Received 6 February 2023, accepted 11 March 2023, date of publication 22 March 2023, date of current version 27 March 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3260183

PERSPECTIVE

Privacy-Preserving Social Media With Unlinkability and Disclosure

HIDEAKI MIYAJI[1], PO-CHU HSU[1], AND ATSUKO MIYAJI[1,2], (Member, IEEE)

[1]Graduate School of Engineering, Osaka University, Suita, Osaka 565-0871, Japan
[2]JAIST, Nomi, Ishikawa 923-1292, Japan

Corresponding author: Hideaki Miyaji (hideaki@cy2sec.comm.eng.osaka-u.ac.jp)

This work was supported in part by the Japan Society for the Promotion of Science (JSPS) KAKENHI under Grant JP21H03443, in part by the Innovation Platform for Society 5.0 at Ministry of Education, Culture, Sports, Science and Technology (MEXT), and in part by the Joint Supervisory Team (JST) Next Generation Researchers Challenging Research Program under Grant JPMJSP2138.

ABSTRACT Social media (SM) has become a primary communication tool in the modern world, with an ever-increasing volume of users. Many SM users use anonymous nicknames as their public usernames. However, Zhang et al. (2018) were able to demonstrate an attack that can identify users from the contents of their posts. This attack is caused by the fact that two different posts can be guessed to be the same user. Such linking of different posts is called a linkable feature. On the other hand, usually post under an anonymous nickname, but when a post is thrust into the limelight, we may want to claim the post as our own. Unfortunately, however, current SM offers only two options: using an anonymous nickname or publishing under our own name. In other words, the function of disclosure, which is to make some posts public even though they are usually anonymous, has not been realized in existing SM. In this paper, we propose a SM with unlinkability and disclosure simultaneously, which is achieved by applying a commitment scheme. A commitment scheme consists of commitment and decommitment phases. As for unlinkability, we newly introduce a one-time post name, which is a commitment value of nickname and post. As for disclosure, we use a decommitment phase to one-time post name. We also have demonstrated that our SM is practically feasible.

INDEX TERMS Cryptographic scheme, commitment scheme, homomorphic encryption, ring signature, unlinkable posts, social media.

I. INTRODUCTION

The explosive growth of media over the past decade has drastically transformed the World Wide Web, enabling billions of people to freely engage in various activities within a richly heterogeneous environment [3]. In particular, the emergence of social media (SM) networks, such as Twitter and Facebook, has accelerated the speed of information transmission [23]. For instance, Twitter alone generates an average of 500 million tweets per day from 328 million monthly active users as of June 2017, whereas Facebook has generated multiple petabytes of data per day from approximately 1.3 billion daily active users during the same period [26]. SM is used to not only communicate with friends but also share life experiences and express opinions on sociopolitical events and commercial products [26], Consequently, SM has become inseparable from our daily lives.

As SM has ingrained itself into everyday life, various problems have arisen regarding its use. One example is the issue of linkable features, wherein a user may set a pseudonym (nickname) as an username to anonymize their identity. Although such a system is employed by existing SM networks, including Twitter and Instagram [2], [3], posts associated with the same username are interlinked, enabling the user to be identified. In fact, a risk has been reported in [22] that users may be identifiable from published posts using linkable features. The vast amount of user information and availability of up-to-date data in SM easily allow a user U to collect and aggregate other users' information [5]. In other words, there is a privacy risk inherent to SM networks [3], [14].

The associate editor coordinating the review of this manuscript and approving it for publication was Ismail Butun.

Although the protection of privacy is essential, it may also be desirable to control the disclosure of identity. For instance, users may desire to assert ownership of specific posts [6]. In other words, they may want the option to provide their name *in response to certain posts*. However, such a feature is currently not offered by most SM services.

SM users are generally presented with two choices when publishing posts: they may either use their own names, or an anonymous nickname.

A. EXISTING RESEARCH ON PRIVACY PRESERVING OF SM

Several prior studies have been conducted on privacy preservation in SM. Gross and Acquisti [10] focused on the importance of privacy among SM services [10], whereas Cutillo et al. addressed security and privacy problems [7]. Elmendili et al. [8] proposed a new approach to control the spread of spam content and malicious profiles, thereby preventing privacy breach attacks [8].

Zhang et al. [26] demonstrated for the first time how a user's username can be identified from post content [26], representing an attack caused by linkable features. As a preventative measure, they proposed a framework for differential privacy-preserving SM data outsourcing. Their differential privacy mechanism is based on the novel notion of ε-text indistinguishability, which they proposed to thwart text-based user-linkage attacks. Although their approach successfully achieved unlinkable features, they were unable to achieve disclosure features simultaneously.

In 2016, Ma et al. conducted an online experiment to study the relationship between post messages and the willingness to disclose personal information, examining the relationship between identification and audience type [18]. Within their study, they considered the need for a disclosure feature in SM services, as well as the potential risks associated with such a feature. Currently, no mainstream SM networks provide unlinkability and disclosure concurrently, which increases the privacy risks associated with these services.

B. OUR CONTRIBUTION

In this paper, we propose the Privacy-Preserving Social Media with Disclosure (PPSMD), which offers both unlinkability and disclosure simultaneously, thereby enabling pseudonymous posting.

In PPSMD, users may generate a one-time post name using a commitment scheme. These names are associated with individual posts, remaining mutually unlinkable. As a result, attacks in [26] can be prevented. Furthermore, users may control the disclosure of their identity respective to specific posts using a function of the decommitment phase. Thus, even if one post is disclosed, all other posts made by the same user remain anonymous. In addition, the use of ring signatures anonymously guarantees all posts to be associated with their legitimate users. We have proven PPSMD to successfully prevent username recovery and spoofing attacks owing to the hiding and binding properties of its underlying commitment scheme. Finally, we confirmed PPSMD to yield satisfactory performance by conducting experiments with an implementation.

