Sounds Good? Fast and Secure Contact Exchange in Groups

FLORENTIN PUTZ, Technical University of Darmstadt, Germany
STEFFEN HAESLER, Technical University of Darmstadt, Germany
MATTHIAS HOLLICK, Technical University of Darmstadt, Germany

Fig. 1. Group of users exchanging their contact information using our acoustic group pairing protocol.

Trustworthy digital communication requires the secure exchange of contact information, but current approaches lack usability and scalability for larger groups of users. We evaluate the usability of two secure contact exchange systems: the current state of the art, SafeSlinger, and our newly designed protocol, *PairSonic*, which extends trust from physical encounters to spontaneous online communication. Our lab study ($N = 45$) demonstrates PairSonic's superior usability, automating the tedious verification tasks from previous approaches via an acoustic out-of-band channel. Although participants significantly preferred our system, minimizing user effort surprisingly decreased the perceived security for some users, who associated security with complexity. We discuss user perceptions of the different protocol components and identify remaining usability barriers for CSCW application scenarios.

Authors' Contact Information: Florentin Putz, Technical University of Darmstadt, Darmstadt, Germany, fputz@seemoo.de; Steffen Haesler, Technical University of Darmstadt, Darmstadt, Germany, haesler@peasec.tu-darmstadt.de; Matthias Hollick, Technical University of Darmstadt, Darmstadt, Germany, mhollick@seemoo.de.

ACM 2573-0142/2024/11-ART425
https://doi.org/10.1145/3686964

CCS Concepts: • **Human-centered computing → Empirical studies in ubiquitous and mobile computing**; **Empirical studies in collaborative and social computing**; **Computer supported cooperative work**; **Collaborative interaction**; *Smartphones*; • **Security and privacy → Usability in security and privacy**; *Authentication.*

Additional Key Words and Phrases: usable security; group pairing; secure device pairing; trust establishment; acoustic communication; data-over-sound; authentication ceremony; key verification; peer-to-peer-authentication; public-key cryptography; security; privacy; social cybersecurity; end-to-end-encrypted messaging; MitM attacks; instant-messaging; online collaboration; contact management; contact sharing; device association; binding; bonding; coupling; decentral; ad-hoc; interoperability; decentral; spontaneous; nearby; proximity; smartphone; user interaction; user perception; prototype; open-source; WiFi; NFC

ACM Reference Format:
Florentin Putz, Steffen Haesler, and Matthias Hollick. 2024. Sounds Good? Fast and Secure Contact Exchange in Groups. *Proc. ACM Hum.-Comput. Interact.* 8, CSCW2, Article 425 (November 2024), 44 pages. https://doi.org/10.1145/3686964

1 Introduction

Imagine a group of seven researchers who just met for the first time at a conference. During the event, they notice that they share a lot of common interests – they might even have some ideas for future collaborations – so they decide to stay in contact. They all have a smartphone, but to securely communicate online, they need a way to *securely exchange their contact information.* In this paper, we study *group pairing* methods for the fast and secure exchange of contact information that fulfill the following two goals: first, they allow the participants to contact each other online, to establish new conversations or to start new digital collaborations. Second, they authenticate the participants by leveraging their physical encounter to verify their contact information.[1]

There are countless similar use cases for online collaboration besides the aforementioned group of researchers, ranging from professional activities (team communication, remote work, project management), to educational settings (interactive learning platforms, knowledge management), and leisure (event planning and coordination), affecting billions of users in today's digital world [42, 57, 75, 82, 109]. For all these scenarios, secure exchange of contact information is required to associate known physical individuals with their online profiles and protect their privacy and security when interacting online. Without such a secure contact exchange, an adversary can eavesdrop on private communication or even impersonate one of the communication participants in order to modify or create fake messages that appear legitimate.

Effective communication and collaboration thrives in a trusted environment, where participants have clear knowledge of the identities of others involved and no concerns about unauthorized access to their conversations. Such an environment encourages open discussions, even on sensitive topics [104]. Previous research has shown that users desire authenticated conversations – wanting to know exactly *who* they are communicating with – when talking to friends and family members [26], but also when communicating with colleagues or business partners [25], and especially so for vulnerable groups such as journalists and activists [45]. Secure contact exchange is therefore an important prerequisite for fruitful and trustworthy online communication and collaboration [86].

1.1 Current Group Pairing Is Insufficient

Throughout the last decade, users have increasingly adopted end-to-end encrypted (E2EE) tools such as Signal [106], Whatsapp [80], Telegram [112], and Threema [113], facilitating group collaboration with features like real-time messaging, file sharing, and video calls. These E2EE tools also offer the

[1] The exchanged contact information usually contains additional metadata and public-key material, e.g., for secure end-to-end encrypted (E2EE) instant messaging.

verification of contact information through an *authentication ceremony*, which requires users to perform a manual verification action, either by scanning quick-response (QR) codes or by manually comparing public key fingerprints on their devices. This authentication ceremony allows users to transfer an existing trust relationship from a physical encounter to the corresponding digital identity of their communication partners, enabling authenticated conversations. There are two problems with the design of authentication ceremonies in state-of-the-art tools for secure communication and collaboration:

(1) **Low usability.** Current authentication ceremonies are *hard to use*, even for professionals.
(2) **Bad scalability.** Current authentication ceremonies *do not support larger groups* of users.

First, previous usability research has shown that users struggle to understand and perform authentication ceremonies [86, 122, 123, 129]. Most users expect that E2EE tools like Signal are secure by default without further verification actions, when in reality these tools cannot protect against active machine-in-the-middle (MitM) attacks and rogue service operators unless the users diligently perform the authentication ceremony [45, 121]. These authentication ceremonies, however, require considerable user interaction [44], take a long time [122], and are difficult to use even for security professionals [26, 100]. Recently, the risks of active MitM attacks in E2EE tools have gained more attention due to reports of law enforcement agencies compromising app providers [37, 69], demonstrating a practical and concerning risk for privacy infringements, especially if such vulnerabilities were to be exploited by malicious actors in the future.

Second, state-of-the-art tools like Signal and WhatsApp do not support efficient and secure contact exchange for *groups of multiple users*, but only between two users. This is a substantial shortcoming, as collaboration often involves more than two participants. While each pair of group members could separately perform the bilateral authentication ceremony one after another, this is infeasible as the number of manual verification actions scales quadratically ($\frac{n(n-1)}{2}$) with the group size (n). To illustrate this: if our exemplary group of seven researchers were to use Signal, they would need to perform 21 manual actions amongst each other for authentication, but each user can only perform one of these actions at a time. Such an authentication takes too much time and would discourage the spontaneous decision to collaborate.

The main challenge is how to authenticate the exchanged contact information and key material without having to rely on any trusted third parties or public-key infrastructures (PKIs) that facilitate the exchange. Farb et al. [23] achieved a breakthrough towards solving this problem when they proposed the more usable *SafeSlinger* protocol for the secure exchange of contact information in the absence of trusted third parties. While SafeSlinger offers significantly higher usability than alternatives [23], it still requires the participants to perform $\frac{n(n-1)}{2}$ manual text comparisons to achieve its security guarantees. Since the release of SafeSlinger, several studies on authentication methods suggested that users generally prefer simpler schemes requiring less involvement [78, 130] and that involving users for manual comparisons is prone to errors [53, 111, 122]. Furthermore, recent research has explored the intuitive actions users might take to associate multiple devices [14, 48], suggesting that the user experience in SafeSlinger may not align with user expectations. The required amount of user interaction is potentially discouraging, especially considering larger groups. Instead of facilitating trustworthy communication and collaboration, state-of-the-art methods rather inhibit it (see Section 3 for a more extensive discussion of related work).

1.2 Approach

In this work, we approach the challenge of securely exchanging contact information from a different angle, under the research hypothesis that *users would prefer a contact exchange method with minimal user interaction* [64, 78, 130]. Consequently, we design a new group pairing protocol named *PairSonic* that requires less effort and can efficiently scale to multiple participants. The illustrative group of researchers can exchange their contact information securely and quickly by starting PairSonic and simply holding their devices close together for a few seconds, aligning with intuitive association behavior [14, 48, 50]. PairSonic automatically exchanges the participants' contact information using an ad-hoc WiFi channel for communication and a location-limited acoustic out-of-band (OOB) channel for verification. The effectiveness of the acoustic channel has already been demonstrated in previous research [32, 74, 76, 78] and in industry adoption, such as with the Sonos smart speaker [52], for the similar use case of pairwise key verification. Even though the acoustic channel seems promising, its applicability and usability between more than two devices remains unexplored.

1.3 Contributions

Our main contribution to CSCW is the usability evaluation of our novel acoustic approach PairSonic for the secure exchange of contact information and cryptographic public keys for groups of multiple users as a basic requirement for teams to work with sensitive content. We conduct a lab study with $N = 45$ participants – consisting of practical authentication tasks on smartphones, subsequent interviews, and a questionnaire – to compare the state-of-the-art system SafeSlinger with our novel protocol PairSonic, which we design and implement for smartphones. PairSonic requires only minimal user interaction by leveraging an acoustic OOB channel between the users' smartphones, automating the cumbersome verification tasks from previous commercial and academic approaches [19, 45, 117, 118, 122]. This paper is structured as follows:

- We define the problem and establish our requirements for secure contact exchange (Section 2), discussing why the state of the art fails to solve this problem (Section 3).
- We design PairSonic, an improved protocol for fast and secure contact exchange, and implement it for smartphones (Section 5).
- To answer our research questions (Section 4), aiming to understand how automating the cumbersome verification can improve on the current state of the art (SafeSlinger), we conduct a lab study with $N = 45$ participants to compare both systems (Section 6).
- Based on the quantitative insights from our questionnaire (Section 7) and qualitative interviews (Section 8), we discuss our observation that less user interaction seems to improve usability in our case, but can have unexpected negative effects on perceived security. We evaluate the suitability of the acoustic OOB channel for pairing and give recommendations how to improve the state of the art of secure contact exchange (Section 9).
- We provide a replication package with our evaluation scripts and the pseudonymized dataset from our study, which contains 69 variables for each of the 45 participants [90].

2 Problem Setting

This section defines the problem we are addressing (Section 2.1) and our assumptions (Section 2.2).

2.1 Problem Definition

Our goal is to let a group of two or more users that physically meet (Figure 1) spontaneously exchange their contact information [23]. The expected outcome of this *secure contact exchange*[2] is that each participant obtains the authentic and verified contact information of all other participants, including their cryptographic public keys, without having to rely on external key management

[2] The process of exchanging and verifying the identities with a group of participants is sometimes also called *group key verification*, or *group authentication ceremony*, or *peer-to-peer authentication*, or *secure group contact sharing*, or *key establishment for groups*, or *spontaneous device association*. In this work, we call this process *secure contact exchange* or *group pairing*, as we always target groups of two or more users unless otherwise noted.

infrastructure, prior associations, or shared secrets. We target a group of users who physically meet for the pairing process (Section 9.9) and want to be able to set up secure communication channels between all subsets of users within the group. After this pairing process, the users can subsequently communicate in a confidential and authenticated manner over untrusted channels such as the Internet. Our contact exchange protocol has the following requirements:

Usability Requirements

(1) **Minimal user interaction**, due to our research hypothesis that users would prefer such a contact exchange method [64, 78, 130]. The protocol should be easy to use for lay users, facilitating spontaneous ad-hoc collaboration plans [13, 16].
(2) **Scalable to larger groups of users.** The pairing protocol should be fast and not noticeably depend on the number of users.
(3) **Error-proof,** as users can make mistakes. In particular, our protocol should be robust to *rushing users* who prioritize speed and efficiency over adhering to instructions or prompts, potentially compromising the security of the protocol.

Deployability Requirements

(4) **Compatibility.** Our protocol should seamlessly run on existing commercial off-the-shelf (COTS) devices, enabling users to leverage its benefits immediately by simply installing our software. We focus on user-controlled wireless devices such as smartphones, tablets, laptops, and smart watches, without requiring further hardware or firmware modifications.
(5) **Ad-hoc.** Our protocol should not require any existing security context in the form of pre-shared keys or a jointly trusted third party.
(6) **Decentral.** Our protocol should operate independently of any central infrastructure and remain fully functional without an active Internet connection. This requirement is crucial to support people's spontaneous decision-making, facilitating seamless collaboration and communication regardless of their location. This means that no metadata can leak to third parties, as part of our privacy-by-design approach. If a security vulnerability arises in our protocol or implementation, the offline nature of our exchange makes exploitation by attackers significantly more challenging.

Security Requirements

(7) **Confidentiality.** Only the intended participants should receive the contact information.
(8) **Contact authentication.** Each received contact information should match the corresponding honest participant. We require mutual authentication, i.e., each participant authenticates all the other participants.
(9) **Collective pairwise security.** This requirement was postulated by Farb et al. [23], stipulating that each pair of participants in our groupwise exchange should get the same security properties as if they had performed a direct pairwise exchange. This implies that adversarial participants cannot degrade the bindings between other honest participants.

2.2 Assumptions

We assume that all users are physically present at the same location and can utilize the concept of *"group demonstrative identification"* [70], which involves legitimate participants excluding unintended participants or imposters based on personal identification, such as appearance or voice. The denotation of intended and legitimate participants is an inherently social classification – the decision of whom *users* expect to communicate with cannot be achieved by technical measures alone and needs human assistance. Users are responsible for verifying the received contact information from their exchange to ensure it matches their intended participants [23]. They should also reject non-participants and detect any instances of impersonation within the group, such as duplicate entries corresponding to honest participants. We also assume that the participants accurately determine their group size.

2.3 Adversary Model

Although a detailed security evaluation is beyond the scope of this work, we consider potential threats from an adversary with diverse capabilities. The attacker's goal is to breach the security requirements stated in Section 2.1. For example, the adversary might attempt to impersonate one of the legitimate participants by making another participant accept an adversarial cryptographic public key. Such an attack would allow the attacker to launch MitM attacks in the future, manipulating or intercepting confidential communication. However, ensuring the uninterrupted availability of the contact exchange process is not one of our security goals. Therefore, if an adversary disrupts the contact exchange (e.g., by jamming the radio channel), users would need to retry the exchange at another time or place.

Our adversary model is based on the model used in SafeSlinger [23], focusing on both nearby and remote adversaries. These adversaries possess full Dolev-Yao [20] control over WiFi network messages, meaning they can eavesdrop, intercept, modify, replay, or inject radio communication between devices. Likewise, we also assume that the adversary has not compromised the hardware and software of the participants' smartphones, as such attacks are beyond this paper's scope.

In addition to SafeSlinger's adversary model, our research also considers the acoustic out-of-band channel. Based on previous research by Stajano and Anderson [108] and Balfanz et al. [5], we regard this channel as location-limited, characterized by its ability to support *demonstrative identification* and ensure *authenticity* due to the inherent physical constraints of acoustic sound pressure waves. Consequently, while adversaries can eavesdrop on this channel, they cannot transmit undetected. We explore this assumption and potential attacks on the acoustic channel in more detail in Section 9.7.

3 Related Work

Our work focuses on the problem of securely exchanging contact information, including cryptographic key material, in groups of users. This problem relates to several branches of research, which we first summarize and then elaborate in further detail:

- **Device association**[3] is the general act of establishing a communication channel between two or more devices, independent of the duration and the association's security [16]. Our problem is a special case of device association with a specific focus on authenticity and security to facilitate trustworthy communication.
- **Secure device pairing** is closely related to our work, as it involves device association aimed at establishing an authentic communication channel [28]. Whereas previous work mostly focused on connecting two devices (*"pair*ing*"*), our work additionally considers the more complex case of connecting multiple users, although it also functions with just two users. A key distinction is that we aim not only to establish a secure communication channel for the entire group, but potentially for every subset of group members as well: CSCW research has shown that larger groups sometimes like to split dynamically into smaller task forces for more effective collaboration [43, 116]. The group exists temporarily to facilitate the exchange of contact information among its members, but subsequent communication can happen independently of the group's context. Naturally, the group members can also opt to collaborate as a whole group.

[3] Device association is sometimes also called binding, coupling, or bonding [16].

- **Acoustic communication** can be used as a physical OOB channel to transmit data (such as contact information) between nearby devices, which is compatible with all smart devices having a microphone and a speaker. Previous work showed promising results for connecting two devices [32, 74, 78]. We are the first, to the best of our knowledge, to study the suitability and usability of acoustic communication for connecting multiple users.
- **Authentication ceremonies** refer to the process by which users can verify their cryptographic keys in the context of E2EE tools [44]. Existing approaches only have limited usability and do not support more than two users. In contrast, our approach aims to be a more usable authentication ceremony supporting multiple participants.

3.1 Device Association

Device association methods establish a basic communication channel between two or more devices [16, 49, 50]. However, since these methods do not consider the security of the association, they do not directly aid secure contact exchange. Nonetheless, usability evaluations exist on preferred user actions or gestures for associating their device with others [13, 40, 61]. These evaluations can help us optimize the user interaction for the secure exchange of contact information, which also requires temporary device association with other participants. One such line of work is especially interesting: guessability studies with plastic surrogates to determine intuitive user actions for associating a group of devices [12, 14, 48, 51]. Instead of multiple pairwise interactions, users preferred a singular group-wise interaction, favorably by pointing their smartphones towards each other or bringing them into proximity. These results inform the design of our approach, where we use a single group-wise interaction, requiring all smartphones to be in close proximity during the exchange.

3.2 Secure Device Pairing

There is a large body of work dealing with the secure exchange of cryptographic public keys, called secure device pairing, which is very similar to our more general problem of the secure exchange of contact information. This line of work mostly focused on establishing keys between two devices [28], and previous usability studies focused mostly on this use case [13, 47, 53, 59, 62, 118, 119]. Pairing between two devices does not securely generalize to more than two parties due to new types of attacks that are possible when there are multiple participants [23, 64].

However, there is also some prior work on establishing keys between multiple devices, which roughly falls into one of four categories: (1) human-in-the-loop, (2) shared homogeneous context, (3) physical layer security, (4) OOB channels.

3.2.1 Humans-in-the-loop. There is previous research focusing on key establishment where each participant has to perform some manual action, such as entering a shared secret [1, 3, 8, 120], comparing text [65, 81, 84, 109, 120] or visual patterns [11, 56, 72, 73], or moving all devices in a correlated manner [58, 77]. Besides the obvious drawback that these schemes require substantial user interaction and do not scale well, they often cannot protect against malicious insiders [64].

