SEPPI: Secure and Privacy-Preserving Invitation for Unified Communication Meetings

Thomas Reisinger
De Montfort University
Leicester, United Kingdom
thomas.reisinger@my365.dmu.ac.uk

Eerke Albert Boiten
De Montfort University
Leicester, United Kingdom
eerke.boiten@dmu.ac.uk

Isabel Wagner
University of Basel
Basel, Switzerland
isabel.wagner@unibas.ch

***Abstract*—Online meetings through various Unified Communication (UC) platforms are a common theme in the daily lives of many people. While the security features of many UC platforms have been improved over time, e.g., through the addition of end-to-end encryption, the invitation process is commonly still based on calendar invites sent by email. These email invitations form a security and privacy weakness for unified communication systems. To mitigate this weakness, we present in this paper the novel Secure and Privacy-Preserving Invitation (SEPPI) architecture. Based on FIDO2 authentication, SEPPI provides confidentiality and integrity for meeting invites and pseudonymity and unlinkability for meeting participants. We analyze the security of SEPPI and conduct a large-scale systematic user evaluation of a SEPPI prototype. Our findings suggest that SEPPI achieves increased privacy and security at an acceptable convenience cost.**

***Index Terms*—unified communication, security, privacy, invitation, email**

I. Introduction

UC, i.e., video and audio conferencing with Instant Messaging (IM), is ubiquitous nowadays, but the currently available tools provide only limited security and privacy guarantees. Our previous research into UC privacy and security threats [17] and UC requirements for digital activists [18] identified several risks. For example, missing end-to-end encryption, anonymous and undetectable communication, transparency and awareness; weak authentication; and infrastructure deployment limitations.

In this paper, we address the security of meeting invitations as one missing key aspect. UC platforms with scheduling and invitation capabilities, such as Zoom, Google Meet, or Microsoft Teams, use email for these functions. Consequently, they inherit the security and privacy properties the email system provides, which are generally insufficient. Invitation emails contain sensitive meeting information, such as connection URLs, participant names, email addresses, and meeting PINs in clear text. This causes threats to confidentiality, integrity, authenticity, and non-repudiation. Further, it also causes privacy threats, including the linkability of participants, which allows (partial) reconstructions of participants' social graphs by adversaries, including at least internet and email providers.

UC meeting organizers and invitees tend to be unaware of this exposure and that it could put them at risk. As with metadata for all communications, adversaries knowing that a meeting was planned between certain parties may cause harm, even if they never gained access to the meeting itself. Digital activists using UC meetings to coordinate on-the-ground activities could be a target for autocratic governments.

Email could, in principle, be protected through encryption. Existing standards and extensions that provide security and privacy properties for email are, for example, PGP or S/MIME. However, they are complex to integrate, and the adoption is low [11], [19]. Particularly, invited participants outside an organizational email system are prone to information exposure because end-to-end secured communication between multiple email domains is hard to ensure.

Providing a separate secure channel to share meeting authentication PIN and other sensitive details would improve security. For example, users could manually exchange details via an encrypted IM channel or use an end-to-end encrypted web session for scheduled meetings. CryptoParties also provide a good platform for key exchange to the general public. However, there is a tension between convenience and security when going beyond clear text email invitations.

***Motivation and contribution**:* This paper focuses on addressing the issue of unsecured email invitations, as offered commonly by UC platforms. It presents an architecture called SEPPI that offers strong pseudonymous authentication based on Fast IDentity Online 2 (FIDO2), pseudonymous meeting scheduling, adding also granular security and privacy settings for UC meetings. Further, we developed a prototype of SEPPI and conducted an online evaluation to understand users' perception of the architecture, focusing on the usability of the registration and scheduling processes. In particular, this paper makes the following contributions:

1) Design of an architecture for secure and privacy-preserving meeting invitations (SEPPI)
2) Analysis of the security and privacy properties of SEPPI
3) Evaluation of the usability of SEPPI based on an implemented prototype.

II. Related Work and Background

***UC security and privacy**:* The security and privacy of UC meetings has been under increased scrutiny since the Covid-19 pandemic. For example, in the widely publicized zoombombing attack, perpetrators join and deliberately disrupt

979-8-3503-7099-7/24/$31.00 ©2024 IEEE

virtual meetings [13]. This attack is often enabled by legitimate participants sharing meeting information, including passwords and names of invited participants, which makes common protections ineffective. The mitigation proposed in [13] is to create unique joining links for each participant.