PPSMD satisfies the following features, proof, and implementation:

1) **Privacy-Preserving Unlinkable Posting**: By introducing a one-time post name $\mathsf{com}_{i,j}$ associated with $\mathsf{username}_i$, posts $\mathbf{M}_{i,j}$ sent by the same U_i are mutually unlinkable. Furthermore, as $\mathsf{com}_{i,j}$ is a commitment value, $\mathsf{username}_i$ is difficult to recover.
2) **Disclosure**: When U_i wishes to disclose a post $\mathbf{M}_{i,j_\ell}$, the post is linked to the user. However, any other post $\mathbf{M}_{i,j_k}$ is not linked to $\mathbf{M}_{i,j_\ell}$ or U_i.
3) **Anonymous Authentication of Posts:** By generating ring signatures associated with posts, each post is verified to be sent by a legitimate user.
4) **Security Measures Against Username Recovery Attacks:** $\mathsf{randomname}_i$ cannot be recovered from the commitment value.
5) **Security Measures Against Spoofing Attacks:** Users cannot be impersonated in the transmission (publication) of posts.
6) **Practicality**: Our experiments found the time required for one post by a user to be 57.94 ms and that required for Manager to verify a ring signature to be 218.07 ms in 100 users and 10000 posts (each user publishes 100 posts). The storage cost of Manager for 10000 posts was 143.66 MB. All experiments were conducted on a CPU based on AMD EPYC 7601.

This paper is the final version of the study presented at CANDAR 2022 [19], wherein only features (1) and (2) were achieved. In this final version, we implemented features (3)-(6) into PPSMD.

C. PAPER ORGANIZATION

The remainder of this paper is organized as follows. Section II summarizes our commitment scheme and other definitions. Section III summarizes existing related works on unlinkability and disclosure. Our proposed PPSMD is presented in Section IV. The details of its implementation and its evaluation are described in Section V. Section VI discusses the issues that need to be addressed in contemporary research and future works related to privacy. Finally, the paper is concluded in Section VII.

II. PRELIMINARIES

The following section presents the definitions pertaining to a commitment scheme and all other building blocks used in this paper.

- 1^k: security parameter
- $\mathtt{pp}$: public parameter
- Hash: hash function
- Writer W: users who publish posts
- Reader R: users who read posts
- Manager: server

- $\mathcal{M}$: message space
- $\mathcal{C}$: encryption space
- space: server for SM posts
- PPSMD: our proposed SM
- U_i: user who joins PPSMD
- $(\mathtt{rsk}_i, \mathtt{rpk}_i)$: secret and public keys of user U_i used in the ring signature
- U: user's public key group
- $\mathsf{U}_{\mathtt{rpk}}$: user's public key group of ring signature
- KeyGen: algorithm that generates $(\mathtt{sk}_i, \mathtt{pk}_i)$ automatically
- $\varepsilon(k)$: negligible function in k
- $\mathsf{username}_i$: registered name of user U_i
- $\mathsf{randomname}_i = \mathsf{Hash}(\mathbf{M}_{i,j}||\mathsf{username}_i)$: randomized pseudonym of U_i used for an unlinkable post
- $\mathbf{M}_{i,j}$: j-th post message sent by user U_i
- $\mathsf{ck}_{i,j}$: commitment key associated with user U_i's post message $\mathbf{M}_{i,j}$
- $\mathsf{Commit}(\mathsf{ck}_{i,j}, \mathsf{randomname}_i)$: probabilistic commitment scheme applied to PPSMD
- $\mathsf{com}_{i,j}$: commitment value of $\mathsf{randomname}_i$ and $\mathbf{M}_{i,j}$, referred to as the "one-time post name".

Definition 1 (Commitment Scheme [6]): A commitment scheme is a two-phase protocol scheme between two probabilistic polynomial-time parties W and R.

During the first phase (commitment phase), W commits message string a to a pair of keys (com, dec) by executing $(\mathsf{com}, \mathsf{dec}) \longleftarrow \mathsf{W}(1^k, \mathsf{a}, PP)$. Then, W sends commitment string com to R.

During the second phase (decommitment phase), W sends keys dec (decommitment string) with a to R. Then, R verifies the decommitment string by executing R(com, dec). If the result is invalid, R(com, dec) outputs a special string $\perp$, indicating that R rejected the decommitment of W. Otherwise, R(com, dec) efficiently computes string a revealed by W, and verifies whether a was selected by W during the first phase.

Let us consider the *KeyGen*, *Commit*, and *Decommit* algorithms, which have 1^k as an implicit input:

- *KeyGen* A PPT algorithm that outputs the public parameters $PP \in \{0,1\}^{poly(k)}$ containing a definition of the message space $\mathcal{M}$
- *Commit* A PPT algorithm that receives the public parameters PP and message $x \in \mathcal{M}$ as input, and outputs $c, r \in \{0,1\}^{poly(k)}$
- *Decommit* A deterministic polynomial-time algorithm that receives the public parameters PP, message $x \in \mathcal{M}$, and values $c, r \in \{0,1\}^{poly(k)}$ as input, and outputs a bit $b \in \{0,1\}$

A secure commitment scheme must satisfy binding properties and hiding properties. In this studdy, we applied a probabilistic commitment scheme with different commitment values for the same input.

