, "Private Webmail 2.0: Simple and Easy-to-Use Secure Email," in *Proceedings of the 29th Annual Symposium on User Interface Software and Technology, UIST*, 2016, pp. 461–472. DOI: 10.1145/2984511.2984580.

[16] E. Vaziripour, J. Wu, M. O'Neill, R. Clinton, J. Whitehead, S. Heidbrink, K. Seamons, and D. Zappala, "Is that you, Alice? A Usability Study of the Authentication Ceremony of Secure Messaging Applications," in *Proceedings of the Thirteenth Symposium on Usable Privacy and Security (SOUPS)*, 2017, pp. 29–47, ISBN: 9781931971393.

[17] S. Ruoti, J. Andersen, L. Dickinson, S. Heidbrink, T. Monson, M. O'neill, K. Reese, B. Spendlove, E. Vaziripour, J. Wu, D. Zappala, and K. Seamons, "A Usability Study of Four Secure Email Tools Using Paired Participants," *ACM Transactions on Privacy and Security (TOPS)*, vol. 22, no. 2, Apr. 2019. DOI: 10.1145/3313761.

[18] S. Ruoti, N. Kim, B. Burgon, T. v. d. Horst, and K. Seamons, "Confused Johnny: When Automatic Encryption Leads to Confusion and Mistakes," in *Proceedings of the Ninth Symposium on Usable Privacy and Security (SOUPS)*, 2013. DOI: 10.1145/2501604.2501609.

[19] 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 *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI)*, 2016, pp. 4298–4308. DOI: 10.1145/2858036.2858400.

[20] S. Ruoti, J. Andersen, T. Monson, D. Zappala, and K. Seamons, "A Comparative Usability Study of Key Management in Secure Email," in *Proceedings of the Fourteenth Symposium on Usable Privacy and Security (SOUPS 2018)*, 2018, ISBN: 978-1-939133-10-6.

[21] 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 *Proceedings of the Twelfth Symposium on Usable Privacy and Security (SOUPS)*, 2016, pp. 113–130, ISBN: 9781931971317.

[22] J. S. Koh, S. M. Bellovin, and J. Nieh, "Why Joanie Can Encrypt: Easy Email Encryption with Easy Key Management," in *Proceedings of the Fourteenth EuroSys Conference 2019*, 2019, pp. 1–16. DOI: 10.1145/3302424.3303980.

[23] J. Blessing, D. Hugenroth, R. J. Anderson, and A. R. Beresford, "SoK: Web Authentication and Recovery in the Age of End-to-End Encryption," in *Proceedings on Privacy Enhancing Technologies (PETS)*, 2025. DOI: 10.56553/popets-2025-0113.

[24] E. Atwater, C. Bocovich, U. Hengartner, E. Lank, and I. Goldberg, "Leading Johnny to Water: Designing for Usability and Trust," in *Proceedings of the Eleventh Symposium On Usable Privacy and Security (SOUPS)*, 2015, pp. 69–88, ISBN: 978-1-931971-249.

[25] A. Lerner, E. Zeng, and F. Roesner, "Confidante: Usable Encrypted Email: A Case Study with Lawyers and Journalists," in *IEEE Symposium on Security and Privacy (S&P)*, 2017. DOI: 10.1109/EuroSP.2017.41.

[26] K. Renaud, M. Volkamer, and A. Renkema-Padmos, "Why Doesn't Jane Protect Her Privacy?" In *Proceedings on Privacy Enhancing Technologies (PETS)*, 2014, pp. 244–262. DOI: 10.1007/978-3-319-08506-7_13.

[27] 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. DOI: 10.1109/SP.2017.65.

[28] S. Gaw, E. W. Felten, and P. Fernandez-Kelly, "Secrecy, Flagging, and Paranoia: Adoption Criteria in Encrypted Email," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI)*, 2006, pp. 591–600. DOI: 10.1145/1124772.1124862.

[29] S. Fahl, M. Harbach, T. Muders, M. Smith, and U. Sander, "Helping Johnny 2.0 to Encrypt His Facebook Conversations," in *Proceedings of the Eighth Symposium On Usable Privacy and Security (SOUPS)*, 2012. DOI: 10.1145/2335356.2335371.

[30] S. Ruoti and K. Seamons, "Johnny's Journey Toward Usable Secure Email," *IEEE Security & Privacy*, vol. 17, no. 6, pp. 72–76, 2019. DOI: 10.1109/MSEC.2019.2933683.

[31] K. Schiller and F. Adamsky, "Work in Progress: Can Johnny Encrypt E-Mails on Smartphones?" In *Socio-Technical Aspects in Security: 11th International Workshop (STAST)*, 2021, pp. 182–193. DOI: 10.1007/978-3-031-10183-0_9.

[32] M. E. Cecchinato, A. Sellen, M. Shokouhi, and G. Smyth, "Finding Email in a Multi-Account, Multi-Device World," in *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, 2016, pp. 1200–1210. DOI: 10.1145/2858036.2858473.

[33] K. Schiller, A. Herrmannsdörfer, Z. Benenson, and F. Adamsky, "Extended Version: PGP and S/MIME in the Age of Multi-Device Email Usage: "Admittedly, an outrageously confusing system."," in *Conference on Sociotechnical Cybersecurity and Privacy (SCP)*, 2026. [Online]. Available: https://florian.adamsky.it/research/publications/2026/2026-email-encryption-extended.pdf.

[34] M. Bailey, D. Dittrich, E. Kenneally, and D. Maughan, "The Menlo Report," *IEEE Security & Privacy*, vol. 10, no. 2, pp. 71–75, 2012. DOI: 10.1109/MSP.2012.52.

[35] P. Mayring, "Qualitative Content Analysis," *Forum Qualitative Sozialforschung / Forum: Qualitative Social Research*, vol. 1, no. 2, 2000. DOI: 10.17169/fqs-1.2.1089.

[36] J. Sauro and J. Lewis. "Describing SEQ® Scores with Adjectives," *MeasuringU*, [Online]. Available: https://measuringu.com/adjective-interpretations-of-seq-scores/. [Accessed: Feb. 18, 2026].

[37] A. Bangor, P. Kortum, and J. Miller, "Determining What Individual SUS Scores Mean: Adding an Adjective Rating Scale," *Journal of Usability Studies*, vol. 4, no. 3, 114–123, May 2009.

[38] S. Santosa and D. Wigdor, "A Field Study of Multi-Device Workflows in Distributed Workspaces," in *Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp)*, 2013, pp. 63–72. DOI: 10.1145/2493432.2493476.

[39] K. Spiel, O. Haimson, and D. Lottridge, "How to Do Better with Gender on Surveys: A Guide for HCI Researchers," *Interactions*, vol. 26, no. 4, pp. 62–65, Jun. 2019. DOI: 10.1145/3338283.