3.2.2 Shared Homogeneous Context (Zero-interaction Pairing). Zero-interaction schemes try to establish a shared key based on the entropy of observable common events in the surroundings [29, 130]. While such approaches at first glance seem to fit well to our requirement of minimal user interaction, they are not suitable for user-controlled device association and take more time than is tolerable for an ad-hoc spontaneous exchange [24, 71]. These schemes were designed for a different use case, namely for pairing multiple devices owned by a single user, where the key establishment process is allowed to take its time in the background, without user involvement. Instead, group pairing is a social use case, which not only involves devices but also multiple people, and thus a group pairing protocol should not neglect the social interaction in the group.

3.2.3 Physical Layer Security. Another approach makes use of the physical laws of signal propagation on the wireless radio channel to securely establish keys [34–36, 41, 46]. A similar line of work uses a Faraday cage to prevent outside attackers from interacting with the key establishment [63, 66]. These approaches are often hard to deploy, as they either require specific firmware or hardware, making them incompatible with COTS devices, or they require additional devices to assist with the pairing process.

3.2.4 Out-of-Band Channels. One line of work studies key establishment using OOB channels, which are low-bandwidth auxiliary channels that are usually location-limited to protect the integrity of its messages [108]. Previous research considered infrared channels [5], light channels [60, 88, 99], and near-field communication (NFC) [15]. The audio channel has also been explored as a promising OOB channel for pairing two devices [38, 39, 89, 107]. As we develop this concept further for pairing multiple devices, we will elaborate on the details of the audio channel in the following section.

3.3 Acoustic Communication

The acoustic channel is an attractive choice for connecting nearby devices, thanks to its ubiquitous compatibility with smartphones by repurposing their integrated audio hardware [32]. The physical layer can be entirely software-defined [74], facilitating easy adaptation of modem properties to fit specific communication scenarios. Audio communication is often employed as a location-limited OOB channel, as it has much shorter range than radio communication and is restricted by physical barriers like walls [107]. Several proposals in the literature have explored using audio hardware for communication, including both audible and ultrasonic inaudible frequencies [33, 67, 83, 89, 98, 125].

We are only aware of one previous study by Mehrabi et al. [78] evaluating the usability of acoustic data transmission. In a lab study involving 12 participants in a pairwise data exchange scenario, they compared audible acoustic communication with Bluetooth Low Energy (BLE) and QR codes. Transaction times were lowest for audio and considerably higher for BLE. QR codes and audio had significantly higher usability than BLE, highlighting the promising nature of the acoustic OOB channel. In our work, we build on these previous works to investigate the acoustic channel's suitability and usability for connecting multiple users simultaneously.

3.4 Authentication Ceremonies in Secure Messaging Applications

Modern E2EE tools such as Signal or WhatsApp use opportunistic E2EE (also known as trust on first use (TOFU)), requiring almost no user interaction. However, TOFU cannot defend against active attacks unless users perform the *authentication ceremony*, which is the process by which users can verify their cryptographic keys [45, 117]. Authentication ceremonies usually require a human in the loop to match specific representations of the cryptographic fingerprint, such as hexadecimal strings, word phrases, or visualizations [19, 111].

The authentication ceremonies in E2EE tools are not designed for multiple people and would require multiple pairwise instantiations, which is a prohibitive amount of effort in a collaborative scenario. In addition to that, previous user studies have shown that even for just two users, current authentication ceremonies are too hard to use [2, 4] and error-prone [102, 103], which means that, in practice, people almost never perform them [26, 86, 100, 122, 123, 129]. This is especially alarming considering that the default mode without performing the authentication ceremony fails to meet users' security expectations; most users are unaware of this issue [44].

3.5 SafeSlinger

The most promising approach to date in literature is the SafeSlinger protocol by Farb et al. [23], enabling the secure exchange of contact information between multiple people. Built upon the recommendations of a previous user study on group pairing protocols [85], SafeSlinger surpasses prior human-in-the-loop approaches [11, 73] in usability, efficiency, and security, making it the state of the art in group pairing protocols. It was designed specifically for groups of people to address the security and usability problems of using pairwise protocols in such scenarios. Participants perform the following steps while using SafeSlinger:

(1) **Initialization.** Each participant must select the group size. Next, each participant sees a number on their smartphone. They must coordinate to identify and enter the lowest number among all participants on their smartphone.
(2) **Verification.** Each participant gets displayed three word phrases. They must coordinate to identify and select the phrase common to all smartphones.
(3) **Finalization.** The protocol then securely exchanges their contact information. Finally, each participant verifies that they received correct contact information from the others.

In their evaluation, SafeSlinger was significantly faster and more usable than the pairwise contact exchange method Bump,[4] as shown by a within-subjects user study ($N = 24$). However, each step of the SafeSlinger protocol requires all participants to perform multiple comparisons with all other group members. This increases effort for larger group sizes and is error-prone [53, 111]. Furthermore, SafeSlinger requires a stable Internet connection during the pairing process and relies on a third-party server to facilitate the contact exchange. These limitations prevent SafeSlinger from meeting our usability and deployability requirements (Section 2).

4 Research Questions

Based on our analysis of related work in the previous section, the main problem of previous approaches is the lack of usability for ordinary users, which gets amplified for larger groups of people due to inadequate scalability. SafeSlinger is the most promising approach so far, but lacks in scalability and deployability. The authentication ceremonies in E2EE tools, as well as most current secure device pairing methods, are solely designed for two devices. Users expect a single associating interaction for the entire group [14, 48, 50], but pairwise schemes cannot be easily generalized to multiple participants [23].

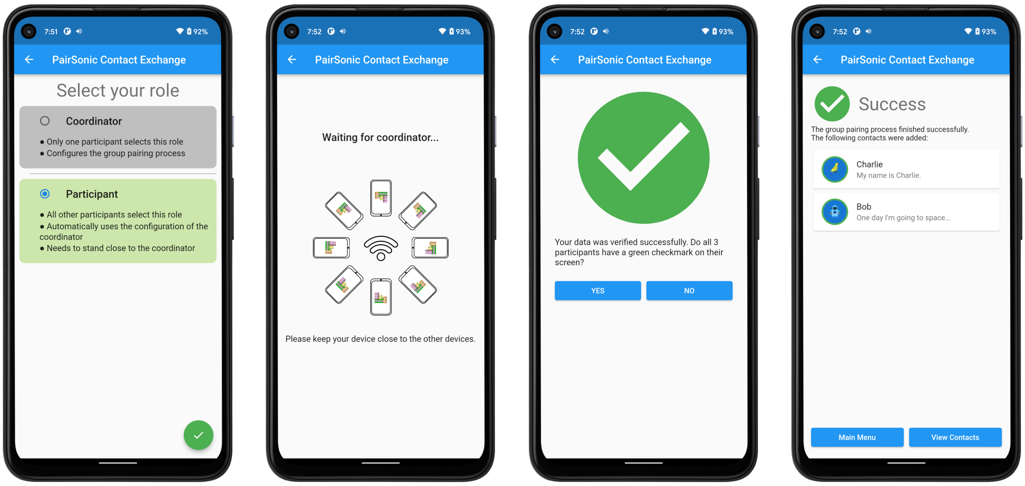
In this work, we explore the problem of secure contact exchange from a new angle, under the following research hypothesis: *users prefer a contact exchange method with minimal user interaction.* Consequently, we design a new protocol named PairSonic that minimizes user interaction, implement it on smartphones, and conduct a lab study comparing it with the current state-of-the-art approach, SafeSlinger. The research questions we address in our work are:

- **RQ1.** *Which initialization and verification steps do users prefer?*
- **RQ2.** *Which method has better usability?*
- **RQ3.** *How do users perceive the security of both methods?*
- **RQ4.** *How do users like the audible acoustic OOB channel for pairing?*

5 PairSonic: Acoustic Group Pairing Protocol

This section describes the design and implementation of our novel acoustic group pairing protocol, called *PairSonic*. We developed it to meet the usability, deployability, and security requirements outlined in Section 2. Our starting point was the design of SafeSlinger, which we identified as the

[4] Bump was discontinued in 2014.


7:51
92%
PairSonic Contact Exchange
7:52
93%
PairSonic Contact Exchange
7:52
93%
PairSonic Contact Exchange
7:52
93%
PairSonic Contact Exchange
YES
NO
Main Menu
View Contacts

Select your role

○ Coordinator

- Only one participant selects this role
- Configures the group pairing process

● Participant

- All other participants select this role
- Automatically uses the configuration of the coordinator
- Needs to stand close to the coordinator

Waiting for coordinator...

Please keep your device close to the other devices.

Your data was verified successfully. Do all 3 participants have a green checkmark on their screen?

Success

The group pairing process finished successfully.
The following contacts were added:

Charlie
My name is Charlie.

Bob
One day I'm going to space...

(a) Initialization: Role. (b) Initialization: Ready. (c) Verification. (d) Finalization.

Fig. 2. This figure shows the PairSonic contact exchange from the perspective of the *participant* role. (a) The participants begin by selecting their role. (b) They then wait for the coordinator to initiate the exchange via the acoustic OOB channel, which is shown in Figure 10. (c) Next, they verify that each group member's screen displays a green checkmark. (d) After this process, the app shows the exchanged contacts.

most promising approach to date in Section 3, as it fulfills all our security requirements – but not all usability and deployability requirements.

5.1 Protocol Overview

PairSonic, our innovative acoustic group pairing protocol, involves several steps as shown in Figure 2. First, a group of users wishing to exchange contact data initiates PairSonic on their smartphones. This brings them to the first screen, where they nominate a temporary *coordinator* (Figure 2a). The rest of the group assumes the role of *participants*, which brings their devices into the ready state (Figure 2b). The coordinator enters the total group size and confirms that all participants' smartphones are ready and nearby.[5]

Following this, the coordinator's smartphone creates a temporary ad-hoc WiFi network, transmitting a short acoustic message containing the network details to other smartphones. This allows them to automatically join the network.[6]

Next, the devices use this WiFi network to perform a cryptographic protocol, which is based on the SafeSlinger protocol. Each participant's smartphone sends a nested commitment with their contact data[7] to the coordinator's smartphone, which then disperses it to the other devices. The coordinator's smartphone generates a hash value from all participants' commitments and contact information, including their cryptographic public keys, and transmits this hash value over the acoustic OOB channel. Other smartphones are monitoring this channel and will abort if they receive any other message. Upon receipt of the correct hash value, a green checkmark appears on the smartphones, prompting users to confirm that all other smartphones display the same checkmark

[5] The coordinator's perspective of the PairSonic contact exchange is depicted in Figure 10 (appendix).
[6] PairSonic can function even if a user is already connected to a WiFi network; the user will temporarily switch to the ad-hoc network instead.
[7] The exchanged data includes the cryptographic public key and, optionally, further information such as a profile picture.

(Figure 2c). Upon confirmation, the smartphones use the WiFi network to publish success nonces. These allow others to verify the commitments and, if verification is successful, display the exchanged contacts (Figure 2d). Conversely, if the checkmark is not confirmed, they release abort nonces, terminating the protocol for all participants. This procedure ensures that all participants receive accurate and verified contact information, including authenticated cryptographic public keys.

5.2 Automating the User Interaction With Acoustic Communication

To reduce user effort during the protocol, we automated the majority of SafeSlinger's user interaction. Instead of relying on users to act as data channels via error-prone shared device interactions, we utilize an OOB channel to verify the integrity of the exchanged contact information by directly connecting the devices, thereby minimizing the required user interaction. We use an acoustic location-limited channel where smartphones repurpose their integrated speakers and microphones to exchange data within a short radius of about one meter. The smartphone speaker, rather than playing music, emits sound waves that encode information by varying their frequency or amplitude. This process has been used to connect two devices in related work (Section 3.3), and we extend this concept to multiple devices. The other smartphones capture the sound waves, decode the embedded data, and use it as an additional security layer to verify the integrity of the exchanged contact information.

The acoustic channel uniquely meets all our deployability requirements, with the added advantage that its physical layer is entirely customizable in software. This means the security of the exchanged data can be directly protected on the physical layer [89], an advantage when implementing our ad-hoc requirement without having to rely on existing security contexts such as pre-shared keys.

Another reason for choosing an acoustic OOB channel is that the acoustic data transmission only works when the users are in proximity and bring their devices close to each other, as depicted in Figure 1. The physical interaction of moving all devices close to each other matches the intuitive behavior that users would naturally like to perform to connect their devices [12, 61], similar to a handshake [64]. In contrast, WiFi or Bluetooth also work over longer distances and even through walls, failing to ensure proximity.

Furthermore, we use the acoustic OOB channel not only for verification but also to initiate the protocol, automatically connecting all participants' smartphones to the coordinator's WiFi Direct network. Both WiFi and Bluetooth require a manual pairing process before communication can occur. We automate this process using acoustic communication, making the temporary network discovery and association as part of our group pairing protocol effortless and seamless.

5.3 PairSonic Implementation

We implemented PairSonic as an Android application (app) using the Flutter framework and utilized Android's WiFi Direct API for local ad-hoc WiFi functionality. The acoustic OOB channel was built using the ggwave library.[8] The physical layer uses multi-frequency Frequency-Shift Keying (FSK) modulation with 96 evenly spaced frequencies ranging from 1875 Hz to 6375 Hz, according to ggwave's `AUDIBLE_FAST` profile. We selected an audible frequency range to assess user perception of this form of communication. Practical applications could choose between audible or inaudible frequency ranges, based on factors such as robustness, usability, and hardware compatibility.

As our implementation is a custom prototype using a novel communication stack combining acoustic communication with WiFi Direct, we conducted functionality and compatibility tests with smartphones from various manufacturers before our lab study. Our prototype worked on all tested devices supporting WiFi Direct, with a minimum required Android version 6.0 (API level 23,

[8] Data-over-sound library ggwave: https://github.com/ggerganov/ggwave

Table 1. Overview of the two group pairing protocols compared in our study.

	SafeSlinger	PairSonic
User Grouping	– peer-based	– leader-based
Device Communication	– Internet via central server	– decentral via WiFi ad-hoc – local acoustic communication
Phase 1: Initialization	– each participant counts + enters group size – each participant compares ID with others and enters lowest ID	– only leader counts + enters group size – participants bring devices into proximity
Phase 2: Verification	– each participant compares three 3-word phrases with others and selects matching phrase	– each participant confirms whether all devices show green checkmark
Phase 3: Finalization	– each participant verifies and selects which contact entries to import	– each participant verifies list of new contact entries

released in 2015): LG Nexus 5X, Huawei Nexus 6P, Google Pixel 4, Pixel 4a, Pixel 5, Pixel 6 Pro, Samsung Galaxy S20 Ultra, Galaxy S22, Oppo Reno 6, OnePlus 10 Pro, Xiaomi 11T Pro. Our design is principally also compatible with iOS, given the ubiquitous compatibility of acoustic communication. However, since iOS does not support WiFi Direct, this channel would need to be replaced with another communication layer supported on Apple devices, such as BLE or AWDL [110]. Our design does not inherently depend on any specific radio channel, making such a modification feasible.

5.4 Comparison to SafeSlinger

Our design's key differentiator is the automation of as many manual steps as possible, resulting in a group pairing protocol that is effortless for the user. Such a design allows us to verify our research hypothesis by comparing SafeSlinger with PairSonic in a user study. Table 1 lists the main differences between PairSonic and SafeSlinger, which we now discuss in more detail.

5.4.1 Usability. Whereas in SafeSlinger all participants must enter the group size in the initialization phase, in PairSonic only one participant needs to enter the group size. This smartphone then starts broadcasting initialization information to the other participants' smartphones via the acoustic OOB channel. The other participants only need to bring their devices close together. The second part of SafeSlinger's initialization phase, which requires all participants to coordinate to find and enter the lowest number shown on their devices, is completely automated in PairSonic using the acoustic OOB channel.

After exchanging their contact information, the participants need to verify that no adversary interfered with the protocol. Whereas SafeSlinger requires all participants to find a matching word phrase among three options, we automate this using an acoustic verification message in PairSonic. This reduces the information that participants need to verify manually to a single bit, which is represented by a clear green checkmark or a red warning symbol. The participants must abort the process in case not all devices show the green checkmark. In PairSonic, the devices automatically verify a second acoustic message sent by the initiating smartphone to determine whether it matches the data received over the WiFi channel. PairSonic minimizes the required user interaction compared to SafeSlinger, without reducing the security guarantees.

In the final phase of the protocol, the participants must verify that the exchanged contact entries correspond to the involved participants. This step cannot be automated as it is a social decision to determine if a contact entry matches the person standing in front of them, rather than a technical one. In SafeSlinger, each participant also has to select which contact entries to import. In PairSonic, we defer this decision by importing all contact entries, considering that users can manually delete contact entries whenever they want.

5.4.2 Deployability. In addition to usability improvements, our design also addresses the deployability limitations of SafeSlinger by decentralizing the protocol. Whereas SafeSlinger requires a stable Internet connection during the pairing process, PairSonic operates completely offline. Thus, it can function even in areas without WiFi or cellular connectivity, such as rural zones, emergency situations, or developing countries without Internet access. Our protocol does not require a third-party server because it operates in a decentralized manner, i. e., directly between the participants. This results in greater availability and resilience since our pairing process does not depend on external infrastructure. It also offers a privacy advantage, as no metadata reaches any third party.

5.4.3 Security. While PairSonic fundamentally changes both the user interaction and physical layer of SafeSlinger, it retains the same cryptographic protocol, thereby preserving the strong security benefits from SafeSlinger. PairSonic thus fulfills all security requirements of confidentiality, contact authentication, and collective pairwise security. We rely on SafeSlinger's cryptographic primitives, namely hierarchical multi-value commitments and Group Diffie-Hellman key agreement (for further information, see Farb et al. [23]). Additionally, we employ an acoustic OOB channel for location-limited verification as an extra security layer.