Definition 2 (Probabilistic Commitment Scheme): Let k be a security parameter for a given x and x' where $x, x' \in \{0,1\}^*(x = x')$, and Commit is a commitment scheme. We assume Commit to be a probabilistic commitment scheme if the following holds:

$$\Pr[\mathsf{Commit}(x) = \mathsf{Commit}(x')] < \varepsilon(k).$$

The following section describes the security properties of the commitment scheme Commit(com, dec).

Definition 3 (Computational Binding Property [1]): Let Commit be a commitment scheme, com be a commitment string, dec be a decommitment string, and $\mathcal{A}$ be a PPT adversary. The commitment scheme satisfies the binding property if the following equation is satisfied:

$$\Pr\left[\begin{array}{c}\mathcal{A}(\mathsf{com}) \rightarrow (\mathsf{dec}, \mathsf{dec}') \\ \text{s.t. } \mathsf{Commit}(\mathsf{dec}) = \mathsf{Commit}(\mathsf{dec}') = \mathsf{com} \\ \wedge \mathsf{dec} \neq \mathsf{dec}'\end{array}\right] < \varepsilon(k)$$

We now define the computational hiding property of a commitment scheme.

Definition 4 (Computational Hiding Property [1], [11]): Let PPT adversary $\mathcal{A}$ be given a commitment string $\mathsf{com} \leftarrow \mathsf{Commit}$, where com is constructed by $\mathsf{x} \in \mathcal{C}$. We assume that the commitment scheme satisfies the computational hiding property if the following holds:

$$|\Pr[\mathcal{A}(\mathsf{com}) = 1] - \Pr[\mathcal{A}(U) = 1]| < \varepsilon(k)$$

Definition 5 (Collision-Resistance): We have an arbitrary probabilistic polynomial-time algorithm *Adv*, given a description of the hash function and the length parameter as inputs. If the probability of *Adv* outputting $x, x' \in \{0,1\}^k$ that satisfies $x \neq x'$ *and* $f(x) = f(x')$ is negligible, then the function is a collision-resistant hash function.

$$\Pr[Adv(f, 1^k) \rightarrow (x, x')\ s.t.\ x \neq x', f(x) = f(x')] < \varepsilon(k).$$

Definition 6 (A Ring Signature Scheme [24]): A ring signature scheme RS = (RGen, RSign, RVerify) is constructed using three polynomial-time algorithms.

- $\mathsf{RGen}(1^k)$: Receives a security parameter k as input, and outputs a private/public key pair $(\mathtt{rsk}, \mathtt{rpk})$.
- $\mathsf{RSign}(\mathtt{rsk}_s, m, \mathsf{U})$: Receives a message $m \in \mathcal{M}$, group $U = \{\mathtt{rpk}_1, \mathtt{rpk}_2, \ldots, \mathtt{rpk}_n\}$ of the signers' public keys, and private key $\mathtt{rsk}_s (1 \leq s \leq n)$ of member $U_s (1 \leq s \leq n)$ as input, and outputs a ring signature $\mathsf{RSign}(\mathtt{rsk}_s, m, U) \rightarrow \sigma_R$.
- $\mathsf{RVerify}(U, m, \sigma_R)$: Receives a ring signature σ_R, message m, and $U = \{\mathtt{rpk}_1, \mathtt{rpk}_2, \ldots, \mathtt{rpk}_n\}$ as input, and outputs "valid" if $\mathsf{RVerify}(U, m, \sigma_R) = 1$ is satisfied or "$\perp$" otherwise.

A secure ring signature scheme must satisfy existential unforgeability and signer ambiguity.

An adversary's advantage $Adv_{RS,\mathcal{A}}$ is defined as its probability of success in the following game between a challenger C and adversary $\mathcal{A}$:

- Setup. The challenger runs the KeyGen algorithm, and $PK_1, \ldots, PK_n$ is given to $\mathcal{A}$.

- **RSign**Queries. Proceeding adaptively, $\mathcal{A}$ requests a signature on a message m for a group $U = \{PK_1, \ldots, PK_n\}$ of signers' public keys; C returns a ring signature $\mathsf{RSign}(SK_s, m, U) \rightarrow \sigma_i$
- Output. Eventually, $\mathcal{A}$ outputs σ on a message m for U and wins the game if
  1) The message m has not been requested to the RSign oracle for U.
  2) $\mathsf{RVerify}(m, U, \sigma_R) = 1$.

Definition 7 (Existential Unforgeability [24]): A forger $\mathcal{A}(t, q_H, q_S, \varepsilon)$ breaches a ring signature scheme RS if $\mathcal{A}$ runs at most time t and makes at most q_S signature queries and q_H general queries to the hash function, and $Adv_{RS,\mathcal{A}}$ is at least ε. A ring signature scheme $\mathcal{A}(t, q_H, q_S, \varepsilon)$ is existentially unforgeable under an adaptive chosen-message attack if no forger $Adv_{RS,\mathcal{A}}|(t, q_H, q_S, \varepsilon)$ breaks it.

Definition 8 (Signer Ambiguity [24]): A ring signature scheme is said to have an unconditional signer ambiguity if for any group $U = \{PK_1, \ldots, PK_n\}$ of users' public keys, message m, and signature $\mathsf{RSign}(SK_s, m, U) \rightarrow \sigma$, no verifier $\mathcal{A}$ with arbitrary computing resources can identify the actual signer with a probability higher than that of a random guess. That is, $\mathcal{A}$ can only output the actual signer indexed by s with a probability no higher than $1/n$.