Another attack uses face recognition on publicly available screenshots from video conferences to construct a facial image dataset containing personal details about meeting participants, including facial characteristics, age, gender, usernames, and sometimes even full names. Cross-referencing with social networks allows additional inferences, including social links between participants [12]. Recommended mitigations all rely on actions by individual users, for example, avoiding video streaming, using generic pseudonyms and generic backgrounds, and avoiding to post meeting photos online.

To address attacks on UC meetings, UC platforms have implemented a range of protections, including waiting rooms and end-to-end encryption [5]. While these protections mitigate some threats, others remain. For example, an analysis of end-to-end encrypted UC platforms found that all platforms are vulnerable to active man-in-the-middle attacks because of missing or optional user authentication [2].

In addition, despite the prevalence of platforms using insecure email channels to distribute meeting information, to the best of our knowledge there are currently no proposals for alternative meeting invitation processes that provide security and privacy properties. In this paper, we propose such a process, relying on FIDO2 as a building block.

FIDO2: FIDO2 is a set of open standards developed by the FIDO Alliance to enable secure and passwordless multi-factor authentication (MFA). The digital credential used in this authentication method is often called *passkey*. Web Authentication (WebAuthn) is a component of FIDO2 that specifically focuses on providing a standard web API for implementing strong, public key-based authentication.

The WebAuthn Application Programming Interface (API) allows users to authenticate to websites using a FIDO2 authenticator. The authenticator can be internal, such as biometrics or PIN authentication provided by the operating system, or external, such as a security key, mobile device, or wearable. The FIDO2 authenticator generates a public/private key pair, where the private key is stored on the FIDO2 authenticator, and the public key is shared with the website. The FIDO2 authenticator uses the Client-to-Authenticator Protocol (CTAP) to communicate with the client, e.g., a browser, to complete the authentication process. During authentication, the website uses the public key to verify the signature generated with the private key, which confirms the user's identity [9]. This process avoids the need for per-website passwords.

Two formal security analyses of FIDO2 confirmed the security of some aspects of the FIDO2 protocols, including authentication security of WebAuthn [4] and confidentiality, authentication, and privacy properties under specific assumptions [10]. However, the analyses also showed design flaws in the CTAP2 protocol that lead to possible attacks, including authenticator rebinding and man-in-the-middle attacks [10]. Replacing the unauthenticated Diffie-Hellman key exchange in CTAP2 with password-authenticated key exchange (PAKE) improves the design and results in stronger security [4].

Leading browser vendors, including Google, Apple, Microsoft, and Mozilla, support FIDO2. In addition, they aim to act as passkey providers and offer syncing of passkeys in real-time across a user's devices. For example, Apple offers passkey synchronization via its iCloud Keychain. This convenience feature allows users to recover from the loss of a device, and means that users do not have to register every device for each website.

III. SEPPI Architecture

In this section, we describe SEPPI, a new architecture that provides secure and privacy-preserving meeting invites based on FIDO2. We first outline the traditional invitation process and our threat model, then describe the architecture and analyze its security.

Email invitation: The scheduling and invitation process provides meeting participants with self-service meeting coordination and invitation. The meeting *organizer* specifies the meeting subject, description, and date/time, and selects the participants. The *UC platform* sends this information, together with dial-in instructions, a PIN/passcode, and a web link to the meeting, as a meeting invite in iCalendar format to all participants via email, most commonly in clear text. *Participants* receive the iCalendar invitation and, upon accepting the meeting, add the information to their *calendar system* which can be local or cloud-based, for example, associated with Google Gmail or Microsoft Outlook accounts. When it is time to join a specific meeting, participants use the instructions and link stored in their calendar system to access the meeting.

Threat model: Meeting invites essentially consist of *metadata* which reveal potentially sensitive information: the subject and description can reveal what the meeting is about, the list of participants reveals social connections between participants, and the joining instructions can be used to access meeting content.

In the traditional email invitation process, this sensitive information is stored in the UC system's database, in each participant's calendar system and local computer, and is transmitted over the public internet as unencrypted email.

In this paper, we disregard threats to calendar systems and local machines. Instead, we assume that *external* adversaries are able to observe clear text emails and read/modify their contents, while *internal* adversaries can access the UC system's database. Real-world adversaries with these capabilities include providers of technical systems, including email and internet service providers, the UC system, and governments or intelligence agencies who may be able to enforce access to these technical systems by legal means.