6 Study Methods

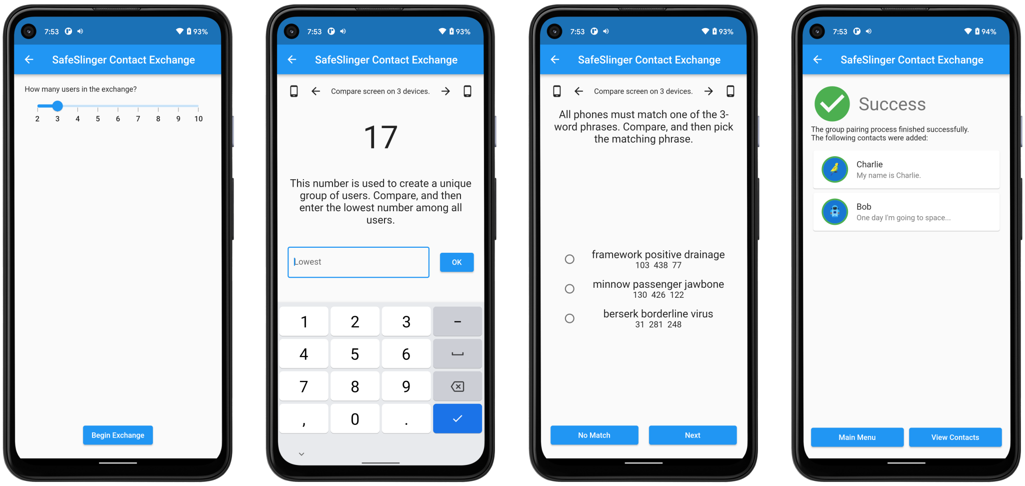
To answer our research questions on the usability and security of the group pairing protocols SafeSlinger and PairSonic, we conducted a comparative lab study featuring practical contact exchange tasks (Section 6.1), followed by a quantitative questionnaire and qualitative interviews (Section 6.2). We also discuss our study sample (Section 6.3), ethical considerations (Section 6.4), and our analysis methodology (Section 6.5).

6.1 Material

Our goal for the study setting was to resemble a typical contact exchange scenario. We built a dummy smartphone app of a fictional social networking platform consisting of user profiles, direct messaging functionality, and public message boards. This app was built using Flutter targeting Android smartphones and supports E2EE messaging using public-key cryptography. Each user has a profile consisting of a username and additional profile information such as an avatar and a profile description. Additionally, the app automatically generates public-key credentials for each user, which is used to secure the communication. The app's main menu has a button to initiate the contact exchange process, triggering either SafeSlinger or PairSonic.

We integrated SafeSlinger into our app using the official Android library[9] by Farb et al. [23], including their user interface. We only modified the look and feel to match the rest of our app.

[9] SafeSlinger Android client library: https://github.com/SafeSlingerProject/exchange-android



How many users in the exchange?

This number is used to create a unique group of users. Compare, and then enter the lowest number among all users.

All phones must match one of the 3-word phrases. Compare, and then pick the matching phrase.

○ framework positive drainage 103 438 77

○ minnow passenger jawbone 130 426 122

○ berserk borderline virus 31 281 248

Success

The group pairing process finished successfully.
The following contacts were added:

Charlie — My name is Charlie.

Bob — One day I'm going to space...

(a) Initialization: Count. (b) Initialization: Lowest ID. (c) Verification. (d) Finalization.

Fig. 3. This figure shows the SafeSlinger contact exchange, which we integrated into our app using the official Android library, only adjusting its aesthetics to align with our app's design. (a) The SafeSlinger contact exchange process begins with each participant selecting the total number of group members. (b) Next, all members compare their IDs to identify and input the smallest ID. (c) During the subsequent verification phase, members compare their word phrases to find and choose the phrase that appears on all devices. (d) After this process, the app displays the exchanged contacts.

Figure 3 shows the different steps that participants have to perform using SafeSlinger. We used Google's cloud platform AppEngine to host SafeSlinger's server component[10] written in Python.

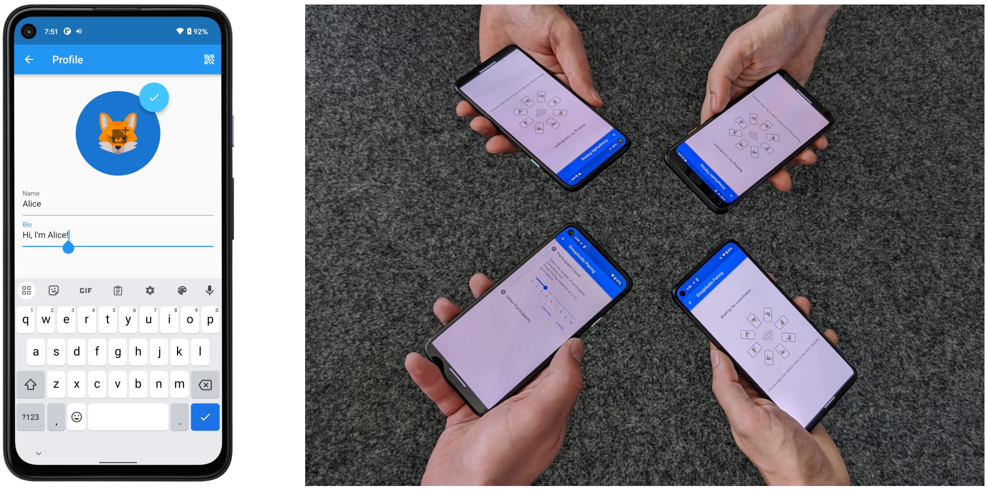
6.2 Study Design

We followed a mixed methods approach, combining usability testing within a lab study, open-ended oral interviews, and a post-test questionnaire. We performed our lab study with different groups of two to six participants, who got to exchange their contact information amongst each other in an actual group scenario. We used a within-subjects design to identify the differences between PairSonic and the state of the art (SafeSlinger), meaning that each participant performed two rounds of tasks within the same group of participants to gain hands-on experience with both systems. We used counterbalancing to vary the order in which the participants encountered these systems, which reduced practice and boredom effects. We decided for a within-subjects design due to greater statistical power than a between-subjects design. We successfully tested our study setup using a pilot study with six participants.

We conducted our lab study in a neutral meeting room, where each group of participants met the study conductor, who was always present during each trial to verify that the participants performed the tasks and to conduct the interviews. The lab setting in a neutral meeting room with the same smartphones[11] ensured consistent conditions for all participants, minimizing confounding factors. During the study, the participants filled in our questionnaire section by section after each

[10] SafeSlinger Python server: https://github.com/SafeSlingerProject/SafeSlinger-AppEngine

[11] We always used the same smartphone models for the experiment, in the following order depending on the group size: 3×Google Pixel 4a, Google Pixel 5, Samsung Galaxy S22, Oppo Reno 6.



Name
Alice

Bio
Hi, I'm Alice!

(a) Profile screen.

(b) User group exchanging their contact information using PairSonic.

Fig. 4. Our participants first created their profiles in the app, then collaboratively tested both contact exchange systems with the other group members. During PairSonic, the participants bring their devices close together.

experiment round. The study conductor additionally interviewed them about their impressions after the experiment.

6.2.1 Briefing. Participants first reviewed and signed our consent form, which outlined the study's aim, procedure, and privacy policy. We then explained the study's agenda, briefly introducing the problem of securely exchanging contact information and how current approaches fail for larger groups (approximately four minutes total). We used the same slide set for all groups to increase the internal validity of our study by ensuring that all participants received the same information. The goal of the briefing was to ensure that all participants had the same understanding of the systems they were about to try, reducing potential confounding effects from misunderstandings. Importantly, we did not disclose that we had designed one of the systems ourselves.

We then handed each participant the questionnaire (Appendix B) and a smartphone, and asked them to fill in their profile information within the app (name, avatar from a gallery of options, freely chosen text description; shown in Figure 4a). We gave each participant a choice between using their real name or a pseudonym within the app.

6.2.2 Experiment. Upon distributing the smartphones, we conducted our experiment in two rounds (A and B) corresponding to both group pairing systems. Each round was structured identically, but we alternated the order of SafeSlinger and PairSonic between each group to neutralize order effects.

Each round started with a warm-up task, where we asked the group to try the contact exchange system on their own without further guidance except the prompts within the app. Figure 4b illustrates how PairSonic's context exchange looked like for a group of four users. The study conductor noted down potential problems in the study protocol and clarified the process with the group after they successfully exchanged their contact information.

We then asked the group to use the system a second time, but remotely and unbeknownst to the participants we manipulated the experiment in such a way as to simulate an active attack on the pairing process. This resulted in the protocol failing on one of the smartphones. The study conductor observed the group reacting to this situation and asked them why they thought the process failed. We opted to simulate an attack because this provides a controllable and reproducible method to ensure all participants have a consistent experience. By simulating rather than actively disrupting the wireless communication, we also avoided technical risks to nearby networks. Our simulated attack mirrors the maximum capabilities of our theoretical adversary, who can remotely control smartphone radio communication undetected (Section 2.3). Incorporating this attack into our study helped us evaluate the usability of PairSonic, especially in adversarial scenarios our system aims to counter.

After explaining that this protocol failure could indicate a security problem, we asked the group to use the system a third time. We remotely disabled the attack simulation and told them that this time there would be no attack. At the same time, we measured the time it took the group to perform this third task from the beginning till the end, to get an assessment of how fast the system works after some practice.

After the first round, we collected all smartphones, switched to the other group pairing protocol, and conducted the second round likewise. Before concluding the study, we thanked the participants for their time. In total, each of our participants spent 60 minutes to take part in our study.

6.2.3 Questionnaire. After each of the two rounds, participants filled in the respective sections of the questionnaire to reflect on their experiences during the experiment tasks. Once both rounds were completed, they filled in sections about their preference of both system, about their usage patterns of different social or collaborative tools, and demographic questions at the end. Our corresponding questionnaire, presented in Appendix B, was used to collect the following variables:

- **Dependent variables.** The dependent variables captured our participants post-test assessment of both PairSonic and SafeSlinger. For each system, we used the System Usability Scale (SUS) [7] to answer RQ2 on usability. Because efficiency is one aspect of usability [95], the study conductor also measured the time it took the groups to complete the protocol during the third task. To answer RQ3, we asked an additional question about the perceived security of the systems, using a five-level Likert item.
- **Preference.** To answer RQ1, this binary variable captures the participants' preference: either they preferred PairSonic or SafeSlinger.
- **Collaborative/social tool usage.** To measure our participants' use of collaborative platforms and tools, we provided a list of nine distinct types of digital communication groups. For each type, we asked the participants to state whether they actively took part in them, the average number of participants in such groups, and if security was important to them in these contexts. Participants could also provide information about additional group types using free text fields. Additionally, we requested them to estimate the total number of digital communication groups they overall participate in.
- **Control variables.** We used two control variables: participants' prior experience with smartphones and their Affinity for Technology Interaction (ATI) [31]. Both variables help to assess the background of our sample and to explore potential correlations between technology affinity and the usability or security of both systems [14]. As a standardized measure, the ATI also improves the utility of our dataset for future research on usable security.
- **Demographic variables.** We collected the gender, age, education, field of study/work, and student status of our participants as demographic variables.

6.2.4 Interview. At four stages during the study, we interviewed our participants with open-ended qualitative questions to better understand their behavior and reasoning. All participants agreed to be recorded in our consent form. The study conductor also asked follow-up questions to learn

Table 2. Our participants' demographic data and control variables. We report percentages (and frequencies).

	Variable	Our Sample
Gender	Female	40% (18)
	Male	49% (22)
	Diverse	4% (2)
	No answer	7% (3)
Age	18–19	2% (1)
	20–24	49% (22)
	25–29	36% (16)
	30–34	11% (5)
	40–44	2% (1)
Student	Yes	82% (37)
	No	18% (8)
Smartphone Exp.	> 2 years	98% (44)
	≤ 2 years	2% (1)
ATI	Median	4.6
	IQR	1

	Variable	Our Sample
School Education	Intermediary secondary	4% (2)
	University entrance qualification	93% (42)
	No answer	2% (1)
Prof. Education	Vocational training	11% (5)
	Technical college	4% (2)
	Bachelor	22% (10)
	Master	22% (10)
	PhD	2% (1)
	No degree	36% (16)

more about the intricacies of their reasoning, but always asked the same set of starting questions for each group, in accordance with our research questions.

After filling in each round's questionnaire section about the prior used system, we asked participants:

- What did you like or dislike about this system?
- How secure do you think this system is?

Once participants gave feedback to each approach individually and completed the questionnaire about their general preference, we interviewed participants to evaluate the two approaches in direct comparison:

- Which system do you prefer, and why?
- ⟨*Study conductor clarifies which part of each system constitutes the initialization step.*⟩ Which initialization step do you prefer, and why?
- ⟨*Study conductor clarifies which part of each system constitutes the verification step.*⟩ Which verification step do you prefer, and why?

After the participants filled in the questionnaire section about their usage patterns for various types of collaborative tools, we asked about security:

- How important is secure communication to you in group or collaborative scenarios?

6.3 Participants

We conducted our study with 47 participants between January and March 2023. For recruitment, we used mailing lists, social media groups, word-of-mouth, and snowball sampling, both within and outside our university. Participants had to be over 18 years old to be eligible. Interested participants self-registered on an online form to choose all suitable time slots for voluntary participation. We randomly distributed our participants into groups of two to six participants based on their availability and invited them to our lab study. This resulted in five groups with four people, two

Table 3. Comparison of the dependent variables.

Dependent Variable	**SafeSlinger** (N = 45)	**PairSonic** (N = 45)	**Statistic**	**ES**
System Usability Scale (SUS)	75 (25)	85 (17.5)	$V = 597.5$ $\boldsymbol{p = .004}$	$r = -.43$
Security	4 (1)	4 (1)	$P = .5$ $p = 1$	$g = 0$
Preference	31% (14)	69% (31)	$P = .69$ $\boldsymbol{p = .016}$	$g = .19$
Completion Time (Groupwise, N = 12)	35.5 s (8.5 s)	33.5 s (10 s)	$V = 49.5$ $p = .432$	$r = -.23$

Note: for the binary preference rating, we report percentages (and frequencies), the binomial test, and Cohen's g [17] as the effect size (ES). For the other variables, we report the median (and IQR). For the SUS and completion times, we additionally report the Wilcoxon signed-rank test, and the ES estimate based on Rosenthal's method [94]; for the security ratings, we report the sign test and Cohen's g.

groups with each of two, three, and six people, and one group with five people, for a total of 12 unique groups. Each participant was compensated 15 EUR (in cash) for taking part in our study.

Two participants did not fully complete the questionnaire, so we removed them from our final sample (N = 45). Of these, 22 identified as male, 18 as female, 2 as diverse, and 3 preferred not to answer their gender. While 16 participants did not have any professional or university degree, the education level of our participants was generally high, with 10 participants having a Master's degree. Most participants (37) were students. Our participants were aged 18–44; most participants (38) were between 20 and 29 years old (Table 2).

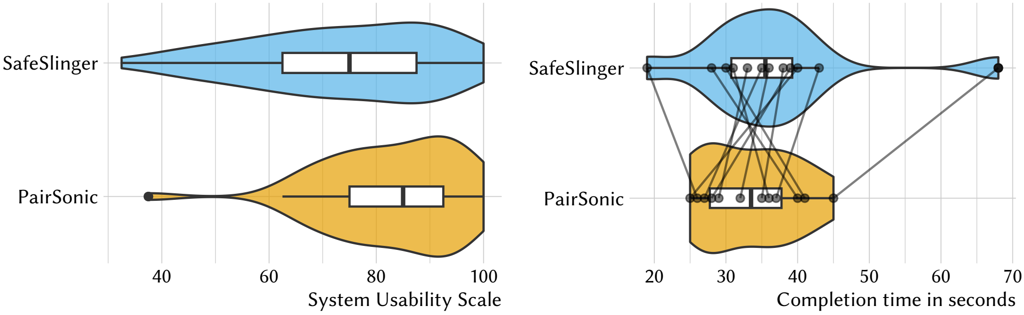
6.4 Ethical Concerns

Our university's institutional review board (IRB) reviewed and approved this study. We gathered written consent from all participants after informing them about the study's purpose and data collection in compliance with the General Data Protection Regulation (GDPR).

We did not collect any sensitive data, and we made answering demographic questions in our questionnaire optional. Participants had the voluntary choice to permit interview recordings; all agreed, so we recorded all interviews and deleted the files after transcription to safeguard participants' privacy. We stored each participant's responses pseudonymously, using sequential numbers without including any identifying information. Participants voluntarily engaged in our study and received compensation higher than our country's minimum wage as recognition for their time and effort.

6.5 Data Analysis

We used R 4.2.2 for all quantitative data analysis [92] and MaxQDA 2022 for the qualitative data coding [124]. We calculated the central tendencies and correlations to answer our research questions. We compared SafeSlinger and PairSonic using the Wilcoxon signed-rank test [128] for interval variables, the sign test [105] for ordinal variables, Pearson's chi-squared test [87] for nominal variables, and the binomial test [105] for proportions. We report descriptive statistics for our quantitative variables using the median (Mdn) and the interquartile range (IQR). We used Kendall's rank correlation coefficient [55] to determine the relationship between our control and dependent variables. For all statistical tests, we used an alpha level of .05, following the common practice in our field to balance the risk of type I and type II errors.



(a) SUS scores.

(b) Completion times.

Fig. 5. Comparison of our participants' SUS scores and completion times for SafeSlinger and PairSonic. The violin plots show a density estimation of the distributions. The boxplots show quartiles, median, and outliers.

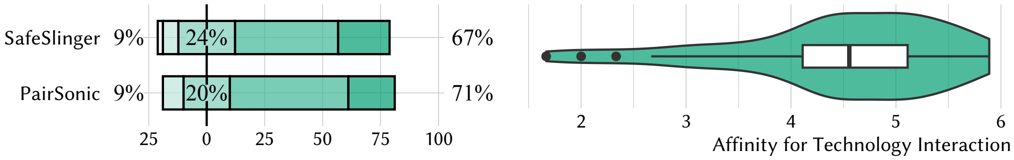
Furthermore, we conducted a qualitative analysis of the interview transcripts, performing the following qualitative coding steps: (1) We first deductively created an initial codebook covering all our research questions (Section 4). (2) Two researchers then independently coded the transcripts from the study session of a single group, to find out how well the codebook fits the participants' statements. During this separate coding process, they inductively extended the codebook with new codes to match interesting observations outside the research questions [79]. (3) The researchers then discussed and merged their two codebooks into the final codebook: preference, mental model, scenarios, suggestions/improvements, security, relevance of security in groups, initialization step, verification step, Internet requirement, third party involved, user interface, and user roles. (4) The two researchers then split the remaining transcripts and deductively coded them according to the final codebook. (5) Finally, the researchers swapped the transcripts to review and extend the other researcher's codes, and to discuss and resolve coding differences.