III. RELATED WORKS

This section describes related works on privacy with unlinkability and disclosure. First, we introduce related work that satisfies unlinkability in Section III-A. In this section, we describe how they achieve unlinkability. Next, in Section III-B, we explain the importance of disclosure in SM. Finally, in Section III-C, we discuss a related work of SM satisfying unlinkability.

A. EXISTING RELATED WORKS THAT SATISFY UNLINKABILITY [16]

This section describes related works proposed by Liu et al. that satisfy unlinkability in vehicular networks [16]. In vehicular networks, each vehicle receives messages before taking action to avoid interaction with other vehicles. Most of them do not provide privacy protection. For example, the sensed data could reveal the capacity of a vehicle's sensor and hence reveal the personal information of the vehicle [13]. Another concern is location privacy [17] since the location of a vehicle is closely related to its driver. To achieve location privacy, It is desirable to use unlinkable pseudonyms that are periodically changed when publishing messages.

They used the 0-1 encoding technology [15] to achieve location privacy. They included the different random numbers in messages by using 0-1 encoding technology. This made it difficult to infer the original messages and achieved unlinkability.

B. IMPORTANCE OF SM SATISFYING DISCLOSURE [18]

In this section, we explain the importance of disclosure in SM as investigated by Ma et al [18]. They investigated the intimacy and disclosure of posted messages. In their study, they described that the more anonymous the content, the more willing to disclose information about themselves.

In SM, it is necessary to be unlinkable with its own information to protect privacy. On the other hand, it is difficult to propose an SM that satisfies both unlinkability and disclosure at the same time, and there is no SM that satisfies both unlinkability and disclosure.

C. EXISTING SM SATISFIES UNLINKABILITY [26]

In this section, we describe the method proposed by Zhang et al. (ZSZZH) [26]. They proposed SM that satisfies unlinkability using perturbations and differential privacy, against user-linkage attacks. They describe two methods of user-linkage attacks.

1) By finding some target users through random browsing or searching, an attacker can obtain textual data on target users with real IDs. Then, he/she can link the corresponding anonymous IDs to the real IDs.
2) By finding some interesting posts/tweets in the anonymized dataset, an attacker can find the real users by simply matching text on social networks. Once the real users are located, the attacker can learn their latest information.

They performed perturbations on the posted content to prevent user-linkage attacks. Their proposed defense against user-linkage attacks consists of three steps. First, they mapped the intact data of all users to a high-dimensional user keyword matrix. Second, they added controlled noise to the user keyword matrix to satisfy differential privacy. Finally, they disclosed the user keyword matrix to the data users and anonymize each user ID.

However, since unlinkability depends on the parameters used for perturbation, it is not always possible to achieve unlinkability for every post. Furthermore, this method does not have a function that allows users to disclose themselves.

D. THE SUMMARY OF RELATED WORKS

Liu et al. showed that unlinkability is necessary to protect privacy and Ma et al. explained the importance of disclosure in SM. However, although the SM of Zhang et al. can satisfy unlinkability, it can not satisfy disclosure simultaneously. In addition, their unlinkability depends on the parameters of perturbation.

To solve these problems, we propose PPSMD in which each post satisfies a function of unlinkability and disclosure at the same time. Remark that, by realizing unlinkability with a commitment scheme, it is possible to realize SM with the feature that all published posts are unlinkable.

IV. OUR PRIVACY-PRESERVING SOCIAL MEDIA WITH UNLINKABILITY AND DISCLOSURE

This section describes our privacy-preserving social media with unlinkability and disclosure (PPSMD), which satisfies the Feature 1.

Feature 1 (Feature of PPSMD*):*

1) **Privacy-Preserving Unlinkable Posting**: All posts sent by a user U_i are mutually unlinkable. Furthermore, any post with a one-time username and signature cannot reveal personal information associated with $\mathsf{username}_i$.
2) **Disclosure**: When U_i wishes to disclose a post $\mathbf{M}_{i,j_\ell}$, the post is linked to the user. However, any other post $\mathbf{M}_{i,j_k}$ is not linked to $\mathbf{M}_{i,j_\ell}$ or U_i.
3) **Anonymous Authentication for Posts:** All posts are anonymously verified to be sent by legitimate users.
4) **Security Against Username Recovery Attacks:** $\mathsf{randomname}_{i,j}$ cannot be recovered from the commitment value.
5) **Security Against Spoofing Attacks:** Users cannot be impersonated during post transmission.

We tested an implementation of our PPSMD protocol to verify the time required for a user to make a post, as well as that required for a Manager to disclose each post by verifying its authenticity.

A. CONSTRUCTION

PPSMD comprises the following algorithms:

1) Registration phase by user U_i and Manager (Algorithm 1)
   - (User U_i): generates secret and public keys for a ring signature $(\mathtt{rsk}_i, \mathtt{rpk}_i)$, and registers their name $\mathsf{username}_i$ together with their public key $\mathtt{rpk}_i$.
   - (Manager): publishes the set of public keys of all users $\mathsf{U}_{\mathtt{rpk}}$.
2) Unlinkable post phase by user U_i (Algorithm 2)
   - (User U_i): constructs a one-time post name $\mathsf{com}_{i,j}$ and generates a ring signature $\sigma_{R_{i,j}}$.
3) Disclose post phase by user U_i and Manager (Algorithm 3)[1]
   - (User U_i): sends $(\mathsf{username}'_i, \mathsf{ck}'_{i,j})$ for the decommitment value.
   - (Manager): verifies that the correct registered name was published.
4) Read phase by viewer U_s (Algorithm 4)
   - (User U_s): verifies whether it satisfies a $\mathsf{RVerify}(U_{\mathtt{rpk}}, \{\mathbf{M}_{i,j}, \mathsf{com}_{i,j}\}, \sigma_{R_{i,j}}) = 1$.