Design goals: The proposed architecture should provide protection primarily against external attackers (we outline how SEPPI can be extended to protect against internal attackers in Section V). In particular, SEPPI should provide confidentiality

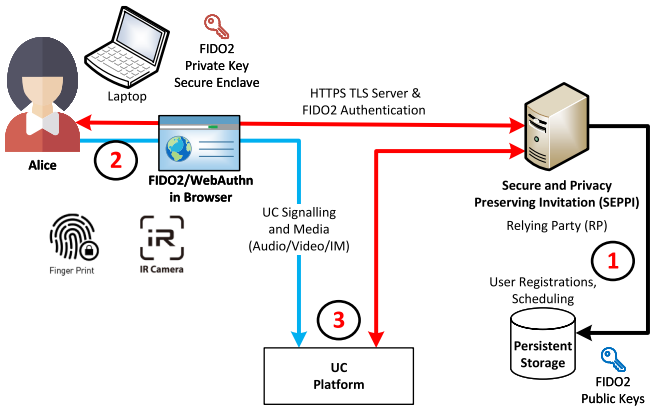


Fig. 1: Secure and Privacy-Preserving Invitation (SEPPI) Architecture.

and integrity for meeting invites as well as pseudonymity and unlinkability for users of SEPPI.

In addition, the architecture should be easy to use and require minimal additional effort from its users. In particular, SEPPI should not require additional software to be installed and should work with typical desktop and smartphone operating systems and web browsers, while being able to notify users of new meeting invites and allowing users to add meetings to their calendar systems.

Overview of SEPPI architecture: Figure 1 shows the main components of SEPPI. Compared with typical email-based invitation systems, SEPPI adds a web platform to handle invitations (component (1) in Figure 1). Instead of using unencrypted emails to share meeting information, SEPPI only uses emails to notify the meeting participants and organizers about the *availability* of new or changed invitations. The notification emails contain a hyperlink that allows users to interact with SEPPI via a secure web session. Once successfully authenticated, users can create, join, update, and delete UC meetings. SEPPI prevents the delivery of meeting information (iCalendar) in clear text via email and instead offers a secure way to import calendar entries directly from SEPPI into systems such as Microsoft Outlook, Apple Calendar, and Google Calendar. Instead of full joining details for meetings, calendar entries only contain a hyperlink to SEPPI, so that only authenticated participants can join meetings.

The sequence diagrams in Figure 2 compare the steps to schedule and join meetings in traditional systems (Figure 2a) and SEPPI (Figure 2b), where different or additional steps in SEPPI are highlighted in red.

Registration: Before users can start scheduling and joining meetings, they need to register with SEPPI. For this, SEPPI uses the FIDO2 registration process, which generates the private key material stored in the user's computer/smartphone Trusted Platform Module (TPM). At the same time, the public key of the user's FIDO2 authenticator is stored in the SEPPI database. During the registration, each FIDO2 authenticator of a user receives a unique ID in SEPPI, with which users can identify themselves pseudonymously to other users.

A valid email address is only required if users wish to be notified about meeting invitations and changes. In addition, the user can provide an alias, which will be used as a pseudonymous identifier to invite participants by alias instead of their real name or email address. Users can close their accounts at any time, and the data is deleted immediately. Any meetings that the user has organized are deleted, and the user is removed from all meetings they are a participant in.

Account recovery, needed for example when the user loses their only registered authenticator, can be addressed by registering multiple authenticators. Insecure account recovery processes such as text messages or emails with magic links are not used because they could serve as attack vectors.

Scheduling: All SEPPI-authenticated users can schedule UC meetings with typical settings including meeting start time and duration, participants (via aliases or email addresses), subject, and further descriptive information about the meeting. SEPPI also allows users to configure security/privacy options for the meeting already during scheduling, for example, enabling or disabling the lobby, video stream, chat, meeting recording, and whiteboard functionality.