7 Quantitative Results

This section presents our quantitative results and the analysis of participant task performance (Table 3). The questionnaire includes quantitative scales, numeric fields, and control variables (Section 6.2.3). We studied the participants' perceptions of secure contact exchange methods regarding their usability (Section 7.1), security (Section 7.2), and preference (Section 7.3). We further examined their task completion times (Section 7.4) and general usage patterns for social and collaborative tools (Section 7.5). An analysis of the control variables is also provided (Section 7.6).

7.1 Usability

After getting hands-on experience with both group pairing protocols, the participants completed our questionnaire, including the SUS [7]. We use the Shapiro-Wilk test [101] to determine whether this interval variable is normally distributed. The SUS scores for SafeSlinger ($W = 0.96$, $p = .095$) are approximately normal, but the scores for PairSonic ($W = 0.92$, $p = .003$) are significantly non-normal distributed. Figure 5a depicts the statistics of the SUS scores per protocol. Both distributions have visible negative skew, indicating that most participants selected values on the higher end of the scale. Additionally, there is a single outlier for PairSonic. Parametric methods such as the t-test assume normal sampling distributions and can give inaccurate results in the presence of outliers [127]. We, therefore, use non-parametric statistical methods for data analysis (Section 6.5).



(a) Security scores.

(b) ATI scores.

Fig. 6. Comparison of our participants' security and ATI scores for SafeSlinger and PairSonic. The stacked bars for the security scores correspond to the five levels of agreement, ranging from *"strongly disagree"* (left) to *"strongly agree"* (right), centered at the neutral response. The percentages (left, middle, right) represent the share of negative, neutral, and positive responses, respectively. The violin plot shows a density estimation of the ATI distribution. The boxplot shows quartiles, median, and outliers.

SUS scores for SafeSlinger (Mdn = 75) differed significantly from PairSonic (Mdn = 85) according to the Wilcoxon signed-rank test, $V = 597.5$, $p = .004$. The effect size is $r = -.43$, which corresponds to a large difference according to Cohen [17]. Thus, **PairSonic showed significantly better usability than SafeSlinger.**

7.2 Security

Based on their experience during the lab tasks, we asked our participants how secure they think both systems are on a five-level Likert item. Figure 6a shows the security scores, which are ordinal assessments to the statement *"I think that this system is secure"*, ranging from *"strongly disagree"* to *"strongly agree"*. According to the sign test, which is appropriate for ordinal values [105], the security scores for SafeSlinger (Mdn = *"agree"*) did not differ significantly from PairSonic (Mdn = *"agree"*), $p = 1$, $g = 0$. Hence, **users have a similar security impression of both methods.**

7.3 Preference

After our participants tried both systems, we asked them which one they liked better. Figure 9 in the appendix shows how many participants preferred PairSonic in each study group. Most participants (69%; 31) preferred PairSonic compared to the state of the art (SafeSlinger), which is a significant difference according to the binomial test, $p = .016$. The effect size is $g = .19$, which corresponds to a medium difference according to Cohen [17]. Overall, **participants prefer PairSonic compared to SafeSlinger.**

7.4 Completion Time

As an objective measure, we determined the time it took our participants to complete each contact exchange protocol, after a brief learning period during the initial two tasks. Figure 5b shows the completion times of each individual group, connected for both protocols, to show the change within each group. The completion times for PairSonic are quite consistent and approximately normal distributed ($W = .94$, $p = .528$). There is an outlier for SafeSlinger as it took one group 68 s to complete the protocol, making the SafeSlinger completion times significantly non-normal ($W = .85$, $p = .035$). This reaffirms our choice for the robust non-parametric Wilcoxon signed-rank test. Most groups (8; 67%) were faster with PairSonic, but the completion times between SafeSlinger (Mdn = 35.5 s) and PairSonic (Mdn = 33.5 s) did not differ significantly ($V = 49.5$, $p = .432$, $r = -.23$). The completion times for both schemes showed no significant correlation with the group size ($\tau = .46, p = .052$ in both cases). In summary, **the completion times are similar for PairSonic and SafeSlinger.**

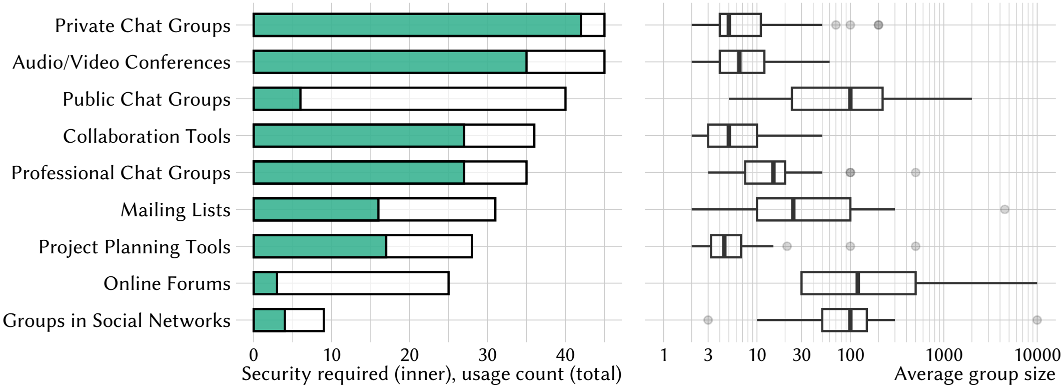


Fig. 7. The left-hand bar chart illustrates the number of participants using the given social or collaborative tools (total bar length), with the inner green bar representing those who deem security and authentication important. The right-hand boxplots depict the distribution of the average number of group members indicated by our participants for each tool type. They show quartiles, median, and outliers on a logarithmic axis. One outlier is not included: one participant gave an average online forum group size of 1.6 million.

7.5 Usage of Social and Collaborative Tools

We were also interested in understanding which types of social and collaborative tools our participants use, as this would reveal the most prominent use cases. Figure 7 summarizes our participants' responses. All participants use private chat groups, such as WhatsApp or Signal, as well as audio or video conferencing tools like Zoom. Additionally, a substantial portion partakes in public chat groups (89%), online collaboration tools like Google Docs (80%), and professional chat groups such as Microsoft Teams (78%), showing the relevance of digital collaboration applications.

We also queried our participants about their security desires for these tools, i. e., ensuring no unauthorized access. Tools requiring such security measures could benefit from group pairing protocols, such as PairSonic, indicating potential use cases.

Figure 7 demonstrates that participants seek authentication for most tools, especially private chat groups, audio/video conferencing, online collaboration tools, and professional chat groups.

Futhermore, we asked our participants to estimate the average number of group members for each type of collaboration. Groups demanding security requirements were typically smaller: the group sizes for private chat groups (Mdn = 5), online collaboration tools (Mdn = 5), and audio or video conferences (Mdn = 6) were much smaller compared to public chat groups (Mdn = 100) and online forums (Mdn = 135). Our participants are active in a significant number of groups, with a median of 70 groups (IQR = 70).

7.6 Control Variables

As a control variable, we measured our participant's ATI (Mdn = 4.6, IQR = 1) and smartphone familiarity. All 45 participants except one have been using a smartphone for more than two years, indicating general familiarity with smartphone usage.

We calculate bivariate Kendall's rank correlation coefficients in Table 4 to determine the relationship between the control and dependent variables. As shown, our participants' SUS scores, security ratings, and preferences do not significantly depend on their technology affinity or smartphone familiarity. The SUS scores for SafeSlinger significantly correlated with our participant's preference, indicating that participants who found SafeSlinger less usable preferred PairSonic instead. There was also a significant negative correlation between SafeSlinger SUS scores and completion times. Additionally, we found that a method's SUS score significantly correlated with that method's security assessment.

We noticed that the order in which our participants encountered the systems significantly correlates with the SUS scores for SafeSlinger (Figure 11), but not with the SUS scores for PairSonic or with the other variables. We discuss this potential order effect in Section 10.2.

8 Qualitative Results

Our previous section's quantitative evaluation indicated that participants found PairSonic to be more usable and preferred it over SafeSlinger. In this section, we complement the quantitative analysis with a qualitative analysis: we conducted post-test interviews with the participants to understand the reasons behind their preference. We begin by determining the factors contributing to their overall preference (Section 8.1). Subsequently, we closely compare user perceptions regarding the two main phases of group pairing protocols: the initialization phase (Section 8.2) and the verification phase (Section 8.3). Lastly, we capture our participants' perspectives on acoustic communication (Section 8.4).

8.1 Which Method Do Users Prefer Overall?

During the interviews, our participants' comments aligned with their preference for PairSonic overall. They specifically highlighted two key advantages: better *usability* and *scalability*:

> *"If you have used [PairSonic] once or twice, then it's also just much, much more convenient than [SafeSlinger]. I mean, you open the app, say »here, you're the person who sets this up«, hold the phones together, done."* (Group 7)

However, some participants argued in favor of SafeSlinger, especially for *smaller groups* where the manual comparison tasks require less effort.

Regarding *security*, the qualitative results were mixed, supporting the results of our quantitative analysis (Section 7.2). Some participants expressed that SafeSlinger's additional user interaction and complexity instilled a sense of enhanced security compared to the simpler PairSonic. Conversely, other participants favored the security of PairSonic, as it did not solely rely on potentially error-prone user interaction but also incorporated acoustic verification as an additional layer of security.

Most participants raised concerns about the *Internet requirement* of SafeSlinger, citing instances where they lacked cellular connectivity or access to public WiFi. They worried that this could limit the situations in which they can exchange their contact data. Additionally, a few participants expressed privacy concerns since the contact data was transmitted over the Internet. Despite SafeSlinger being designed to safeguard the confidentiality of exchanged information from third parties [23], the fact that the protocol relied on Internet connectivity was enough to make some participants worry about their data. In contrast, the decentralized nature of PairSonic emerged as a distinct privacy and availability advantage, as no third parties are required to facilitate the protocol.

8.2 Which Initialization Step Do Users Prefer?

The initialization step of SafeSlinger and PairSonic temporarily associates the group members' devices to exchange contact information and cryptographic public keys. However, there are distinct approaches employed by each protocol. In SafeSlinger, a peer-based approach is used, requiring each group member to input the group size and determine the lowest identification number among all members. On the other hand, PairSonic utilizes a coordinator-based approach, where only one group member needs to input the group size, and the rest of the initialization is automated through acoustic communication. It is important to note that the role of the group leader in PairSonic does not confer any special privileges and is only relevant to the contact exchange without implications for subsequent communication.

During the interviews, we asked participants to compare the two initialization steps. The majority expressed a preference for our coordinator-based initialization, although some participants argued in favor of the peer-based initialization. This observation is intriguing, particularly when considering a previous user study by Nithyanand et al. [85], which suggested a greater overall acceptance of peer-based methods. This discrepancy invites a deeper analysis to understand the reasoning behind our participants' preferences:

8.2.1 Effort. Some participants did not like the additional effort in SafeSlinger to determine the smallest identification number:

> *"It is annoying that you have to find out the smallest numbers first. Sure, you can do that relatively efficiently, but if you are in a group with a bunch of people, then maybe people just call in their numbers and then nobody has an overview anymore. "* (Group 3)

8.2.2 Scalability. Other participants highlighted a scalability issue regarding the effort required, particularly as group size increases:

> *"If it is a larger group, it is more practical to have a coordinator."* (Group 12)

Participants noted that in SafeSlinger, all group members are required to enter the group size, which can potentially lead to failures if one member miscounts.

8.2.3 Determining the Coordinator. Although no group in our study encountered difficulties in appointing a coordinator during the protocol, some participants expressed concerns about this aspect during the interviews. Even though SafeSlinger's peer-based approach requires more user interaction for each participant, it does not require the explicit social decision of determining the coordinator [49]. Interestingly, some participants preferred this trade-off:

> *"I think [SafeSlinger's initialization step] would probably be better, because it leads to fewer disputes about who is now the coordinator. Although, of course, that doesn't matter at all in the end. But in different social groups this can lead to disputes."* (Group 11)

Other participants argued that there often already is someone who proposes to exchange contact information in the first place, so there would likely be few disputes about the coordinator in practice:

> *"But if I think about it now, typically someone comes up with the idea of exchanging the contact data. That means that in the group there will already be some spokesperson who has crystallized, even if only for this specific situation. That means that, okay, I'll say here, I'm the coordinator, who else wants to participate and wants to exchange contacts with me, let's do that. And then I know that if four or five people approach me, okay, we are six people, I can set it up and off we go. That is, it actually fits relatively organically into the group dynamic, as I would expect in such a situation."* (Group 4)

8.3 Which Verification Step Do Users Prefer?

In SafeSlinger, all group members are required to compare and select a matching three-word phrase, whereas PairSonic mostly automates the verification, with group members only needing to verify that all devices display a green checkmark symbol. We now compare participants' perceptions of the verification step in the group pairing protocols.

8.3.1 Acoustic Verification Is Effortless. Most participants preferred PairSonic, arguing that it is easier to use:

> *"For the verification at the end, I think I prefer [PairSonic], because all you have to do is just make sure there is the green check. So even at that, there's no need to communicate and just raise your phone and show the green check to everyone around. And it's faster that way."* (Group 10)

8.3.2 Word Phrases Are Prone to Errors. Some users felt nervous comparing the word phrases and worried about making mistakes. Others liked the thorough verification in SafeSlinger, as it made the process feel more secure:

> *"So I think, because I have to look again carefully, that [SafeSlinger] is really secure."* (Group 9)

8.3.3 Rushing Users. However, many participants mentioned that they did not really check whether all participants have the same word phrases, and only roughly compared the words:

> *"We had the situation where only one person didn't match, but the others did. It's tempting to say right after the first match, okay, yes, it matches, I'll tick the box, great, done. And the other person is then either passed over or I don't know what happens."* (Group 4)

Some rushed the comparison due to the pressure of not wanting to keep the group waiting, particularly in the group scenario:

> *"The reason why you don't take the time to compare accurately is that you just hurry. In a larger constellation, you don't want to be the person because of whom the procedure takes longer, that you need more time to go through the individual words."* (Group 11)

Some participants also rushed with PairSonic, mistakenly assuming that the green checkmark symbol indicated a successful completion of the protocol, and thus confirmed the dialog without reading the instructions. In the subsequent discussion, the group proposed using a different symbol or adding a confirmation dialog to mitigate this issue.

8.3.4 Accessibility. Some participants mentioned accessibility problems with the word phrases due to non-native speakers, foreign accents, or use of uncommon words.

> *"The word phrases are basically nonsense in English. That is, it depends on how well it is pronounced, how well it is understood, which again becomes difficult in larger groups. And it also becomes more and more difficult depending on the countries of origin. Red and green as signal colors are actually quite international and easy to distinguish."* (Group 4)

8.4 What Do Users Think of Acoustic Communication?

As most users lack prior experience with acoustic communication, our study aimed to gauge their reactions to this new technology. Our interviews revealed a generally positive sentiment; many participants found its novelty appealing and enjoyed their devices producing melodic sounds. Past research suggests that introducing fun elements, such as gamification, can encourage users to adopt secure behaviors [44].

> *"That you can actually hear the sound is very funny. And it also somehow gives a feeling of, yes, I can hear what's happening. It feels nice to use."* (Group 6)

There are mainly two concerns that our participants had with acoustic communication, specifically related to its security and robustness.

8.4.1 Security. Some participants felt the audible acoustic verification enhanced safety, while others doubted its security, fearing potential eavesdroppers. It is important to note, however, that we designed PairSonic to maintain security even if acoustic messages are intercepted.

Some technically minded participants questioned the system's security, noting that the acoustic messages always sounded identical:

> *"I think intuitively I would find [PairSonic] a little bit more insecure because you see how they communicate. Or you hear it in this case. If you don't know how it works, what they're doing exactly, I might think, yes, if the smartphones are talking there, then maybe someone can listen in. Or record it."* (Group 12)

Despite the acoustic messages sounding similar, they always encode varying data based on the group participants. Better in-app explanations can address both of these security concerns. Alternatively, a different modulation scheme can be used to produce audibly distinct melodies.

8.4.2 Robustness and Noise Interference. During the study, unexpected device interactions led to occasional protocol restarts due to failed acoustic transmissions – something we hadn't encountered in our pre-study testing. These failures typically occurred when users placed their devices on the table during acoustic transmission, causing impulsive noise that complicated message reception.

Participants also voiced concerns about the system's performance in loud environments, such as parties or festivals. Our proof-of-concept implementation is not yet optimized to handle these challenging acoustic environments. Some participants criticized the inconsistent reliability of audible acoustic communication, expressing a preference for the more reliable SafeSlinger system. During our study, SafeSlinger's performance remained mostly consistent, but this requires a stable Internet connection, which is not available in many scenarios (Section 5.4.2). We derive recommendations for future work to improve PairSonic's usability by addressing this criticism (Section 9.10).

8.5 Alignment with Questionnaire Responses

We cross-validated whether the interview statements of our participants align with their questionnaire responses in our mixed-methods study and found general agreement between the two data sources. In the questionnaire, two groups favored SafeSlinger, two were equally split, whereas the majority (8 of 12) preferred PairSonic (Figure 9), aligning with their interview statements. For instance, group 11 unanimously preferred PairSonic in the questionnaire, also rating it higher in SUS scores. During the interviews, every member of group 11 consistently expressed a preference for PairSonic, arguing for its better usability and scalability.

9 Discussion

We now discuss the main findings of our study. First, we found users favoring PairSonic for its simplicity (Section 9.1), particularly valuing the reduced effort by automating SafeSlinger's manual verification (Section 9.2). Interestingly, though, some users perceived the more complex system as more secure (Section 9.3). We address the associated issue that complex systems tend to be more error-prone (Section 9.4) and investigate how increased transparency and education may enhance perceived security, even for simpler systems (Section 9.5). We discuss the applicability of PairSonic for different usage scenarios within the CSCW community based on our participants' comments (Section 9.6), alongside a brief security analysis (Section 9.7). We then discuss the scalability of group pairing protocols and compare which actions users have to perform in PairSonic and SafeSlinger (Section 9.8). This section concludes by addressing PairSonic's requirement that the users have to physically meet (Section 9.9), and proposing directions for future research (Section 9.10).

9.1 Users Generally Prefer Simple and Effortless Group Pairing Systems

We designed PairSonic to require less user effort than SafeSlinger, in line with our research hypothesis that users would prefer a simpler, more effortless system (Section 4). Our lab study supports this hypothesis, as participants significantly favored PairSonic over SafeSlinger, giving it higher SUS scores (Section 7). When we interviewed our participants, most explained that they prefer PairSonic because it is easier to use and requires less effort. They especially liked the automated verification step, which only requires a simple binary comparison. Furthermore, the group-wise user interaction of moving the devices closer for pairing worked very well, confirming previous research on intuitive device association gestures [14, 48, 50, 61].