The following algorithm corresponds to the detailed structure of PPSMD.

[1]The verification steps from (2) to (5) in Algorithm 3 are executed by a Manager.

Algorithm 1 Registration Phase by User U_i and Manager

Each user U_i generates a secret and public key corresponding to a ring signature $(\mathtt{rsk}_i, \mathtt{rpk}_i)$, and registers their $\mathsf{username}_i$ together with their public key $\mathtt{rpk}_i$.
U_i sends $\mathtt{rpk}_i$ to Manager.
Manager collects public keys $\mathtt{rpk}_i$ from all users, and generates $U_{\mathtt{rpk}} = \{\mathtt{rpk}_1, \mathtt{rpk}_2, \ldots, \mathtt{rpk}_n\}$.

Algorithm 2 Unlinkable Post Phase by User U_i

Input: registered name $\mathsf{username}_i$, U_i's j-th message $\mathbf{M}_{i,j}$, public key $U_{\mathtt{rpk}}$
Output: $\left(\{\mathbf{M}_{i,j}, \mathsf{com}_{i,j}\}, \sigma_{R_{i,j}}\right)$
U_i constructs $\mathsf{randomname}_{i,j} = \mathsf{Hash}(\mathbf{M}_{i,j} || \mathsf{username}_i)$ for $\mathbf{M}_{i,j}$.
U_i randomly generates a key (commitment key) $\mathsf{ck}_{i,j}$.
U_i constructs a one-time post name $\mathsf{com}_{i,j}$ using $\mathsf{randomname}_{i,j}$ and commitment key $\mathsf{ck}_{i,j}$. $\mathsf{Commit}(\mathsf{ck}_{i,j}, \mathsf{randomname}_{i,j}) = \mathsf{com}_{i,j}$.
U_i constructs a ring signature $\sigma_{R_{i,j}} = \mathsf{RSign}\left(\mathtt{rsk}_i, \{\mathbf{M}_{i,j}, \mathsf{com}_{i,j}\}, U_{\mathtt{rpk}}\right)$.
U_i posts $\left(\{\mathbf{M}_{i,j}, \mathsf{com}_{i,j}\}, \sigma_{R_{i,j}}\right)$.
return $\left(\{\mathbf{M}_{i,j}, \mathsf{com}_{i,j}\}, \sigma_{R_{i,j}}\right)$.

Algorithm 3 Disclose Post Phase by User U_i and Manager

Input: $\mathsf{username}'_i$ (registered name) of user U_i, commitment key $\mathsf{ck}'_{i,j}$, and message (post text) $\mathbf{M}_{i,j}$
Output: $\perp$ or $(\mathbf{M}_{i,j}, \mathsf{username}'_i)$
1: U_i sends $(\mathsf{username}'_i, \mathsf{ck}'_{i,j})$ for the decommitment value to Manager.
2: Manager constructs $\mathsf{randomname}'_{i,j}$ from $\mathsf{randomname}'_{i,j} = \mathsf{Hash}(\mathbf{M}_{i,j} || \mathsf{username}'_i)$.
3: Manager constructs a one-time post name as $\mathsf{Commit}(\mathsf{ck}'_{i,j}, \mathsf{randomname}'_{i,j})$.
4: Manager verifies if the correct username was sent by checking whether

$$\mathsf{Commit}(\mathsf{ck}'_{i,j}, \mathsf{randomname}'_{i,j}) = \mathsf{com}_{i,j}.$$

5: Manager outputs "there is no match in the space", if $\mathsf{Commit}(\mathsf{ck}'_{i,j}, \mathsf{randomname}'_{i,j}) \neq \mathsf{com}_{i,j}$. Conversely, Manager outputs $\{\mathbf{M}_{i,j}, \mathsf{username}'_i\}$ to U_i, and $(\mathbf{M}_{i,j}, \mathsf{username}'_i)$ is published.
6: **return** $\perp$ or $(\mathbf{M}_{i,j}, \mathsf{username}'_i)$.

Algorithm 4 Read Phase by Viewer U_s

Input: public key $U_{\mathtt{rpk}}$, $\{\mathbf{M}_{i,j}, \mathsf{com}_{i,j}\}$, ring signature $\sigma_{R_{i,j}}$
Output: 1 or $\perp$.
User U_s verifies whether $\mathsf{RVerify}(U_{\mathtt{rpk}}, \{\mathbf{M}_{i,j}, \mathsf{com}_{i,j}\}, \sigma_{R_{i,j}}) = 1$ is satisfied.
return 1 or $\perp$.