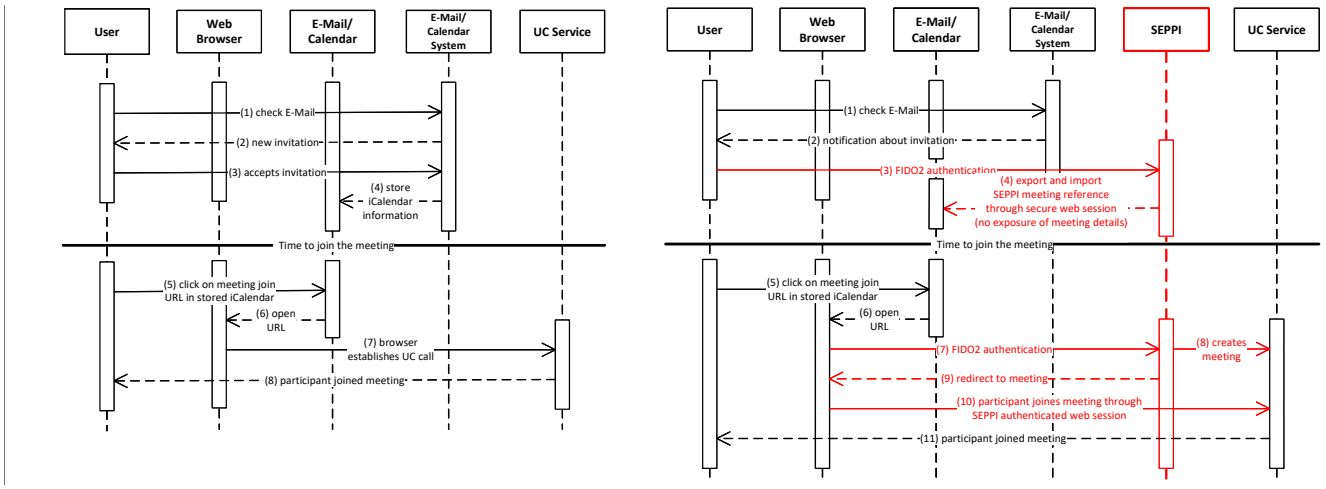
Joining meetings: To join a meeting, users log on to SEPPI, which displays join buttons for all of the user's meetings. SEPPI then directs the user to the UC meeting. For the first user to join a given meeting, SEPPI also sets up the UC meeting on the embedded UC platform. In this way, SEPPI acts as an authentication proxy between the UC platform and the user, which prevents sharing of meeting credentials and ensures authenticity (at the level of the user's alias or email address) of participants.

Data storage: To provide its functionality, SEPPI stores some data in a SQL database, in particular related to users and meetings. However, the amount of data stored is minimized, and data is only revealed to users when necessary.

The user data includes FIDO2 related data items (e.g., per-device authenticator public keys) as well as login names (typically email addresses). Meeting data includes the meeting organizer, participants, timing, subject, description, as well as the privacy and security options configured for the meeting.

Security and privacy analysis: **Confidentiality and integrity.** User authentication with FIDO2 for the SEPPI web session provides authenticity based on the e-mail address or alias. Further, this authentication is resistant against Man-in-the-middle (MITM) and phishing attacks. In addition, the web sessions are end-to-end encrypted with Transport Layer Security (TLS) between the user web browser and the SEPPI platform, ensuring confidentiality and integrity of the sensitive information transferred.

Unlinkability. SEPPI exposes meeting information, including the subject and participant list, only to authenticated participants. External adversaries are therefore unable to link participants to particular meetings. External adversaries including the internet provider for the SEPPI system itself could, through traffic analysis on notification emails, observe which email addresses have been invited to a meeting and



(a) Traditional invitation system

(b) SEPPI invitation system

Fig. 2: Sequence diagrams showing the meeting invitation (top) and meeting join (bottom) processes.

thereby partially link participants to each other and reconstruct social graphs. This can be mitigated by either introducing random delays to notification emails, or by standardizing the time when notifications are sent (e.g., once per hour). Traffic analysis could also be possible when multiple participants connect to SEPPI to join their meeting. This risk reduces the more concurrent meetings are scheduled with SEPPI. Corrupt meeting participants learn about the other participants in their meetings (aliases or provided e-mail addresses), but learn nothing about other meetings.

Pseudonymity. SEPPI does not require real names of users, and valid email addresses only for the notification functionality. This means that users can choose how they are identified on SEPPI: through their real email address, a disposable email address, or any freely chosen pseudonym. The choice of pseudonym can also affect a user's linkability. In particular, users can reduce their own linkability, even against other SEPPI users, by signing up without an email address (or with a disposable address).

Undetectability and unobservability. SEPPI does not provide undetectability of the web session, UC, or e-mail traffic. As a result, adversaries can observe web sessions with the SEPPI platform. If SEPPI is not very widely used, adversaries could then reasonably assume that any user is an activist or other suspicious person. However, users could protect themselves to some extent by using Tor to interact with SEPPI for the scheduling process.

IV. Usability Evaluation

In addition to the security analysis, we conducted a user study to understand whether the additional steps in SEPPI are usable and acceptable. Our hypothesis was that FIDO2 registration and login could cause friction in the UC scheduling process, because FIDO2 is not widely adopted yet and could confuse users used to password-based authentication. The additional steps needed to schedule meetings and configure their security and privacy settings could also be confusing or perceived as an inconvenience.