9.2 Automating Verification Tasks Using Acoustic Communication

Previous user studies have shown that users find it difficult to perform authentication ceremonies with current E2EE tools like Signal or WhatsApp, because they are time-consuming and require much effort [44, 122]. Our system, PairSonic, improves the state of the art because it automates manual verification tasks such as comparing phrases, digits, or pictures. Users responded favorably to this streamlined approach in our interviews, with most participants being able to exchange contact information in less than 35 s.

Our system utilizes an acoustic OOB channel for automated verification, which offers several unique advantages over other wireless channels that modern smartphones support: unlike WiFi or Bluetooth, it is limited to close proximity. This enhances usability by reinforcing the intuitive gesture of bringing devices closer together to share contacts, and also improves security by making distant attacks more challenging. In comparison to NFC or QR codes, the acoustic OOB channel can easily accommodate multiple devices – a necessary feature for group scenarios.

Moreover, the acoustic OOB channel is a software-defined physical layer, unlike WiFi or Bluetooth whose physical layers cannot be customized in smartphones. This adaptability enables advanced physical layer security techniques, which can protect the group pairing protocol against sophisticated attacks, adding an extra layer of security [89]. This reduces the need for meticulous user behavior during verification. By safeguarding the acoustic channel at the physical layer, we can further enhance both security and usability of our system, while meeting all our deployability requirements (Section 2).

9.3 Complex and Effortful Systems Can Feel More Secure

While most participants favored PairSonic, our quantitative analysis revealed no significant difference in how users perceive the security of PairSonic and SafeSlinger (Section 7). In fact, the interviews even indicate that some users have less trust in PairSonic due to its simplicity. They felt that the added complexity of SafeSlinger made the system seem more secure:

> *"I think the perceived security with [SafeSlinger] was higher because, well it sounds stupid, but because you enter so many things. [...] But I thought, because it took much longer, that it's probably more secure."* (Group 4)

Our research hypothesis emphasizes minimal user interaction, but our interviews revealed a potential issue for security systems like PairSonic. Its effortless nature can make it seem too simple to be secure, almost like magic, leading to distrust. When a system automatically handles security, users may feel a loss of control, viewing the system as a black box that they can't understand [68, 96, 97]. They might also equate complexity with security due to everyday experiences like complex password requirements or cumbersome multi-factor authentication [54].

This unexpected finding emphasizes a dilemma: a system needs to be simple and effortless for usability, but if it's too simple and requires minimal effort, users may doubt its security. Our results highlight the challenging trade-off that security engineers must balance – not just optimizing usability, deployability, and security, but also the *perceived* security.

9.4 Error-Prone Collaborative User Interaction

User interaction requiring substantial effort often leads to errors. This poses a particular problem for systems like SafeSlinger, which depend on meticulous user behavior for security, such as the verification step involving comparison of word phrases. We noticed in our study that many participants did not carefully compare SafeSlinger's word phrases on each device – often checking only the first word, a behavior referred to as *"rushing users"*. This tendency has also been observed in a previous user study by Tan et al. [111].

As another example, in some larger groups, our participants only compared SafeSlinger's words with adjacent group members and accepted the matching word phrase without confirming this with the rest of the group. In other groups, a single member took charge of coordinating the verification step. However, once a few members confirmed a match, the group often selected that phrase based on majority agreement, potentially neglecting quieter members who did not share the same phrase. Differently to SafeSlinger, the security of PairSonic does not heavily depend on user diligence, thus offering systematic security advantages against active adversaries.

In contrast to the traditional pairwise authentication schemes described in Section 3, in our collaborative situation the whole group works together to achieve a common security goal. We made the interesting observation that some users checked and helped each other, leading to fewer overall failures. This phenomenon was also observed in previous user studies in collaborative scenarios [48, 54].

However, Kainda et al. found that the primary source of failure was poor communication among group members [54]. We noted similar issues in some groups, particularly where participants were more reserved and hesitant to voice problems or discrepancies in word phrases. These findings highlight that a system requiring less user interaction and communication results in a more reliable and less error-prone solution.

9.5 Transparency and Comprehensibility Can Improve Perceived Security

Technical users typically like to understand how and why a system works, whereas non-technical users need convincing evidence that a system is secure, without necessarily needing to understand all technical details [25]. Our interviews indicated that most users were not familiar with acoustic communication, especially if it is audible, and would benefit from better explanations within the app. Systems like SafeSlinger, which depend on third parties over the Internet, could also enhance user experience with increased transparency and education regarding data access, addressing users' privacy concerns. CSCW research during the COVID-19 pandemic has shown that privacy-related trust in the application provider plays an important role in the user's willingness to use online collaborative tools [82].

9.6 Usage Scenarios

There are a number of collaboration scenarios that have been proposed by CSCW researchers, where we argue that PairSonic could improve the usability of the group formation process. Previous research focused mainly on remote collaboration using tools like groupwork platforms [82], crowdsourcing [43], and mobile devices [131]. Additionally, there are separate lines of work studying computer-supported collaboration for creative group work and education [10], for networking and meeting new collaborators [93, 116, 126], and within the context of spontaneous ad-hoc interaction [18, 22]. A common challenge in these studies is connecting people and their devices, often using technologies like QR codes that struggle with usability and scalability in larger groups (Section 1.1). PairSonic offers a scalable, user-friendly drop-in replacement to these legacy pairing technologies, providing a more efficient solution for associating devices in group settings.

There are also usage scenarios potentially involving larger groups of participants. For example, Tolmie et al. [114] and Fosh et al. [30] explored collaborative interactions in cultural sites such as museums, suggesting that connecting visitors, possibly in ad-hoc groups with a tour guide, enhances engagement with both the group and the exhibits. PairSonic can facilitate this by connecting visitors through their own devices, avoiding the need for museum rental devices. Our testing confirms PairSonic's effectiveness for larger groups, including those with more than 10 participants, when their devices are close and in low-noise environments. Museums are ideal for PairSonic, especially using the inaudible frequency range instead of audible acoustic signals (Section 5.3). The quiet setting ensures reliable acoustic communication, and its inaudibility ensures it does not disturb other visitors. Additionally, like QR codes, the acoustic channel can transmit small data packets, which could further improve engagement with exhibitions. For example, exhibitions could emit an audio beacon offering additional information to nearby visitors, benefiting from the deployability advantages of the acoustic channel compared to alternatives such as Bluetooth or WiFi (Section 5.4.2).

Our study participants identified several scenarios where PairSonic could serve as a secure method for exchanging contacts, such as private Signal chat groups, Zoom video conferences, and online collaboration tools (Section 7.5). These situations usually have higher security requirements, often relating to personal, familial, or professional collaborations. Many interviewees shared their preferences for online security in social contexts:

> *"For me, it depends on how many people are in the group and who is in the group. If it's a group where I'm only with friends, where maybe more sensitive data is discussed, then [security] is the most important requirement of all, I would say."* (Group 12)

Others felt knowing the other participants is essential for effective online collaboration:

> *"I find it very awkward to write in groups where I don't know who else can read it. That's why I usually just don't participate at all."* (Group 6)

Our results align with previous CSCW research on the security and privacy needs of users, which found that especially vulnerable types of users and users discussing sensitive topics have a substantial need for interpersonal trust and knowing the other participants [104]. Similarly, our participants also named specific scenarios where they saw the potential for securely exchanging contact information using PairSonic. These included festivals, exhibitions, conventions, conferences, but also everyday situations when meeting new people, like meeting new people through friends. Some expressed interest in using PairSonic with non-technical users, anticipating potential difficulties these users might face with SafeSlinger:

> *"You can also use [PairSonic] with people who normally would need further assistance. Instead it's enough if one person is the coordinator, and then the others just have to click yes or no. I think that's actually quite good, because it takes the hurdle out for people who are not so tech-savvy, if you tell them that they don't have to do anything except click yes or no once."* (Group 6)

9.7 PairSonic Inherits SafeSlinger's Strong Security

While our primary focus is on the usability of PairSonic, and a comprehensive security analysis is not within this work's scope, we do provide a brief discussion on its security aspects. We adhered to the security best practice of not unnecessarily creating an entirely new cryptographic protocol from scratch, opting instead to rely on the established security guarantees of the SafeSlinger protocol whenever possible. Consequently, PairSonic protects against all types of attacks mentioned in the SafeSlinger paper, such as threats from malicious bystanders, impersonation, sybil/hidden node attacks [21, 54], and Group-in-the-Middle attacks [64].

Unlike the SafeSlinger protocol, PairSonic automates information exchange stages that previously required manual effort, using the acoustic out-of-band channel for this purpose. The acoustic channel must have the same security properties as the human-mediated channel in order to not degrade PairSonic's security. Our adversary model assumes the acoustic channel's authenticity, implying that adversaries cannot tamper with it. This assumption is supported by previous studies on wireless authentication, which highlight the physical constraints of sound waves, such as their significantly shorter range compared to radio communication and their inability to penetrate physical barriers like walls [5, 107, 108]. Furthermore, the authenticity of the audio channel can be explicitly ensured using physical layer security techniques [89], whose assumptions and system model align with PairSonic's protocol. Future work could incorporate them to enhance or replace the current ggwave physical layer used in our protocol.

In our adversary model, we assume that an adversary can eavesdrop on the audio channel, enabling them to gain information about the ad-hoc WiFi network and potentially interfere with it. However, any WiFi interference attempt by the adversary to alter the group's contact information would be futile, as PairSonic is designed to detect such modifications and terminate the protocol during the acoustic verification phase. The use of the acoustic out-of-band channel in PairSonic also enhances security by supporting longer hash values than SafeSlinger, reducing the likelihood of hash collisions. The SafeSlinger protocol employs 24-bit hash values as Short Authentication Strings. Increasing the entropy of these hashes would negatively impact usability, as it would require users to manually compare a greater number of words or more complex phrases. In contrast, PairSonic streamlines this process by automatically transmitting the hash value via the acoustic channel. This method allows for longer hash values without user involvement and is less susceptible to security-critical user errors, a concern highlighted in our study (Section 9.4).

9.8 Scalability

Besides usability, one motivation for PairSonic was improved scalability. PairSonic requires less interaction between participants by design, making it better suited for larger groups of users than SafeSlinger. To illustrate this, Figure 8 shows timelines of PairSonic compared to SafeSlinger for a typical user group during our study. From a user's perspective, each protocol consists of

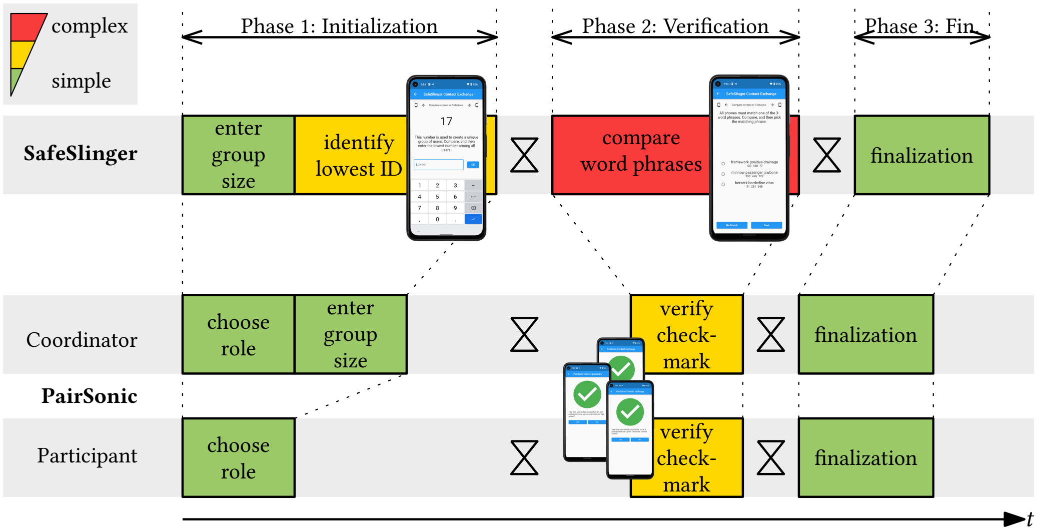


Fig. 8. Qualitative timelines show which actions users have to perform during all three phases of the contact exchange process (Table 1). The symbol ⧖ indicates waiting time. The yellow and particularly the red user actions are more complex than the green actions and take more time for larger groups, showing that PairSonic scales better than SafeSlinger.

phases where they have to wait, but also phases requiring them to read, choose an option, compare information, or enter a number: these are phases requiring manual actions. The total cumulative duration that users have to diligently perform these actions (i.e., excluding waiting times) is lowest for PairSonic with the participant role, followed by the coordinator role, with SafeSlinger requiring the longest attentive time for all users.

The figure also highlights the protocol phases that take noticeably more time for larger groups. SafeSlinger contains two such phases: identifying and entering the lowest identification number amongst all participants, and the word phrase comparison. PairSonic, however, only has a single phase scaling with the group size, namely determining that every device shows a large checkmark symbol, a task that is comparatively much simpler than the tasks in SafeSlinger, since it only requires a quick glance at the smartphones. In summary, PairSonic requires less attentive user duration, making it well-suited also for larger groups.

Larger groups further reinforce the issue of error-prone collaborative user interaction that we discussed previously (Section 9.4). With SafeSlinger, larger group sizes involve more manual text comparisons, which not only raises the likelihood of mistakes, but also reinforces the problem of *rushing users*. We observed that some users felt pressured by larger groups and wanted to avoid keeping the group waiting (Section 8.3.3). PairSonic was less affected by group size, but our proof-of-concept implementation struggled with acoustic noise interference (Section 8.4), which might occur more often with more participants.

Contrary to our expectation, we did not find a significant difference in completion times between both protocols, even though most groups in our study were faster with PairSonic. The completion times of SafeSlinger were faster than expected and can be attributed to many participants not thoroughly comparing the word phrases on each device, as explained earlier (Section 9.4). While this explains the comparable completion times, it degrades SafeSlinger's security in practice, as partial comparisons involve less entropy [111]. We also note that the scalability advantage of PairSonic is likely more pronounced for participants who take longer with SafeSlinger's word comparison, such as older users, users with bad eyesight, or shy participants.

9.9 Physical Meetings

PairSonic assumes that all users are co-located, making it suitable for offline and online collaboration, provided the users meet in-person first (Section 9.6). This limitation is by design and arises from our requirement that the system functions without relying on any central infrastructure and without necessitating any pre-established security context (Section 2.1). While SafeSlinger can operate remotely over the Internet, it violates these requirements by assuming the existence of another trusted communication channel to authenticate each other. However, such a channel typically does not exist yet, which is precisely why the group pairing process is needed to begin with.

This requirement for physical meetings means that to securely collaborate with a new contact using PairSonic, you must first meet them in person. In larger collaborative groups, it can often be impractical to meet with every new member as they join the group. However, PairSonic should be seen as just one component of a broader key exchange process, which cannot address the whole key management lifecycle by itself [6]. To address this limitation, PairSonic could be integrated into a more comprehensive key management system like the decentralized web of trust (e.g., OpenPGP [27]). This integration could replace the traditional manual verification of PGP keys in person or as part of keysigning parties [9, 115].

In such a system, transitive trust can be utilized, meaning that you might choose to trust the contact list of someone you have already verified, rather than verifying every new contact yourself. As a result, a new group member would only need to meet and exchange keys with one existing group member. This member then signs and forwards the authenticated contact information to the rest of the group. In this scenario, the initial group members use PairSonic to establish an initial security context, which then serves as a base for other key management protocols that do not require physical interaction.[12]

9.10 Future Work

Our study identified the primary weakness of PairSonic as the limited robustness of the acoustic OOB channel in noisy environments, reducing the potential use cases for contact exchange. Whereas our paper focused on the usability evaluation of PairSonic's prototype, the interviews have shown that future work is needed to improve the OOB channel's reliability concerning realistic noise sources, exploring both audible and inaudible frequency ranges. Additional future work should include a comprehensive evaluation of the acoustic channel's functionality and performance for various environments, to determine the practical limits of this novel technology. The acoustic physical layer in the user study by Mehrabi et al. [78] was more reliable, suggesting that a better design can address this issue.

Additionally, enhancements to the user interface could further increase PairSonic's usability, reflecting findings from previous studies on other authentication ceremonies [123, 129]. During our study, we observed that some participants rushed the verification step in PairSonic. This was not to bypass a tedious task, as seen in previous approaches, but because they misconstrued the green checkmark symbol as a sign of successful completion (Section 8.3). Using a different symbol, such as a question mark, could signal the need for user involvement in security verification, potentially resolving this issue. Alternatively, a confirmation dialog could be introduced following the initial

[12] Similar established strategies could be employed to deal with other aspects of the key management lifecycle, such as the loss or change of keys. For example, OpenPGP typically manages key replacement by generating a new keypair, signing it with the old key, and then issuing a revocation certificate for the old key. This process ensures a seamless transition from the old to the new key while maintaining security and trust within the system.

prompt to avoid unintentional acceptance. Future research could validate these enhancements by complementing our lab study with an additional field study analyzing the long-term usage of group pairing protocols in real-world conditions.

Lastly, tech-savvy users expressed curiosity about PairSonic's operation, its security properties, and how the acoustic channel protects security. Including an in-app help menu with detailed explanations could assist users with questions about the system's functionality or security. Similarly, Herzberg et al. [44] suggested that for an authentication ceremony to be effective, it must not only be usable, but users also need to understand its necessity.

10 Limitations

This section addresses the limitations of our work due to our recruitment process and study design.

10.1 Recruitment

Our sample was relatively young, mainly comprising students and participants aged 20-29. Rather than selectively sampling for a representative distribution, we controlled for factors potentially influencing usability (Section 7.6). Neither our control variables nor age significantly correlated with usability, security, or preference scores.

However, a previous study by Kobsa et al. [59] reported significant differences in completion times between younger and older participants in pairing protocols. Given this, our younger sample may underestimate completion times for the broader population. Particularly, the scalability difference between SafeSlinger's word comparison and our automated verification could be more notable among older users.