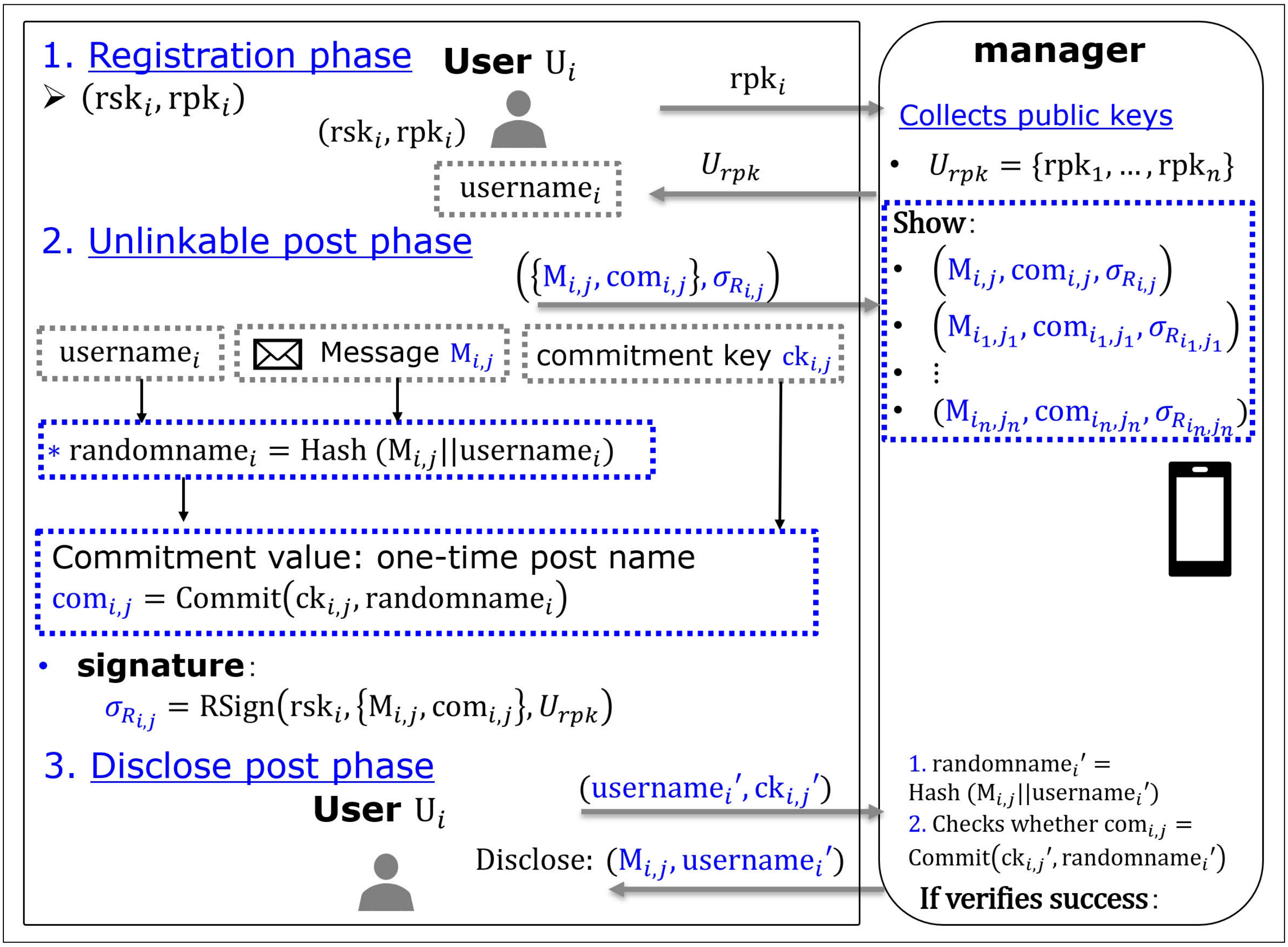


FIGURE 1. Our proposed scheme PPSMD.

B. SECURITY

The following subsection discusses the security of PPSMD. We first define the username recovery attack in Attack 1, and the spoofing attack in Attack 2.

Attack 1 (Username Recovery Attack in Unlinkable Post Phase): In the unlinkable post phase, an attacker may recover $\mathsf{randomname}_{i,j}$ from the commitment value $\mathsf{com}_{i,j}$.

Attack 2 (Spoofing Attack in Disclosed Post Phase): In the disclosed post phase, an attacker may send a different registered name or commitment key.

Theorem 1 illustrates that PPSMD prevents username recovery attacks (Attack 1), whereas Theorem 2 shows that PPSMD prevents spoofing attacks (Attack 2).

Theorem 1: Let $\mathbf{M}_{i,j}$ be a message value, $\mathsf{com}_{i,j}$ be a commitment value, Commit be a commitment scheme, and $\mathsf{randomname}_{i,j}$ be constructed from $\mathsf{username}_i$ of U_i. The commitment value $\mathsf{com}_{i,j}$ is generated from $\mathsf{randomname}_{i,j}$ and a commitment key $\mathsf{ck}_{i,j}$ in the unlinkable post phase. When the computationally hiding property of the commitment scheme is satisfied, then it is difficult to execute a username recovery attack in PPSMD.

Proof: Assume that $\mathcal{A}$ is an adversary seeking to execute a username recovery cttack. In other words, $\mathcal{A}$ can obtain $\mathsf{randomname}_{i,j}$ from the post-name $\mathsf{com}_{i,j}$ of PPSMD. We verify whether a different adversary $\mathcal{A}'$ can break the computationally hiding property of the commitment scheme.

First, the oracle of the computationally hiding property constructs a value y, either by commitment scheme $\mathsf{Commit}(\mathsf{ck}_{i,j}, \mathsf{randomname}_{i,j})$ or uniform randomness. The oracle then sends y to adversary $\mathcal{A}'$, who sends y to $\mathcal{A}$. If y is a value constructed from the commitment scheme, then $\mathcal{A}$ can derive the following Equation (1):

$$\left|\Pr_{\mathsf{y}}\left[\mathcal{A}(\mathsf{y}) \rightarrow (\mathsf{randomname}_{i,j}, \mathsf{ck}_{i,j})\right]\right| > \varepsilon(k) \quad (1)$$

$\mathcal{A}$ is thus able to obtain $(\mathsf{randomname}_{i,j}, \mathsf{ck}_{i,j})$ from the commitment value y. Subsequently, $\mathcal{A}$ sends $(\mathsf{randomname}_{i,j}, \mathsf{ck}_{i,j})$ to a second adversary $\mathcal{A}'$, who in turn sends $(\mathsf{randomname}_{i,j}, \mathsf{ck}_{i,j})$ to the oracle. The oracle can easily verify whether $\mathsf{Commit}(\mathsf{randomname}_{i,j}, \mathsf{ck}_{i,j}) = \mathsf{y}$ satisfies.