A. Prototype implementation

To conduct our user study, we implemented a prototype of the SEPPI architecture using TypeScript and Node.js for the web platform, JavaScript for the user interface, and SimpleWebAuthn [16] for the FIDO2 integration. On the user side, our prototype supports Windows 11 (Google Chrome/Firefox) and macOS (Safari) as FIDO2 authenticators. In principle, extension to smartphones, tablets, and Linux desktops, as well as hardware-based external authenticators and roaming authenticators is possible.

The SEPPI prototype provides the registration and authentication of users as well as meeting scheduling as a web service. UC meetings themselves are integrated via API calls to the Jitsi Meet platform [1]. For the user to join a meeting, they need to authenticate with FIDO2 first and join the Jitsi meeting via SEPPI. As a backend UC platform, any open architecture can be used as long it can be integrated via an API. We selected Jitsi because it is an open-source UC, security-focused platform.

A MariaDB service is used for persistence to store user registrations and scheduled meetings. Data is encrypted at the database level, i.e., not at table or field level, and the key material is available to the administrator.

B. Design of the evaluation study

We used two standard quantitative instruments to evaluate usability: the System Usability Scale (SUS) and the Technology Acceptance Model (TAM) [6]–[8]. After giving consent to their participation in the study and passing the screening questions, participants completed a questionnaire about the socio-demographics and their prior UC experience. Then, participants were asked to complete two tasks using our prototype: registration and login (task 1), and scheduling of a

TABLE I: Time in minutes for completing the study.

Age in years	Number of Participants	Time Taken Average	Time Taken StdDev	Registration/Login Average	Registration/Login StdDev	Schedule Meeting Average	Schedule Meeting StdDev
18-24	44	19:31	10:14	02:59	07:35	06:29	03:43
25-34	71	17:20	11:28	01:10	01:40	06:28	05:17
35-44	23	15:27	08:52	00:58	00:37	06:39	06:40
45-54	7	13:55	02:13	01:05	00:49	05:05	00:48
55-64	7	22:19	04:45	00:50	00:20	12:09	06:14
65-74	3	17:15	04:03	00:53	00:17	08:38	03:41

meeting (task 2). We provided a screen recording to explain how to complete both tasks. Finally, participants completed two questionnaires about the perceived usability of SEPPI using the standard SUS and TAM questions. Participants could also provide optional free-text feedback.

Ethics: The study received Research Ethics Approval from De Montfort University in October 2023, with participants providing voluntary and informed consent for data collection. The collected data was anonymized by Prolific and accessible only to the authors. Prolific enforces a minimum payment of £6 per hour and recommends £9 per hour. Participants in our study received an average payment of £8.62 per hour after completing the study.

Pilot Study: We conducted a pilot study with eleven participants to validate the study flow and instructions. The results from the pilot study are not included in the evaluation.

Recruitment: We recruited regular computer users with some UC experience, but not necessarily security or privacy education, through Prolific, and realized the study workflow, i.e., questionnaires and tasks, with Gorilla. Participants were required to use a Windows 11 or macOS desktop computer with Chrome or Safari browser due to restrictions of our prototype. In addition, the participant's minimum age was 18 years, and they were screened for their approval rating (>95%) and previously completed tasks on Prolific (>50). We closed recruitment after n=155 completed submissions, which provides a 95% confidence level with an 8% margin of error for population sizes of one million and above.

C. Evaluation Results

Participants and Their Demographics: Of our 155 participants, 79% had attained a bachelor's degree or higher. Further, most users were below 35 years of age (74%) and employed in some form (72%). This means our participants were significantly younger and better-educated than a representative sample of the population.

Table I lists the time in minutes for completing the study, grouped by participant age. Participants took on average 17 minutes to complete the study, of which two minutes were spent on task 1 (registration and login), seven minutes on task 2 (scheduling of a meeting), and the remaining eight minutes on the questionnaires.

Perception of Security and Privacy: All participants were experienced UC users, with WhatsApp (96%) the most popular UC application, followed by Zoom (95%), Microsoft Teams (91%), and Google Meet (74%). Platforms providing more security and privacy features, such as Signal Messenger (22%), Jitsi (5%), or Wire (1%), were not as widely used. Interestingly, most participants agreed or strongly agreed (78%) that they felt secure and that these platforms protected their privacy. This finding indicates user unawareness of, for example, unsecured email invitations, in line with our earlier research [18].

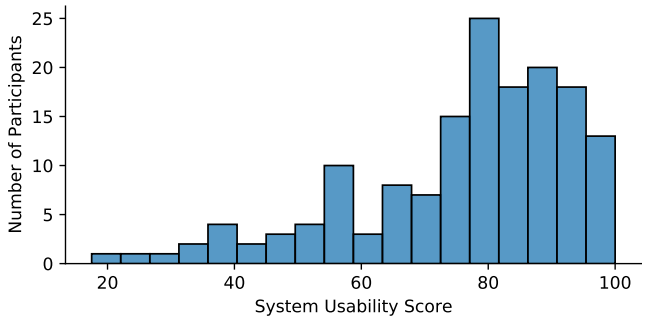


Fig. 3: Distribution of SUS scores for the SEPPI prototype.