10.2 Order Effect in Our Within-Subjects Design

In our correlational analysis (Section 7.6), we found some evidence of a potential order effect impacting the SUS scores. Participants who first tried PairSonic generally rated SafeSlinger lower (Mdn = 67.5), possibly due to the extra effort SafeSlinger requires (contrast effect). However, we found no evidence of a contrast effect in reverse: participants who started with SafeSlinger typically assigned high initial SUS scores (Mdn = 86.25), likely because they had to fill in the SUS questionnaire before trying the second system. Although these participants generally rated PairSonic higher after using it, the high initial score for SafeSlinger limited the relative improvement. We counterbalanced the sequence of systems in our study design to minimize potential order effects. Regardless of the order of exposure to the systems, participants generally gave PairSonic higher SUS scores.

10.3 Anchor Effect in Groups

We conducted our study with groups of multiple participants. During the group interviews, we sequentially asked the same question to each participant, which may have triggered an anchoring effect, as participants' responses could have influenced each other. We noticed that some participants picked up arguments from previous respondents, but we also noticed instances of disagreement; most groups did not have a unanimous opinion (Figure 9). Our study design partially mitigates the anchoring effect by having participants complete questionnaire sections before the corresponding interviews. This approach allowed them to reflect on the tasks and form their own opinions prior to the discussion. Given the cooperative nature of the group pairing process, conducting the experiment within a group of participants appears justified.

10.4 Lab Study

Our participants tested PairSonic and SafeSlinger three times each in our lab. Although a lab setting does not fully replicate real-world conditions [16], it offers a consistent and reproducible environment, which minimizes confounding factors. While a field study might have more accurately represented real-world usage, the presence of uncontrollable environmental and external variables could have adversely impacted the quality of the data. Given that PairSonic is still a proof-of-concept implementation, the lab environment was particularly advantageous for observing our participants' interactions and reactions to the apps in real time and to discover problems in the user experience. While observing participants might influence their behavior, this setup enabled us to identify an unexpected user interaction that resulted in protocol failures (Section 8.4). Such issues, which were not evident during our testing, would have been challenging to detect in a field study. In addition, our study design strategically interleaved practical tasks with post-test questionnaires and interviews. This allowed us to promptly capture our participants' thoughts on each system before they experienced the other, a process more feasibly managed in the lab setting. We also wanted to specifically evaluate how our participants would handle potential active attacks on the pairing process. Simulating such attacks in a controlled manner is much easier in a lab environment.

11 Conclusion

We introduced a novel, practical method for secure contact information exchange, known as *PairSonic*, and compared it to the state-of-the-art system SafeSlinger [23] in a within-subjects lab study ($N = 45$). Our system's primary advantage is that it reduces user effort by automating laborious manual tasks using acoustic communication. Our results reveal that while users value ease in authentication systems, they often associate more complex systems with higher security.

RQ1. *"Which initialization and verification steps do users prefer?"* Our participants showed a significant preference for PairSonic over SafeSlinger. In the interviews, most preferred our leader-based initialization method and the acoustic verification step.

RQ2. *"Which method has better usability?"* PairSonic is significantly more usable than SafeSlinger. Our participants appreciated the seamless contact exchange and the effortless scalability to larger groups. However, there's a need to enhance the reliability of the OOB channel.

RQ3. *"How do users perceive the security of both methods?"* We observed no significant difference in the security ratings for both methods. Although minimizing user interaction improved usability, it surprisingly decreased perceived security for some participants, according to our interviews. Many users, drawing from their experiences, associate security with complexity, underscoring a tricky trade-off between usability and perceived security.

RQ4. *"How do users like the audible acoustic OOB channel for pairing?"* Our study indicates that acoustic communication is a promising, user-friendly method for data exchange between nearby devices, reinforcing the intuitive action of bringing devices close for association. However, future work should focus on enhancing the robustness of the physical layer in everyday scenarios.

Availability

Together with this paper, we provide an overview of the PairSonic project online at https://seemoo.de/s/pairsonic. PairSonic is available as open-source software on GitHub: https://github.com/seemoo-lab/pairsonic. The PairSonic app will also be presented as a demo at CSCW 2024 [91].

Alongside this paper, we release a replication package that includes our evaluation scripts and the pseudonymized dataset from our study [90]. This dataset contains usability, security, and preference scores, completion times, reported usage of nine types of social and collaborative tools, and seven demographic and control variables, for each of our 45 participants.

Acknowledgments

This work has been funded by the LOEWE initiative (Hesse, Germany) within the emergenCITY center [LOEWE/1/12/519/03/05.001(0016)/72]. We thank the anonymous reviewers for their helpful suggestions. Furthermore, we thank Maximilan Gehring for his contributions to the initial PairSonic prototype, and Lea Holaus for drawing the illustration in Figure 1. We also acknowledge SafeSlinger's substantial innovation over previous pairwise systems, which greatly inspired us in developing PairSonic.

References

[1] Michel Abdalla, Emmanuel Bresson, Olivier Chevassut, and David Pointcheval. 2006. Password-Based Group Key Exchange in a Constant Number of Rounds. In *Public Key Cryptography - PKC 2006 (Lecture Notes in Computer Science)*, Moti Yung, Yevgeniy Dodis, Aggelos Kiayias, and Tal Malkin (Eds.). Springer, Berlin, Heidelberg, 427–442. https://doi.org/10.1007/11745853_28

[2] Ruba Abu-Salma, M. Angela Sasse, Joseph Bonneau, Anastasia Danilova, Alena Naiakshina, and Matthew Smith. 2017. Obstacles to the Adoption of Secure Communication Tools. In *2017 IEEE Symposium on Security and Privacy (SP)*. 137–153. https://doi.org/10.1109/SP.2017.65

[3] N Asokan and Philip Ginzboorg. 2000. Key Agreement in Ad Hoc Networks. *Computer Communications* 23, 17 (Nov. 2000), 1627–1637. https://doi.org/10.1016/S0140-3664(00)00249-8

[4] Hala Assal, Stephanie Hurtado, Ahsan Imran, and Sonia Chiasson. 2015. What's the Deal with Privacy Apps? A Comprehensive Exploration of User Perception and Usability. In *Proceedings of the 14th International Conference on Mobile and Ubiquitous Multimedia (MUM '15)*. Association for Computing Machinery, New York, NY, USA, 25–36. https://doi.org/10.1145/2836041.2836044

[5] Dirk Balfanz, Diana K. Smetters, Paul Stewart, and H. Chi Wong. 2002. Talking to Strangers: Authentication in Ad-Hoc Wireless Networks. In *NDSS Symposium*.

[6] Matt Blaze, Joan Feigenbaum, and Jack Lacy. 1996. Decentralized Trust Management. In *Proceedings 1996 IEEE Symposium on Security and Privacy*. 164–173. https://doi.org/10.1109/SECPRI.1996.502679

[7] John Brooke. 1996. SUS: A Quick and Dirty Usability Scale. *Usability Evaluation in Industry* (1996).

[8] Mike Burmester and Yvo G. Desmedt. 1997. Efficient and Secure Conference-Key Distribution. In *Security Protocols (Lecture Notes in Computer Science)*, Mark Lomas (Ed.). Springer, Berlin, Heidelberg, 119–129. https://doi.org/10.1007/3-540-62494-5_12

[9] Germano Caronni. 2000. Walking the Web of Trust. In *Proceedings IEEE 9th International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WET ICE 2000)*. 153–158. https://doi.org/10.1109/ENABL.2000.883720

[10] Bin Chen, Koki Hatada, Keiju Okabayashi, Hiroyuki Kuromiya, Ichiro Hidaka, Yoshiharu Yamamoto, and Kazumasa Togami. 2019. Group Activity Recognition to Support Collaboration in Creative Digital Space. In *Companion Publication of the 2019 Conference on Computer-Supported Cooperative Work & Social Computing (CSCW '19 Companion)*. Association for Computing Machinery, New York, NY, USA, 175–179. https://doi.org/10.1145/3311957.3359471

[11] Chia-Hsin Owen Chen, Chung-Wei Chen, Cynthia Kuo, Yan-Hao Lai, Jonathan M. McCune, Ahren Studer, Adrian Perrig, Bo-Yin Yang, and Tzong-Chen Wu. 2008. GAnGS: Gather, Authenticate 'n Group Securely. In *Proceedings of the 14th ACM International Conference on Mobile Computing and Networking (MobiCom '08)*. Association for Computing Machinery, New York, NY, USA, 92–103. https://doi.org/10.1145/1409944.1409957

[12] Ming Ki Chong and Hans Gellersen. 2011. How Users Associate Wireless Devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11)*. Association for Computing Machinery, New York, NY, USA, 1909–1918. https://doi.org/10.1145/1978942.1979219

[13] Ming Ki Chong and Hans Gellersen. 2012. Usability Classification for Spontaneous Device Association. *Personal and Ubiquitous Computing* 16, 1 (Jan. 2012), 77–89. https://doi.org/10.1007/s00779-011-0421-1

[14] Ming Ki Chong and Hans W. Gellersen. 2013. How Groups of Users Associate Wireless Devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. Association for Computing Machinery, New York, NY, USA, 1559–1568. https://doi.org/10.1145/2470654.2466207

[15] Ming Ki Chong, Fahim Kawsar, and Hans Gellersen. 2011. Spatial Co-Location for Device Association: The Connected Object Way. In *Proceedings of the 2011 International Workshop on Networking and Object Memories for the Internet of Things (NoME-IoT '11)*. Association for Computing Machinery, New York, NY, USA, 21–26. https://doi.org/10.1145/2029932.2029941

[16] Ming Ki Chong, Rene Mayrhofer, and Hans Gellersen. 2014. A Survey of User Interaction for Spontaneous Device Association. *Comput. Surveys* 47, 1 (May 2014), 8:1–8:40. https://doi.org/10.1145/2597768

[17] Jacob Cohen. 1992. A power primer. *Psychological bulletin* 112(1) (1992), 155–159.

[18] Adrian A. de Freitas and Anind K. Dey. 2015. The Group Context Framework: An Extensible Toolkit for Opportunistic Grouping and Collaboration. In *Proceedings of the 18th ACM Conference on Computer-Supported Cooperative Work & Social Computing (CSCW '15)*. Association for Computing Machinery, New York, NY, USA, 1602–1611. https://doi.org/10.1145/2675133.2675205

[19] Sergej Dechand, Yasemin Acar, Dominik Schurmann, Sascha Fahl, Karoline Busse, and Matthew Smith. 2016. An Empirical Study of Textual Key-Fingerprint Representations. *USENIX Security* (2016).

[20] Danny Dolev and Andrew Yao. 1983. On the Security of Public Key Protocols. *IEEE Transactions on Information Theory* 29, 2 (March 1983), 198–208. https://doi.org/10.1109/TIT.1983.1056650

[21] John R. Douceur. 2002. The Sybil Attack. In *Peer-to-Peer Systems (Lecture Notes in Computer Science)*, Peter Druschel, Frans Kaashoek, and Antony Rowstron (Eds.). Springer, Berlin, Heidelberg, 251–260. https://doi.org/10.1007/3-540-45748-8_24

[22] W. Keith Edwards, Mark W. Newman, Jana Z. Sedivy, Trevor F. Smith, Dirk Balfanz, D. K. Smetters, H. Chi Wong, and Shahram Izadi. 2002. Using Speakeasy for Ad Hoc Peer-to-Peer Collaboration. In *Proceedings of the 2002 ACM Conference on Computer-Supported Cooperative Work (CSCW '02)*. Association for Computing Machinery, New York, NY, USA, 256–265. https://doi.org/10.1145/587078.587114

[23] Michael Farb, Yue-Hsun Lin, Tiffany Hyun-Jin Kim, Jonathan McCune, and Adrian Perrig. 2013. SafeSlinger: Easy-to-Use and Secure Public-Key Exchange. In *Proceedings of the 19th Annual International Conference on Mobile Computing & Networking (MobiCom '13)*. Association for Computing Machinery, New York, NY, USA, 417–428. https://doi.org/10.1145/2500423.2500428

[24] Habiba Farrukh, Muslum Ozmen, Faik Ors, and Berkay Celik. 2023. One Key to Rule Them All: Secure Group Pairing for Heterogeneous IoT Devices. In *IEEE S&P 2023*. https://www.computer.org/csdl/proceedings-article/sp/2023/933600b693/1Js0EfOauaI

[25] Matthias Fassl, Lea Theresa Gröber, and Katharina Krombholz. 2021. Exploring User-Centered Security Design for Usable Authentication Ceremonies. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems (CHI '21)*. Association for Computing Machinery, New York, NY, USA, 1–15. https://doi.org/10.1145/3411764.3445164

[26] Matthias Fassl and Katharina Krombholz. 2023. Why I Can't Authenticate — Understanding the Low Adoption of Authentication Ceremonies with Autoethnography. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (CHI '23)*. Hamburg, Germany. https://publications.cispa.saarland/3895/

[27] Hal Finney, Lutz Donnerhacke, Jon Callas, Rodney L. Thayer, and Daphne Shaw. 2007. *OpenPGP Message Format*. Request for Comments RFC 4880. Internet Engineering Task Force. https://doi.org/10.17487/RFC4880

[28] Mikhail Fomichev, Flor Alvarez, Daniel Steinmetzer, Paul Gardner-Stephen, and Matthias Hollick. 2018. Survey and Systematization of Secure Device Pairing. *IEEE Communications Surveys & Tutorials* 20, 1 (2018), 517–550. https://doi.org/10.1109/COMST.2017.2748278

[29] Mikhail Fomichev, Max Maass, Lars Almon, Alejandro Molina, and Matthias Hollick. 2019. Perils of Zero-Interaction Security in the Internet of Things. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 3, 1, Article 10 (mar 2019), 38 pages. https://doi.org/10.1145/3314397

[30] Lesley Fosh, Steve Benford, and Boriana Koleva. 2016. Supporting Group Coherence in a Museum Visit. In *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing (CSCW '16)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/2818048.2819970

[31] Thomas Franke, Christiane Attig, and Daniel Wessel. 2019. A Personal Resource for Technology Interaction: Development and Validation of the Affinity for Technology Interaction (ATI) Scale. *International Journal of Human–Computer Interaction* (2019).

[32] Vadim Gerasimov and Walter Bender. 2000. Things That Talk: Using Sound for Device-to-Device and Device-to-Human Communication. *IBM Systems Journal* 39, 3.4 (2000), 530–546–530–546. https://doi.org/10.1147/sj.393.0530

[33] Pascal Getreuer, Chet Gnegy, Richard F. Lyon, and Rif A. Saurous. 2018/june. Ultrasonic Communication Using Consumer Hardware. *IEEE Transactions on Multimedia* 20, 6 (2018/june), 1277–1290. https://doi.org/10.1109/TMM.2017.2766049

[34] Nirnimesh Ghose, Loukas Lazos, and Ming Li. 2017. HELP: Helper-Enabled in-Band Device Pairing Resistant against Signal Cancellation. In *Proceedings of the 26th USENIX Conference on Security Symposium (SEC'17)*. USENIX Association, USA, 433–450.

[35] Nirnimesh Ghose, Loukas Lazos, and Ming Li. 2018. Secure Device Bootstrapping Without Secrets Resistant to Signal Manipulation Attacks. In *2018 IEEE Symposium on Security and Privacy (SP)*. 819–835. https://doi.org/10.1109/SP.2018.00055

[36] Nirnimesh Ghose, Loukas Lazos, and Ming Li. 2022. In-Band Secret-Free Pairing for COTS Wireless Devices. *IEEE Transactions on Mobile Computing* 21, 2 (Feb. 2022), 612–628. https://doi.org/10.1109/TMC.2020.3015010

[37] Dan Goodin. 2018. *Police Decrypt 258,000 Messages after Breaking Pricey IronChat Crypto App.* https://arstechnica.com/information-technology/2018/11/police-decrypt-258000-messages-after-breaking-pricey-ironchat-crypto-app/ [Retrieved 2024-01-07].

[38] Michael T. Goodrich, Michael Sirivianos, John Solis, Claudio Soriente, Gene Tsudik, and Ersin Uzun. 2009/february. Using Audio in Secure Device Pairing. *Int. J. Secur. Netw.* 4, 1/2 (2009/february), 57–68–57–68. https://doi.org/10.1504/IJSN.2009.023426

[39] Michael T. Goodrich, Michael Sirivianos, John Solis, Gene Tsudik, and Ersin Uzun. 2006. Loud and Clear: Human-Verifiable Authentication Based on Audio. In *26th IEEE International Conference on Distributed Computing Systems (ICDCS*. IEEE. https://doi.org/10.1109/icdcs.2006.52

[40] Jens Emil Grønbæk, Mille Skovhus Knudsen, Kenton O'Hara, Peter Gall Krogh, Jo Vermeulen, and Marianne Graves Petersen. 2020. Proxemics Beyond Proximity: Designing for Flexible Social Interaction Through Cross-Device Interaction. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–14. https://doi.org/10.1145/3313831.3376379

[41] Dianqi Han, Ang Li, Jiawei Li, Yan Zhang, Tao Li, and Yanchao Zhang. 2021. DroneKey: A Drone-Aided Group-Key Generation Scheme for Large-Scale IoT Networks. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security (CCS '21)*. Association for Computing Machinery, New York, NY, USA, 1306–1319. https://doi.org/10.1145/3460120.3484789

[42] Mark Handel and James D. Herbsleb. 2002. What Is Chat Doing in the Workplace?. In *Proceedings of the 2002 ACM Conference on Computer Supported Cooperative Work (CSCW '02)*. Association for Computing Machinery, New York, NY, USA, 1–10. https://doi.org/10.1145/587078.587080

[43] Mahboobeh Harandi. 2019. Supporting Occasional Groups in Crowdsourcing Platforms. In *Companion Publication of the 2019 Conference on Computer-Supported Cooperative Work & Social Computing (CSCW '19 Companion)*. Association for Computing Machinery, New York, NY, USA, 52–55. https://doi.org/10.1145/3311957.3361856

[44] Amir Herzberg and Hemi Leibowitz. 2016. Can Johnny Finally Encrypt? Evaluating E2E-encryption in Popular IM Applications. In *Proceedings of the 6th Workshop on Socio-Technical Aspects in Security and Trust (STAST '16)*. Association for Computing Machinery, New York, NY, USA, 17–28. https://doi.org/10.1145/3046055.3046059

[45] Amir Herzberg, Hemi Leibowitz, Kent Seamons, Elham Vaziripour, Justin Wu, and Daniel Zappala. 2021. Secure Messaging Authentication Ceremonies Are Broken. *IEEE Security & Privacy* 19, 2 (March 2021), 29–37. https://doi.org/10.1109/MSEC.2020.3039727

[46] Yantian Hou, Ming Li, and Joshua D. Guttman. 2013. Chorus: Scalable In-band Trust Establishment for Multiple Constrained Devices over the Insecure Wireless Channel. In *Proceedings of the Sixth ACM Conference on Security and Privacy in Wireless and Mobile Networks*. ACM, 167–178–167–178. https://doi.org/10.1145/2462096.2462124

[47] Iulia Ion, Marc Langheinrich, Ponnurangam Kumaraguru, and Srdjan Čapkun. 2010. Influence of User Perception, Security Needs, and Social Factors on Device Pairing Method Choices. In *Proceedings of the Sixth Symposium on Usable Privacy and Security (SOUPS '10)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/1837110.1837118

[48] Tero Jokela, Ming Ki Chong, Andrés Lucero, and Hans Gellersen. 2015. Connecting Devices for Collaborative Interactions. *Interactions* 22, 4 (June 2015), 39–43. https://doi.org/10.1145/2776887

[49] Tero Jokela and Andrés Lucero. 2013. A Comparative Evaluation of Touch-Based Methods to Bind Mobile Devices for Collaborative Interactions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. Association for Computing Machinery, New York, NY, USA, 3355–3364. https://doi.org/10.1145/2470654.2466459

[50] Tero Jokela and Andrés Lucero. 2014. FlexiGroups: Binding Mobile Devices for Collaborative Interactions in Medium-Sized Groups with Device Touch. In *Proceedings of the 16th International Conference on Human-computer Interaction with Mobile Devices & Services (MobileHCI '14)*. Association for Computing Machinery, New York, NY, USA, 369–378. https://doi.org/10.1145/2628363.2628376

[51] Tero Jokela, Parisa Pour Rezaei, and Kaisa Väänänen. 2016. Natural Group Binding and Cross-Display Object Movement Methods for Wearable Devices. In *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '16)*. Association for Computing Machinery, New York, NY, USA, 206–216. https://doi.org/10.1145/2935334.2935346

[52] Daniel Jones. 2022. How Near-Ultrasonic Audio Adds Spatial Awareness to the Sonos System. https://tech-blog.sonos.com/posts/how-near-ultrasonic-audio-adds-spatial-awareness-to-the-sonos-system/ [Retrieved 2024-01-07].