If y is constructed from a uniform distribution, $\mathcal{A}$ cannot output any value from y, which also informs us that y was constructed from a uniform distribution. Using this information, $\mathcal{A}'$ sends the oracle a message indicating that y is uniformly distributed.

Consequently, $\mathcal{A}'$ can obtain the input value sent by the oracle irrespective of the construction of y.

From the contraposition, if the commitment scheme features a computationally hiding property, a username recovery attack is difficult to execute. ∎

Theorem 2: Let $\mathbf{M}_{i,j}$ be a message value, $\mathsf{randomname}_{i,j}$ be constructed from $\mathsf{username}_i$ of U_i, Commit be a commitment scheme, and $\mathsf{com}_{i,j}$ be constructed from $\mathsf{randomname}_{i,j}$ and a commitment key $\mathsf{ck}_{i,j}$ in the unlinkable post phase. When the computationally binding property of the commitment scheme holds, then it is difficult to execute a spoofing attack in PPSMD.

Proof: Assume that $\mathcal{A}$ is an adversary seeking to execute a spoofing attack. We verify whether another adversary $\mathcal{A}'$ can breach the computational binding property of the commitment scheme. First, the oracle of the computationally binding property generates the commitment value $\mathsf{com}_{i,j}$ from $\mathsf{randomname}_{i,j}$ and a commitment key $\mathsf{ck}_{i,j}$, according to Equation (2):

$$\mathsf{Commit}(\mathsf{randomname}_{i,j}, \mathsf{ck}_{i,j}) = \mathsf{com}_{i,j} \tag{2}$$

Then the oracle sends $\mathsf{com}_{i,j}$ to adversary $\mathcal{A}'$. $\mathcal{A}'$ then sends $\mathsf{com}_{i,j}$ to another adversary $\mathcal{A}$, who can execute a spoofing attack and generate Equation (3).

$$\begin{aligned} \mathsf{com}_{i,j} &= \mathsf{Commit}(\mathsf{randomname}_{i_1}, \mathsf{ck}_{i_1,j_1}) \\ &= \mathsf{Commit}(\mathsf{randomname}_{i_2}, \mathsf{ck}_{i_2,j_2}) \\ &\quad \wedge \mathsf{randomname}_{i_1} \neq \mathsf{randomname}_{i_2} \\ &\quad \wedge \mathsf{ck}_{i_1,j_1} \neq \mathsf{ck}_{i_2,j_2} \end{aligned} \tag{3}$$

Adversary $\mathcal{A}$ is able to obtain two different inputs $((\mathsf{randomname}_{i_1}, \mathsf{ck}_{i_1,j_1}), (\mathsf{randomname}_{i_2}, \mathsf{ck}_{i_2,j_2}))$ that construct the same commitment value $\mathsf{com}_{i,j}$. We set $\mathsf{dec} = (\mathsf{randomname}_{i_1}, \mathsf{ck}_{i_1,j_1})$ and $\mathsf{dec}' = (\mathsf{randomname}_{i_2}, \mathsf{ck}_{i_2,j_2})$. Then, $\mathcal{A}$ sends $(\mathsf{dec}, \mathsf{dec}')$ to $\mathcal{A}'$. $\mathcal{A}'$ derives Equation (4):

$$\Pr\left[\begin{array}{c} \mathcal{A}'(\mathsf{com}_{i,j}) \to (\mathsf{dec}, \mathsf{dec}') \\ \text{s.t. } \mathsf{Commit}(\mathsf{dec}) = \mathsf{Commit}(\mathsf{dec}') = \mathsf{com}_{i,j} \\ \wedge \mathsf{dec} \neq \mathsf{dec}' \end{array}\right] > \varepsilon(k). \tag{4}$$

Equation (4) shows that $\mathcal{A}'$ is able to break the computationally binding property if they obtain $\mathsf{dec} = (\mathsf{randomname}^1_{i,j}, \mathsf{ck}^1_{i,j})$ and $\mathsf{dec}' = (\mathsf{randomname}^2_{i,j}, \mathsf{ck}^2_{i,j})$. Consequently, $\mathcal{A}'$ can obtain the oracle's solution.

From the contraposition, if the computationally binding property holds, a spoofing attack is difficult to execute. ∎

In addition to security against two types of attacks, we demonstrated PPSMD to satisfy the following three features: Privacy-Preserving Unlinkable Posting (Feature (1)), Disclosure (Feature (2)), and Anonymous Authentication (Feature (3)).

Theorem 3: PPSMD satisfies Privacy-Preserving Unlinkable Posting (Feature (1)), Disclosure (Feature (2)), and Anonymous Authentication (Feature (3)) in Feature 1.

Proof:

- Privacy-Preserving Unlinkable Posting (Feature (1)):

Each post uses a one-time post name $\mathsf{com}_{i,j}$, generated by a probabilistic commitment scheme. All one-time post names are mutually unlinkable. Furthermore, the post names cannot reveal any $\mathsf{randomname}$ by the computational hiding property of the probabilistic commitment scheme.

- Disclosure (Feature (2)):

When U_i wants to disclose a post $\mathbf{M}_{i,j}$, then by executing a decommitment phase of the probabilistic commitment scheme, $\mathbf{M}_{i,j}$ can be linked to U_i. However, any other post $\mathbf{M}_{i,\ell}$ is not linked to $\mathbf{M}_{i,j}$ owing to the computational hiding property.