Participants confirmed that they perceived SEPPI's security for the registration and login process as high (average 4.12 on a 5-point Likert scale, 85% agreed or strongly agreed), and 94% agreed or strongly agreed the security and privacy settings for the meeting schedule were understandable and valuable (average of 4.3).

Several participants also provided positive comments about the security and privacy features of SEPPI:

> *"Enjoyed the granular control of meeting security and privacy functions - video, chat, lobby, etc"*
>
> *"I liked how the meeting settings were configurable and that it required authentication to enter."*

System Usability Scale (SUS): Participants rated each of the ten SUS questions on a 1–5 Likert scale. Based on these ratings, we computed the SUS score (0–100) for each participant [6]. The calculated Cronbach's α of 0.91 for the questionnaire responses of the ten SUS items confirms an excellent (α>0.9) internal consistency. The average SUS score was 76.4 (median: 80, standard deviation: 17.6), which corresponds to a "Good" rating according to the adjective scale from Bangor et al. [3]. This is an excellent result, given that participants were evaluating a research prototype, not a consumer product. Figure 3 shows the distribution of SUS scores. The participants' free-text comments reflected the high SUS score, for example praising that SEPPI was easy to understand: *"I found it similar to other UC platforms and that makes it very simple to understand."*

Technology Acceptance Model (TAM): To determine to what extent users would accept scheduling meetings via SEPPI, participants rated the standard twelve TAM questions on a 1–7 Likert scale. Six of these questions refer to the perceived usefulness of the technology, phrased in terms of improved job performance and productivity, while the other six questions refer to the perceived ease of use, i.e., the user's perception of how much less effort they have to exert when using the technology [7], [8].

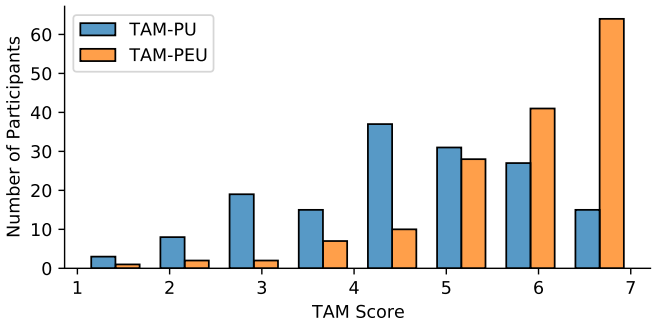


Fig. 4: Distribution of TAM scores for the SEPPI prototype.

TABLE II: Average TAM scores by demographics, from 1=Extremely Disagree to 7=Extremely Agree.

Demographic Categories Age	Average TAM-PU	Average TAM-PEU	Average TAM	n=
18-24 years	4.55	5.84	5.19	44
25-34 years	4.46	5.78	5.12	71
35-44 years	4.99	6.14	5.57	23
45-54 years	4.17	5.38	4.77	7
55-64 years	4.40	5.31	4.86	7
65-74 years	3.78	4.61	4.19	3

The calculated Cronbach's α of 0.95 for the twelve TAM items again indicates an excellent internal consistency. Figure 4 shows the distribution of TAM scores. The average score for perceived usefulness was 4.53 points, while perceived ease of use was rated higher, with an average of 5.79.

Perceived usefulness focuses on aspects of job performance including speed and productivity. Because SEPPI adds steps to the meeting scheduling process, it may take longer to schedule meetings, thereby reducing productivity. This explanation for why TAM-PU scores were lower than TAM-PEU scores was also mentioned by participants: *"I find that Outlook with MS Teams is easier to use, hence why I do not see myself being more productive with SEPPI."*

As Table II shows, the perceived usefulness and perceived ease of use also depended on a participant's age group, where people in the 35–44 age bracket had the highest scores. In particular, their score for perceived usefulness was almost 0.5 points higher than any other age group. A possible explanation for this may be that participants in this group were most exposed to meeting scheduling in their daily lives, and therefore were more convinced of SEPPI's usefulness.