[53] Ronald Kainda, Ivan Flechais, and A. W. Roscoe. 2009. Usability and Security of Out-of-Band Channels in Secure Device Pairing Protocols. In *Proceedings of the 5th Symposium on Usable Privacy and Security (SOUPS '09)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/1572532.1572547

[54] Ronald Kainda, Ivan Flechais, and A. W. Roscoe. 2010. Two Heads Are Better than One: Security and Usability of Device Associations in Group Scenarios. In *Proceedings of the Sixth Symposium on Usable Privacy and Security (SOUPS*

'10). Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/1837110.1837117

[55] Maurice G. Kendall. 1938. A New Measure of Rank Correlation. *Biometrika* 30 (1938).

[56] Sye Loong Keoh, Emil Lupu, and Morris Sloman. 2009. Securing Body Sensor Networks: Sensor Association and Key Management. In *2009 IEEE International Conference on Pervasive Computing and Communications*. 1–6. https://doi.org/10.1109/PERCOM.2009.4912756

[57] Tim Kindberg and Armando Fox. 2002. System Software for Ubiquitous Computing. *IEEE Pervasive Computing* 1, 1 (Jan. 2002), 70–81. https://doi.org/10.1109/MPRV.2002.993146

[58] Darko Kirovski, Michael Sinclair, and David Wilson. 2007. The Martini Synch: Device Pairing via Joint Quantization. In *2007 IEEE International Symposium on Information Theory*. 466–470. https://doi.org/10.1109/ISIT.2007.4557269

[59] Alfred Kobsa, Rahim Sonawalla, Gene Tsudik, Ersin Uzun, and Yang Wang. 2009. Serial Hook-Ups: A Comparative Usability Study of Secure Device Pairing Methods. In *Proceedings of the 5th Symposium on Usable Privacy and Security (SOUPS '09)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/1572532.1572546

[60] Tonko Kovačević, Toni Perković, and Mario Čagalj. 2016. Flashing Displays: User-Friendly Solution for Bootstrapping Secure Associations between Multiple Constrained Wireless Devices. *Security and Communication Networks* 9, 10 (2016), 1050–1071. https://doi.org/10.1002/sec.1400

[61] Christian Kray, Daniel Nesbitt, John Dawson, and Michael Rohs. 2010. User-Defined Gestures for Connecting Mobile Phones, Public Displays, and Tabletops. In *Proceedings of the 12th International Conference on Human Computer Interaction with Mobile Devices and Services (MobileHCI '10)*. Association for Computing Machinery, New York, NY, USA, 239–248. https://doi.org/10.1145/1851600.1851640

[62] Arun Kumar, Nitesh Saxena, Gene Tsudik, and Ersin Uzun. 2009. A Comparative Study of Secure Device Pairing Methods. *Pervasive and Mobile Computing* 5, 6 (Dec. 2009), 734–749. https://doi.org/10.1016/j.pmcj.2009.07.008

[63] Cynthia Kuo, Mark Luk, Rohit Negi, and Adrian Perrig. 2007. Message-in-a-Bottle: User-Friendly and Secure Key Deployment for Sensor Nodes. In *Proceedings of the 5th International Conference on Embedded Networked Sensor Systems (SenSys '07)*. Association for Computing Machinery, New York, NY, USA, 233–246. https://doi.org/10.1145/1322263.1322286

[64] Cynthia Kuo, Ahren Studer, and Adrian Perrig. 2008. Mind Your Manners: Socially Appropriate Wireless Key Establishment for Groups. In *Proceedings of the First ACM Conference on Wireless Network Security (WiSec '08)*. Association for Computing Machinery, New York, NY, USA, 125–130. https://doi.org/10.1145/1352533.1352553

[65] Sven Laur and Sylvain Pasini. 2008. SAS-Based Group Authentication and Key Agreement Protocols. In *Public Key Cryptography – PKC 2008 (Lecture Notes in Computer Science)*, Ronald Cramer (Ed.). Springer, Berlin, Heidelberg, 197–213. https://doi.org/10.1007/978-3-540-78440-1_12

[66] Yee Wei Law, Giorgi Moniava, Zheng Gong, Pieter Hartel, and Marimuthu Palaniswami. 2011. KALwEN: A New Practical and Interoperable Key Management Scheme for Body Sensor Networks. *Security and Communication Networks* 4, 11 (Nov. 2011), 1309–1329. https://doi.org/10.1002/sec.256

[67] Hyewon Lee, Tae H. Kim, Jun W. Choi, and Sunghyun Choi. 2015/april. Chirp Signal-Based Aerial Acoustic Communication for Smart Devices. In *INFOCOM*. 2407–2415. https://doi.org/10.1109/INFOCOM.2015.7218629

[68] Ada Lerner, Eric Zeng, and Franziska Roesner. 2017. Confidante: Usable Encrypted Email: A Case Study with Lawyers and Journalists. In *2017 IEEE European Symposium on Security and Privacy (EuroS&P)*. 385–400. https://doi.org/10.1109/EuroSP.2017.41

[69] Ian Levy and Crispin Robinson. 2018. *Principles for a More Informed Exceptional Access Debate.* https://www.lawfaremedia.org/article/principles-more-informed-exceptional-access-debate [Retrieved 2024-01-07].

[70] Ming Li, Shucheng Yu, Wenjing Lou, and Kui Ren. 2010. Group Device Pairing Based Secure Sensor Association and Key Management for Body Area Networks. In *2010 Proceedings IEEE INFOCOM*. 1–9. https://doi.org/10.1109/INFCOM.2010.5462095

[71] Xiaopeng Li, Qiang Zeng, Lannan Luo, and Tongbo Luo. 2020. T2Pair: Secure and Usable Pairing for Heterogeneous IoT Devices. In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security (CCS '20)*. Association for Computing Machinery, New York, NY, USA, 309–323. https://doi.org/10.1145/3372297.3417286

[72] Yue-Hsun Lin, Ahren Studer, Yao-Hsin Chen, Hsu-Chun Hsiao, Li-Hsiang Kuo, Jonathan M. McCune, King-Hang Wang, Maxwell Krohn, Adrian Perrig, Bo-Yin Yang, Hung-Min Sun, Phen-Lan Lin, and Jason Lee. 2010. SPATE: Small-Group PKI-Less Authenticated Trust Establishment. *IEEE Transactions on Mobile Computing* 9, 12 (Dec. 2010), 1666–1681. https://doi.org/10.1109/TMC.2010.150

[73] Yue-Hsun Lin, Ahren Studer, Hsu-Chin Hsiao, Jonathan M. McCune, King-Hang Wang, Maxwell Krohn, Phen-Lan Lin, Adrian Perrig, Hung-Min Sun, and Bo-Yin Yang. 2009. SPATE: Small-Group PKI-less Authenticated Trust Establishment. In *Proceedings of the 7th International Conference on Mobile Systems, Applications, and Services (MobiSys '09)*. Association for Computing Machinery, New York, NY, USA, 1–14. https://doi.org/10.1145/1555816.1555818

[74] Cristina V. Lopes and Pedro M. Q. Aguiar. 2003/july. Acoustic Modems for Ubiquitous Computing. *IEEE Pervasive Computing* 2, 3 (2003/july), 62–71. https://doi.org/10.1109/MPRV.2003.1228528

[75] Sus Lundgren, Joel E. Fischer, Stuart Reeves, and Olof Torgersson. 2015. Designing Mobile Experiences for Collocated Interaction. In *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing (CSCW '15)*. Association for Computing Machinery, New York, NY, USA, 496–507. https://doi.org/10.1145/2675133.2675171

[76] Anil Madhavapeddy, David Scott, and Richard Sharp. 2003. Context-Aware Computing with Sound. In *UbiComp 2003: Ubiquitous Computing (Lecture Notes in Computer Science)*, Anind K. Dey, Albrecht Schmidt, and Joseph F. McCarthy (Eds.). Springer, Berlin, Heidelberg, 315–332. https://doi.org/10.1007/978-3-540-39653-6_25

[77] Rene Mayrhofer and Hans Gellersen. 2009. Shake Well Before Use: Intuitive and Secure Pairing of Mobile Devices. *IEEE Transactions on Mobile Computing* 8, 6 (June 2009), 792–806. https://doi.org/10.1109/TMC.2009.51

[78] Adib Mehrabi, Antonella Mazzoni, Daniel Jones, and Anthony Steed. 2019. Evaluating the User Experience of Acoustic Data Transmission. *Personal and Ubiquitous Computing* (Dec. 2019), 1–14–1–14. https://doi.org/10.1007/s00779-019-01345-7

[79] Sharan B. Merriam and Elizabeth J. Tisdell. 2015. *Qualitative Research: A Guide to Design and Implementation.* John Wiley & Sons.

[80] Meta. 2009. *WhatsApp.* https://whatsapp.com [Software].

[81] Ghita Mezzour, Ahren Studer, Michael Farb, Jason Lee, Jonathan McCune, Hsu-Chun Hsiao, and Adrian Perrig. 2010. *Ho-Po Key: Leveraging Physical Constraints on Human Motion to Authentically Exchange Information in a Group.* Technical Report CMU-CyLab-11-004. Carnegie Mellon University.

[82] Moses Namara and Bart P. Knijnenburg. 2021. The Differential Effect of Privacy-Related Trust on Groupware Application Adoption and Use during the COVID-19 Pandemic. *Proceedings of the ACM on Human-Computer Interaction* 5, CSCW2 (Oct. 2021), 405:1–405:34. https://doi.org/10.1145/3479549

[83] Rajalakshmi Nandakumar, Krishna K. Chintalapudi, Venkat Padmanabhan, and Ramarathnam Venkatesan. 2013. Dhwani: Secure Peer-to-Peer Acoustic NFC. *ACM SIGCOMM Computer Communication Review* 43, 4 (Aug. 2013), 63–74–63–74. https://doi.org/10.1145/2534169.2486037

[84] Long H. Nguyen and Andrew W. Roscoe. 2008. Authenticating Ad Hoc Networks by Comparison of Short Digests. *Information and Computation* 206, 2 (Feb. 2008), 250–271. https://doi.org/10.1016/j.ic.2007.07.010

[85] Rishab Nithyanand, Nitesh Saxena, Gene Tsudik, and Ersin Uzun. 2010. Groupthink: Usability of Secure Group Association for Wireless Devices. In *Proceedings of the 12th ACM International Conference on Ubiquitous Computing (UbiComp '10)*. Association for Computing Machinery, New York, NY, USA, 331–340. https://doi.org/10.1145/1864349.1864399

[86] Sean Oesch, Ruba Abu-Salma, Oumar Diallo, Juliane Krämer, James Simmons, Justin Wu, and Scott Ruoti. 2022. User Perceptions of Security and Privacy for Group Chat. *Digital Threats: Research and Practice* 3, 2 (Feb. 2022), 15:1–15:29. https://doi.org/10.1145/3491265

[87] Karl Pearson. 1900. X. On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* 50 (1900).

[88] Toni Perković, Mario Čagalj, Toni Mastelić, Nitesh Saxena, and Dinko Begušić. 2012. Secure Initialization of Multiple Constrained Wireless Devices for an Unaided User. *IEEE Transactions on Mobile Computing* 11, 2 (2012), 337–351. https://doi.org/10.1109/TMC.2011.35

[89] Florentin Putz, Flor Álvarez, and Jiska Classen. 2020. Acoustic Integrity Codes: Secure Device Pairing Using Short-Range Acoustic Communication. In *Proceedings of the 13th ACM Conference on Security and Privacy in Wireless and Mobile Networks*. 31–41. https://doi.org/10.1145/3395351.3399420

[90] Florentin Putz, Steffen Haesler, and Matthias Hollick. 2024. *Lab Study Dataset: Fast and Secure Contact Exchange in Groups.* https://doi.org/10.5281/zenodo.13324112 [Dataset].

[91] Florentin Putz, Steffen Haesler, Thomas Völkl, Maximilian Gehring, Nils Rollshausen, and Matthias Hollick. 2024. PairSonic: Helping Groups Securely Exchange Contact Information. In *Companion Publication of the 2024 Conference on Computer-Supported Cooperative Work & Social Computing (CSCW '24 Companion)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3678884.3681818

[92] R Core Team. 2023. *R: A Language and Environment for Statistical Computing.* R Foundation for Statistical Computing.

[93] David A. Robb, Thomas S. Methven, Stefano Padilla, and Mike J. Chantler. 2016. Well-Connected: Promoting Collaboration by Effective Networking. In *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing Companion (CSCW '16 Companion)*. Association for Computing Machinery, New York, NY, USA, 90–93. https://doi.org/10.1145/2818052.2874333

[94] Robert Rosenthal. 1991. *Meta-Analytic Procedures for Social Research.* Sage Publications, Inc.

[95] Jeffrey Rubin and Dana Chrisnell. 2008. *Handbook of Usability Testing* (second edition ed.). Wiley Publishing, Inc.

[96] Scott Ruoti, Jeff Andersen, Scott Heidbrink, Mark O'Neill, Elham Vaziripour, Justin Wu, Daniel Zappala, and Kent Seamons. 2016. "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 '16)*. Association for Computing Machinery, New York, NY, USA, 4298–4308. https://doi.org/10.1145/2858036.2858400

[97] Scott Ruoti, Nathan Kim, Ben Burgon, Timothy van der Horst, and Kent Seamons. 2013. Confused Johnny: When Automatic Encryption Leads to Confusion and Mistakes. In *Proceedings of the Ninth Symposium on Usable Privacy and Security (SOUPS '13)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/2501604.2501609

[98] G. Enrico Santagati and Tommaso Melodia. 2015. U-Wear: Software-Defined Ultrasonic Networking for Wearable Devices. In *Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services*. ACM, 241–256–241–256. https://doi.org/10.1145/2742647.2742655

[99] Nitesh Saxena and Md. Borhan Uddin. 2009. Blink 'Em All: Scalable, User-Friendly and Secure Initialization of Wireless Sensor Nodes. In *Cryptology and Network Security (Lecture Notes in Computer Science)*, Juan A. Garay, Atsuko Miyaji, and Akira Otsuka (Eds.). Springer, Berlin, Heidelberg, 154–173. https://doi.org/10.1007/978-3-642-10433-6_11

[100] Svenja Schröder, Markus Huber, David Wind, and Christoph Rottermanner. 2016. When SIGNAL Hits the Fan: On the Usability and Security of State-of-the-Art Secure Mobile Messaging. In *EuroUSEC*.

[101] Samuel S. Shapiro and Martin B. Wilk. 1965. An analysis of variance test for normality (complete samples). *Biometrika* 52 (1965).

[102] Maliheh Shirvanian and Nitesh Saxena. 2015. On the Security and Usability of Crypto Phones. In *Proceedings of the 31st Annual Computer Security Applications Conference (ACSAC '15)*. Association for Computing Machinery, New York, NY, USA, 21–30. https://doi.org/10.1145/2818000.2818007

[103] Maliheh Shirvanian, Nitesh Saxena, and Jesvin James George. 2017. On the Pitfalls of End-to-End Encrypted Communications: A Study of Remote Key-Fingerprint Verification. In *Proceedings of the 33rd Annual Computer Security Applications Conference (ACSAC '17)*. Association for Computing Machinery, New York, NY, USA, 499–511. https://doi.org/10.1145/3134600.3134610

[104] Erica Shusas, Patrick Skeba, Eric P. S. Baumer, and Andrea Forte. 2023. Accounting for Privacy Pluralism: Lessons and Strategies from Community-Based Privacy Groups. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (CHI '23)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3544548.3581331

[105] Sidney Siegel. 1956. *Nonparametric statistics for the behavioral sciences.* McGraw-Hill, New York.

[106] Signal Foundation. 2014. *Signal.* https://signal.org [Software].