Another participant indicated an obstacle to widespread uptake of a standalone SEPPI solution because of workspace constraints: *"I have to use Teams at work, but would consider SEPPI for other purposes"*

Usability of FIDO2/passkeys: Depending on a user's prior exposure to passkeys, we found that their perceptions of FIDO2 authentication were very different. For example, several participants who were clearly familiar with passkeys gave very positive comments:

> *"I like that it uses passkeys because they are my preferred method of logging in, and I have used passkeys on as many services as possible."*
>
> *"I really like the idea of using FIDO2/PassKeys for this!"*

On the other hand, several other participants were unsure of how passkeys work and therefore felt less secure when they needed to enter their local credentials to authenticate to a website:

> *"I am not sure if having the authentication via my Windows login was good or bad - I think fine as long as that security data is not shared with SEPPI."*
>
> *"I like SEPPI, the only thing that makes me nervous is that it asked for my computer PIN to create my account."*

V. Conclusion, Limitations, and Future Work

In this paper, we have presented SEPPI, an architecture that modifies the invitation process for UC meetings to provide additional privacy and security properties, including confidentiality and integrity for meeting invitations as well as pseudonymity and unlinkability for meeting participants. Although the additional steps introduced in SEPPI may lead to a loss of convenience for users, indicating a tradeoff between security and usability, our usability evaluation showed that participants found SEPPI to be very usable, with convenience remaining high. This is an encouraging result that will support future work on SEPPI.

Limitations: As indicated in Section II, several attacks against FIDO2 are currently known; we have not yet fully evaluated their impact on SEPPI. The same goes for known attacks against TLS or embedded UC platform such as Jitsi [14].

To achieve the right level of authentication, we build on FIDO2. Not all potential users may be familiar or comfortable with this, because FIDO2 and passkeys are not well known. However, the support for it from major companies such as Apple, Google, and Microsoft may reduce this negative impact on the potential adoption rate for SEPPI.

The SEPPI architecture does not allow for logging of activities. Organisations may nevertheless have purposes which require the addition of this. In that case, the consequent increase of attack surface will need explicit mitigation.

To fully embed an UC platform with SEPPI, it needs to be *open*, i.e., it needs to allow API access in order to create meetings and set meeting options. In addition, the security and privacy options offered by SEPPI need to be matched with options offered by the platform. Embedding a closed UC platform instead would mean creating and joining meetings could no longer be done within SEPPI, hence limiting added privacy and security only to the meeting invitation process.

Malicious meeting organisers could perform brute force attacks against client aliases or email addresses. This way they could establish if certain individuals have an account on SEPPI, which could inform targeted surveillance efforts.

In its current form, SEPPI presents a single point of failure. Removing its availability disables meeting management as well as all access to SEPPI-facilitated meetings, which has

a very high impact. Standard defences for availability and against denial of service would apply here, such as rate limitation, captchas, or multiple servers with load balancing.

Future work: Having a single server with all relevant information does not only cause a risk to availability, it also presents a broader security and privacy risk from internal as well as powerful external actors. Meeting information that could otherwise have been gathered through a long-term passive network attack on emails would now be available instantaneously at a single server, increasing its attractiveness as an attack target.

Solutions to this could involve distributing the information in various ways. It looks feasible using standard techniques to extend the SEPPI architecture to one with servers operating in a federation model. Much more challenging would be to remove the separation between clients and servers and move to a peer-to-peer implementation.

Other than by distributing the information, internal attacks and some level of server corruption can also be mitigated by making the meeting information available to the server only when strictly needed. The WebAuthn Psuedo-Random Function (PRF) extension may support this, by encrypting user-sensitive information on the field level in a way that makes it only accessible by the corresponding meeting users. An encryption key could be generated for each meeting, available only to the meeting users, and exposed to SEPPI after the user authenticates shortly before the meeting [15].

Configurable security and privacy options are currently only provided on a per-meeting basis. This model could be extended to include per-user options. These could include additional options such as on visibility of user data like email address, as well as overlapping with the existing per-meeting ones. They might indicate minimum required settings on meetings – e.g., a user registering that they would never accept the recording of meetings they are in. This would enable policy conflict situations that would then need resolving.

The SEPPI architecture describes a single fixed UC instance, which the prototype indeed implements through a public Jitsi instance. A more flexible architecture would allow the meeting organizer to select a UC instance per meeting. This might be from SEPPI-provided instances and/or organizer-provided ones. A selection criterion might be whether an instance is located within a given region, in line with data protection considerations.

While SEPPI comprehensively reduces the sensitivity of information transmitted by email, one could imagine abandoning email in this context altogether. As explained above, SEPPI already allows this for users who are prepared to poll the system rather than receive any notifications. However, other channels for notifications are also conceptually possible, though would bring their own challenges. For example, Signal is based on phone numbers, so SEPPI account creation would then require provision of a valid phone number. Note that disposable phone numbers are harder to obtain than disposable emails.

Currently SEPPI exists as an architecture, with a limited private prototype implementation. A valuable future work, albeit with more development than research effort, would be to contribute a full, securely coded, open-source implementation, e.g., to the Jitsi community.

REFERENCES

[1] 8x8, Inc. Jitsi Meet. https://meet.jit.si/, 2023.

[2] Mashari Alatawi and Nitesh Saxena. SoK: An Analysis of End-to-End Encryption and Authentication Ceremonies in Secure Messaging Systems. In *Proceedings of the 16th ACM Conference on Security and Privacy in Wireless and Mobile Networks*, WiSec '23, pages 187–201, New York, NY, USA, June 2023. Association for Computing Machinery.

[3] Aaron Bangor, Philip Kortum, and James Miller. Determining what individual SUS scores mean: Adding an adjective rating scale. *Journal of usability studies*, 4(3):114–123, 2009.

[4] Manuel Barbosa, Alexandra Boldyreva, Shan Chen, and Bogdan Warinschi. Provable security analysis of FIDO2. In *Advances in Cryptology–CRYPTO 2021: 41st Annual International Cryptology Conference, CRYPTO 2021, Virtual Event, August 16–20, 2021, Proceedings, Part III 41*, pages 125–156. Springer, 2021.

[5] Josh Blum, Simon Booth, Oded Gal, Maxwell Krohn, Julia Len, Karan Lyons, Antonio Marcedone, Mike Maxim, Merry Ember Mou, Jack O'Connor, Miles Steele, Matthew Green, Lea Kissner, and Alex Stamos. E2E Encryption for Zoom Meetings. Technical report, Zoom Video Communications, Inc., June 2020.

[6] John Brooke. SUS: A quick and dirty usability scale. *Usability evaluation in industry*, 189(3):189–194, 1996.

[7] Fred D Davis. Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS quarterly*, pages 319–340, 1989.

[8] Fred D. Davis. User acceptance of information technology: system characteristics, user perceptions and behavioral impacts. *International Journal of Man-Machine Studies*, 38(3):475–487, 1993.

[9] FIDO Alliance. User Authentication Specifications Overview. https://fidoalliance.org/specifications/. Accessed: 12-02-2023.

[10] Jingjing Guan, Hui Li, Haisong Ye, and Ziming Zhao. A Formal Analysis of the FIDO2 Protocols. In Vijayalakshmi Atluri, Roberto Di Pietro, Christian D. Jensen, and Weizhi Meng, editors, *Computer Security – ESORICS 2022*, pages 3–21, Cham, 2022. Springer Nature Switzerland.

[11] Bo Holst-Christensen and Erik Frøkjær. Security Issues in SMTP-based Email Systems. In *2021 14th CMI International Conference - Critical ICT Infrastructures and Platforms (CMI)*, pages 1–6, Nov 2021.

[12] Dima Kagan, Galit Fuhrmann Alpert, and Michael Fire. Zooming Into Video Conferencing Privacy. *IEEE Transactions on Computational Social Systems*, pages 1–12, 2022.

[13] Chen Ling, Utkucan Balcı, Jeremy Blackburn, and Gianluca Stringhini. A First Look at Zoombombing. In *2021 IEEE Symposium on Security and Privacy (SP)*, pages 1452–1467, 2021.

[14] Robertas Maleckas, Kenneth G. Paterson, and Martin R. Albrecht. Practically-exploitable vulnerabilities in the Jitsi video conferencing system. Cryptology ePrint Archive, Paper 2023/1118, 2023. https://eprint.iacr.org/2023/1118.

[15] Matthew Miller. Encrypting Data in the Browser Using WebAuthn. https://blog.millerti.me/2023/01/22/encrypting-data-in-the-browser-using-webauthn/, January 2023. Accessed: 03-12-2023.

[16] Matthew Miller. SimpleWebAuthn. https://simplewebauthn.dev/, 2023.

[17] Thomas Reisinger, Isabel Wagner, and Eerke Albert Boiten. Security and Privacy in Unified Communication. *ACM Comput. Surv.*, 55(3), February 2022.

[18] Thomas Reisinger, Isabel Wagner, and Eerke Albert Boiten. Unified Communication: What do Digital Activists Need? In *2023 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*, pages 141–149, Delft, Netherlands, July 2023. IEEE Computer Society.

[19] Adrian Reuter, Ahmed Abdelmaksoud, Karima Boudaoud, and Marco Winckler. Usability of End-to-End Encryption in E-Mail Communication. *Frontiers in Big Data*, 4, 2021.