[107] Claudio Soriente, Gene Tsudik, and Ersin Uzun. 2008. HAPADEP: Human-Assisted Pure Audio Device Pairing. In *Information Security*, Tzong-Chen Wu, Chin-Laung Lei, Vincent Rijmen, and Der-Tsai Lee (Eds.). Springer Berlin Heidelberg, 385–400–385–400. https://doi.org/10.1007/978-3-540-85886-7_27

[108] Frank Stajano and Ross Anderson. 2000. The Resurrecting Duckling: Security Issues for Ad-hoc Wireless Networks. In *Security Protocols (Lecture Notes in Computer Science)*, Bruce Christianson, Bruno Crispo, James A. Malcolm, and Michael Roe (Eds.). Springer, Berlin, Heidelberg, 172–182. https://doi.org/10.1007/10720107_24

[109] Ahren Studer, Christina Johns, Jaanus Kase, Kyle O'Meara, and Lorrie Cranor. 2008. A Survey to Guide Group Key Protocol Development. In *2008 Annual Computer Security Applications Conference (ACSAC)*. 475–484. https://doi.org/10.1109/ACSAC.2008.28

[110] Milan Stute, David Kreitschmann, and Matthias Hollick. 2018. One Billion Apples' Secret Sauce: Recipe for the Apple Wireless Direct Link Ad Hoc Protocol. In *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking* (New Delhi, India) *(MobiCom '18)*. Association for Computing Machinery, New York, NY, USA, 529–543. https://doi.org/10.1145/3241539.3241566

[111] Joshua Tan, Lujo Bauer, Joseph Bonneau, Lorrie Faith Cranor, Jeremy Thomas, and Blase Ur. 2017. Can Unicorns Help Users Compare Crypto Key Fingerprints?. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. Association for Computing Machinery, New York, NY, USA, 3787–3798. https://doi.org/10.1145/3025453.3025733

[112] Telegram FZ LLC. 2013. *Telegram.* https://telegram.org/ [Software].

[113] Threema GmbH. 2012. *Threema.* https://threema.ch [Software].

[114] Peter Tolmie, Steve Benford, Chris Greenhalgh, Tom Rodden, and Stuart Reeves. 2014. Supporting Group Interactions in Museum Visiting. In *Proceedings of the 17th ACM Conference on Computer-Supported Cooperative Work & Social Computing (CSCW '14)*. Association for Computing Machinery, New York, NY, USA, 1049–1059. https://doi.org/10.1145/2531602.2531619

[115] Alexander Ulrich, Ralph Holz, Peter Hauck, and Georg Carle. 2011. Investigating the OpenPGP Web of Trust. In *Computer Security – ESORICS 2011 (Lecture Notes in Computer Science)*, Vijay Atluri and Claudia Diaz (Eds.). Springer, Berlin, Heidelberg, 489–507. https://doi.org/10.1007/978-3-642-23822-2_27

[116] Gustavo Umbelino, Vivian Ta, Samuel Blake, Eric Truong, Amy Luo, and Steven Dow. 2019. ProtoTeams: Supporting Small Group Interactions in Co-Located Crowds. In *Companion Publication of the 2019 Conference on Computer-Supported Cooperative Work & Social Computing (CSCW '19 Companion)*. Association for Computing Machinery, New York, NY, USA, 392–397. https://doi.org/10.1145/3311957.3359505

[117] Nik Unger, Sergej Dechand, Joseph Bonneau, Sascha Fahl, Henning Perl, Ian Goldberg, and Matthew Smith. 2015. SoK: Secure Messaging. In *2015 IEEE Symposium on Security and Privacy*. 232–249. https://doi.org/10.1109/SP.2015.22

[118] Ersin Uzun, Kristiina Karvonen, and N. Asokan. 2007. Usability Analysis of Secure Pairing Methods. In *Financial Cryptography and Data Security (Lecture Notes in Computer Science)*, Sven Dietrich and Rachna Dhamija (Eds.). Springer, Berlin, Heidelberg, 307–324. https://doi.org/10.1007/978-3-540-77366-5_29

[119] Ersin Uzun, Nitesh Saxena, and Arun Kumar. 2011. Pairing Devices for Social Interactions: A Comparative Usability Evaluation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11)*. Association for Computing Machinery, New York, NY, USA, 2315–2324. https://doi.org/10.1145/1978942.1979282

[120] Jukka Valkonen, N. Asokan, and Kaisa Nyberg. 2006. Ad Hoc Security Associations for Groups. In *Security and Privacy in Ad-Hoc and Sensor Networks (Lecture Notes in Computer Science)*, Levente Buttyán, Virgil D. Gligor, and Dirk Westhoff (Eds.). Springer, Berlin, Heidelberg, 150–164. https://doi.org/10.1007/11964254_14

[121] Diana A. Vasile, Martin Kleppmann, Daniel R. Thomas, and Alastair R. Beresford. 2020. Ghost Trace on the Wire? Using Key Evidence for Informed Decisions. In *Security Protocols XXVII (Lecture Notes in Computer Science)*, Jonathan Anderson, Frank Stajano, Bruce Christianson, and Vashek Matyáš (Eds.). Springer International Publishing, Cham, 245–257. https://doi.org/10.1007/978-3-030-57043-9_23

[122] Elham Vaziripour, Justin Wu, Mark O'Neill, Ray Clinton, Jordan Whitehead, Scott Heidbrink, Kent Seamons, and Daniel Zappala. 2017. Is That You, Alice? A Usability Study of the Authentication Ceremony of Secure Messaging Applications. In *Proceedings of the Thirteenth USENIX Conference on Usable Privacy and Security (SOUPS '17)*. USENIX Association, USA, 29–47.

[123] Elham Vaziripour, Justin Wu, Mark O'Neill, Daniel Metro, Josh Cockrell, Timothy Moffett, Jordan Whitehead, Nick Bonner, Kent Seamons, and Daniel Zappala. 2018. Action Needed! Helping Users Find and Complete the Authentication Ceremony in Signal. In *Proceedings of the Fourteenth USENIX Conference on Usable Privacy and Security (SOUPS '18)*. USENIX Association, USA, 47–62.

[124] VERBI Software. 2021. *MAXQDA 2022*. VERBI Software, Berlin, Germany. https://www.maxqda.com [Software].

[125] Quian Wang, Kui Ren, Man Zhou, Tao Lei, Dimitrios Koutsonikolas, and Lu Su. 2016. Messages behind the Sound. In *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking - MobiCom*. ACM Press. https://doi.org/10.1145/2973750.2973765

[126] Honghao Wei, Cheng-Kang Hsieh, Longqi Yang, and Deborah Estrin. 2016. GroupLink: Group Event Recommendations Using Personal Digital Traces. In *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing Companion (CSCW '16 Companion)*. Association for Computing Machinery, New York, NY, USA, 110–113. https://doi.org/10.1145/2818052.2874338

[127] Rand Wilcox. 2017. *Introduction to Robust Estimation and Hypothesis Testing*. Elsevier, Academic Press.

[128] Frank Wilcoxon. 1945. Individual Comparisons by Ranking Methods. *Biometrics Bulletin* 1, 6 (1945), 80–83. https://doi.org/10.2307/3001968

[129] Justin Wu, Cyrus Gatrell, Devon Howard, Jake Tyler, Elham Vaziripour, Kent Seamons, and Daniel Zappala. 2019. Something Isn't Secure, but i'm Not Sure How That Translates into a Problem: Promoting Autonomy by Designing for Understanding in Signal. In *Proceedings of the Fifteenth USENIX Conference on Usable Privacy and Security (SOUPS'19)*. USENIX Association, USA, 137–153.

[130] Weitao Xu, Junqing Zhang, Shunqi Huang, Chengwen Luo, and Wei Li. 2021. Key Generation for Internet of Things: A Contemporary Survey. *Comput. Surveys* 54, 1 (Jan. 2021), 14:1–14:37. https://doi.org/10.1145/3429740

[131] Jiahuan Zheng, Xin Peng, Jiacheng Yang, Huaqian Cai, Gang Huang, Ying Zhang, and Wenyun Zhao. 2017. CollaDroid: Automatic Augmentation of Android Application with Lightweight Interactive Collaboration. In *Proceedings of the 2017 ACM Conference on Computer-Supported Cooperative Work & Social Computing (CSCW '17)*. Association for Computing Machinery, New York, NY, USA, 2462–2474. https://doi.org/10.1145/2998181.2998278

A Supplementary Figures

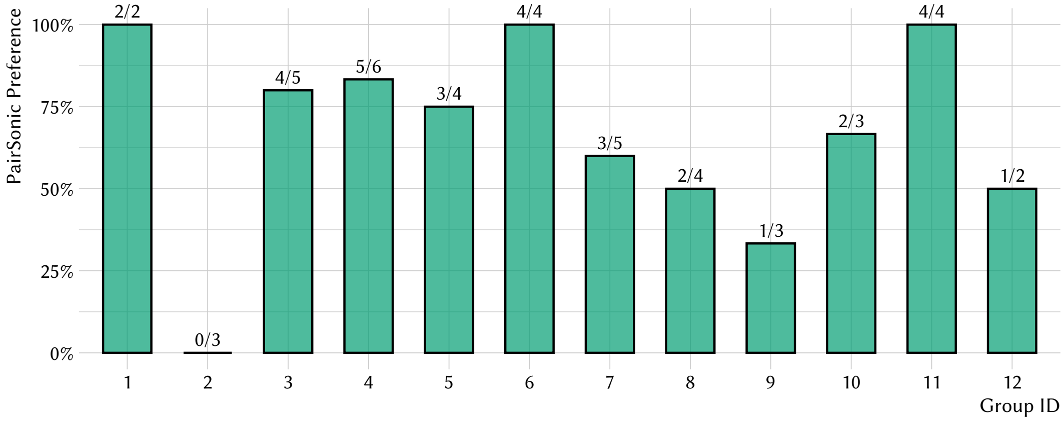
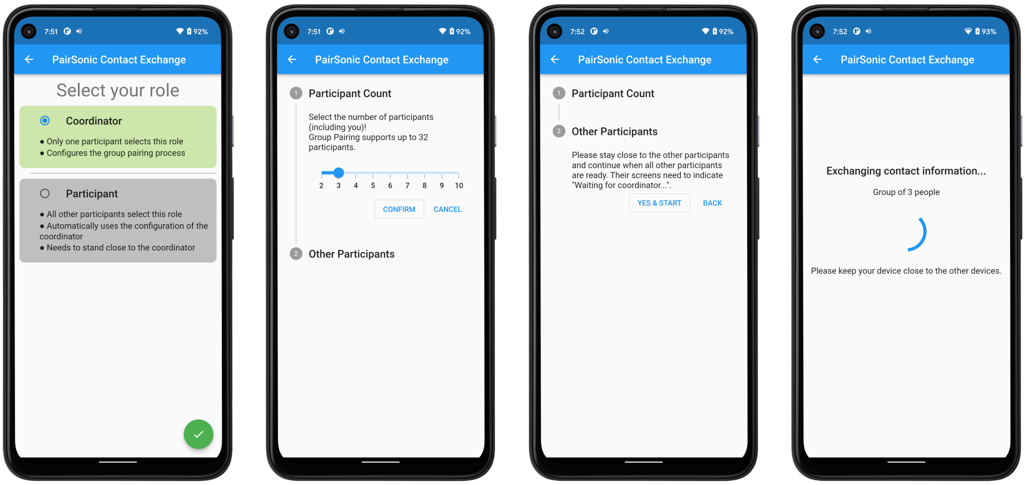


Fig. 9. This bar chart shows the proportion of group members preferring PairSonic in each study group. The label on top of each bar shows the corresponding number of group members who preferred PairSonic and the total group size. Only four groups had a unanimous preference (three favoring PairSonic, one favoring SafeSlinger). This variation suggests the possible anchor effect discussed in Section 10.2 might not be very strong.



- Only one participant selects this role
- Configures the group pairing process

- All other participants select this role
- Automatically uses the configuration of the coordinator
- Needs to stand close to the coordinator

(a) Role selection.

Select the number of participants (including you)!
Group Pairing supports up to 32 participants.

(b) Participant count.

Please stay close to the other participants and continue when all other participants are ready. Their screens need to indicate "Waiting for coordinator...".

(c) Ready check.

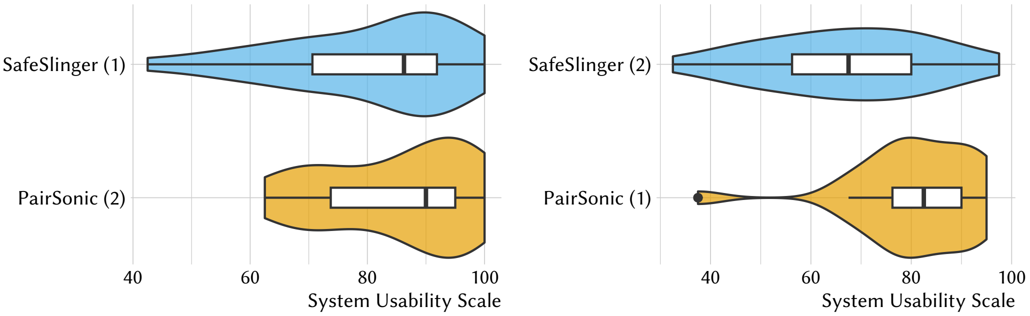
Exchanging contact information...

Group of 3 people

Please keep your device close to the other devices.

(d) Exchange begins.

Fig. 10. This figure depicts PairSonic's initialization phase from the perspective of the *coordinator* role. (a) The coordinator begins by selecting their role. (b) They proceed to select the total number of participants, counting themselves. (c) Before continuing, they ensure that the remaining participants are in the ready state (Figure 2b). (d) The coordinator's smartphone then initiates the exchange via the acoustic OOB channel. The verification and finalization phases that follow are identical for both the *coordinator* and *participant* roles, as shown in Figure 2c and Figure 2d.



(a) Participants who first encountered SafeSlinger.

(b) Participants who first encountered PairSonic.

Fig. 11. Comparison of our participants' SUS scores for SafeSlinger and PairSonic, depending on the order they encountered both systems. The violin plots show a density estimation of the distributions. The boxplots show quartiles, median, and outliers.

Table 4. Kendall's correlation between control and dependent variables.

		2	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**ATI**
1	Smartphone Familiarity	-.15	-.1	.19	.11	.15	.16	-.09	.2	-.14
2	Order (PairSonic First)		.11	**-.33****	-.16	-.15	-.11	.11	-.24	-.18
3	Preference (PairSonic)			**-.33***	.19	-.12	0	.14	.12	.18
4	SUS SafeSlinger				.17	**.29***	**.24***	**-.34****	0	.07
5	SUS PairSonic					**.31****	**.28***	-.12	-.06	.16
6	Security SafeSlinger						**.3***	0	-.05	.09
7	Security PairSonic							-.15	.04	.14
8	Time SafeSlinger								.09	.02
9	Time PairSonic									.14

Note: $N = 45$ $^{*}p < .05$ $^{**}p < .01$

B Questionnaire

We provide a translation of our questionnaire. We gave the questionnaire to the participants in the native language of the country where we ran the study.

B.1 Experiment A

Please indicate the degree to which you agree/disagree with the following statements.
⟨*Five responses ranging from "Strongly disagree" to "Strongly agree"*⟩

(1) I think that I would like to use this system frequently.
(2) I found the system unnecessarily complex.
(3) I thought the system was easy to use.
(4) I think that I would need the support of a technical person to be able to use this system.
(5) I found the various functions in this system were well integrated.
(6) I thought there was too much inconsistency in this system.
(7) I would imagine that most people would learn to use this system very quickly.
(8) I found the system very cumbersome to use.
(9) I felt very confident using the system.
(10) I needed to learn a lot of things before i could get going with this system.

(11) I think that this system is secure.

B.2 Experiment B

⟨*Same content as experiment A*⟩

B.3 Preference

(1) Which system did you like better? ⟨*Experiment A / Experiment B*⟩

B.4 Groups

Which of the types of digital communication groups listed below do you participate in? What is the average number of participants in these groups? For which of these groups do you want to know exactly who is part of the group or that no unauthorized person has access?
⟨*For each of the following categories: a checkbox "Participation?", a field "Size?", a checkbox "Security?".*⟩

(1) Public chat group (e. g., Discord, Telegram, IRC)
(2) Private chat group (e. g., WhatsApp, Signal)
(3) Professional chat group (e. g., Microsoft Teams)
(4) Online forum
(5) Group in social network (e. g., Facebook group)
(6) Audio/video conference (e. g., TeamSpeak, Zoom)
(7) Mailing list
(8) Collaboration tools (e. g., Google Docs, Etherpad, Miro)
(9) Project planning tools (e. g., Jira, Trello, GitHub)
(10) ⟨*Additional free text fields*⟩

B.4.1 Total Number of Groups.

(1) Please estimate the total number of digital communication groups you participate in. ⟨*Number field*⟩

B.5 Demographic Information

(1) Please state your gender. ⟨*Male / Female / Diverse / Free text / No answer*⟩
(2) Please state your field of activity or field of studies. ⟨*Free text response*⟩
(3) How old are you? ⟨*18–19 / 20–24 / 25–29 / 30–34 / 35–39 / 40–44 / 45–49 / 50–54 / 55–59 / 60–64 / 65–69 / 70 or older / No answer*⟩
(4) Please state your general education. ⟨*Still in school / Lower secondary education / Polytechnic high school / Intermediary secondary education / University entrance qualification / No general school leaving certificate / Free text / No answer*⟩
(5) Please state your professional/vocational education. ⟨ *Vocational training or training in dual system / Technical college diploma / Technical college diploma in the former GDR / Bachelor / Master / Diploma / PhD / No professional or vocational degree / Free text / No answer*⟩
(6) Are you currently a student enrolled in a degree program (Bachelor, Master, Diploma, State examination, Magister)? ⟨*Yes / No / No answer*⟩
(7) Have you been using a smartphone for more than two years? ⟨*Yes / No / No answer*⟩

B.6 Technology

In the following questionnaire, we will ask you about your interaction with technical systems. The term *"technical systems"* refers to apps and other software applications, as well as entire digital

devices (e. g., mobile phone, computer, TV, car navigation). Please indicate the degree to which you agree/disagree with the following statements.
⟨*Six responses from "Completely disagree" to "Completely agree"*⟩

(1) I like to occupy myself in greater detail with technical systems.
(2) I like testing the functions of new technical systems.
(3) I predominantly deal with technical systems because I have to.
(4) When I have a new technical system in front of me I try it out intensively.
(5) I enjoy spending time becoming acquainted with a new technical system.
(6) It is enough for me that a technical system works; I don't care how or why.
(7) I try to understand how a technical system exactly works.
(8) It is enough for me to know the basic functions of a technical system.
(9) I try to make full use of the capabilities of a technical system.

Received July 2023; revised January 2024; accepted March 2024