- Anonymous Authentication (Feature (3)):

By generating a corresponding ring signature, each post is anonymously verified to be sent by a legitimate user. ∎

C. THEORETICAL ANALYSIS OF PPSMD

We theoretically analyze how computation cost, communication size, and storage cost in Algorithms 1, 2, 3, and 4 depends on the number of users and the number of posts, which are shown in Table 1. The analysis assumes that there exist n users, and each user U_i publishes p unlinkable posts and reads p posts.

In Algorithm 1, each user generates $\{(\mathtt{rsk}_i, \mathtt{rpk}_i), \mathsf{username}_i, \mathtt{rpk}_i\}$. In this case, the computational cost for each user is $\mathcal{O}(1)$. The computation cost of *manager* is not required since the *manager* does not generate anything. On the other hand, the *manager* receives $\mathtt{rpk}_i$ from all users, the total communication size and the storage cost of the *manager* are $\mathcal{O}(n)$, respectively.

In Algorithm 2, each user publishes unlinkable posts. The computation cost for each user is $\mathcal{O}(np)$. Since n ring signatures need to be generated for one post of one user, the computation cost and communication size for one post of one user is $\mathcal{O}(n)$. Thus, for p posts, the computation cost and communication size for one user is $\mathcal{O}(np)$. Since the manager needs to verify n ring signatures for one post of one user, the computation cost is $\mathcal{O}(n)$. Thus, for n users and p posts, the computation cost is $\mathcal{O}(n^2p)$. Because the total communication size for all n users sending p unlinkable posts is $\mathcal{O}(n^2p)$, the storage cost for the manager is also $\mathcal{O}(n^2p)$.

In Algorithm 3, the user discloses their post. In this phase, the user only needs to send $(\mathsf{username}'_i, \mathsf{ck}'_{i,j})$ to the *manager*, and no computation is required. The *manager* receives np posts and verifies them one by one, so the computation cost of *manager*, total communication size, and the storage cost for *manager* is $\mathcal{O}(np)$, respectively.

In Algorithm 4, since each user only verifies and reads p posts, the user's computation cost is $\mathcal{O}(p)$, and the computation cost of manager is not required. Thus, for n users and p posts, the total communication size is $\mathcal{O}(np)$, and no storage cost is required for the *manager*.

D. DIFFERENCE BETWEEN PPSMD AND ZSZZH

In this subsection, we compare PPSMD with the existing related work of ZSZZH. In PPSMD, Theorem 3 shows that unlinkability and disclosure are simultaneously satisfied. Moreover, PPSMD can prevent user-linkage attacks by making the user's information anonymous, because the user's information cannot be linked from the published post information.

On the other hand, SM of ZSZZH achieves unlinkability using perturbations and differential privacy. In other words, unlinkability and user-linkage attacks depend on the parameter for perturbations. Furthermore, it does not satisfy the disclosure of users' own published posts. Therefore, PPSMD is the first SM that satisfies both unlinkability and disclosure.

Table 2 shows the functional difference between ZSZZH and our proposal.

V. IMPLEMENTATION

The following section examines the implementation and performance of our proposed PPSMD with ring signatures applied. We implemented PPSMD using Python 3.10.6.[2] In the ring signature part, we use the pyring package,[3] which provides a one time ring signature scheme based on libsodium over curve Ed25519 [25]. In the commitment part, we use AES ECB mode provided by the PyCryptodome package. In the hash function part, we use sha256 provided by libhash. Benchmarks are given by an AMD EPYC 7601 CPU.

A. EXPERIMENTAL RESULTS OF PPSMD

In this subsection, we evaluate the following 3 phases of the proposed PPSMD.

1) Registration phase:
   - (User U_i): generates secret and public keys for a ring signature $(\mathtt{rsk}_i, \mathtt{rpk}_i)$, registers their name $\mathsf{username}_i$ together with their public key $\mathtt{rpk}_i$, and gets all user's public keys from Manager.
   - (Manager): publishes the set of public keys of all users $\mathsf{U}_{\mathtt{rpk}}$.
2) Unlinkable post phase:
   - (User U_i): constructs a one-time post name $\mathsf{com}_{i,j}$ and generates a ring signature $\sigma_{R_{i,j}}$.
   - (Manager): verify the ring signature $\sigma_{R_{i,j}}$.
3) Disclose post phase:
   - (User U_i): sends $(\mathsf{username}'_i, \mathsf{ck}'_{i,j})$ for the decommitment value.
   - (Manager): verifies that the correct registered name was published.

We evaluate computation and storage costs, and communication sizes for Registration, Unlinkable post, and Disclose post phase. The computation cost (ms) is calculated for one user and Manager to execute each phase of the process. The communication size (Bytes) is the sum of data a user/Manager sent and received in each phase. The storage cost is the total storage of all public keys and posts kept in Manager. We describe the evaluation conditions as follows.

- Table 3: Computation cost and communication size in Register phase for 10 and 100 users.
- Table 4: Computation cost and communication size in Unlinkable post phase for (10 users, 100 posts), (100 users, 1000 posts), and (100 users, 10000 posts).
- Table 5: Computation cost and communication size in Disclose post phase for (10 users, 100 posts), (100 users, 1000 posts), and (100 users, 10000 posts).

In the registration phase, a user generates both $(\mathtt{rsk}, \mathtt{rpk})$ of a ring signature, and Manager registers all public keys from users. The computation cost of Manager includes the time to register these public keys. Table 3 shows that the computation cost and communication size depend on the number of users. As the number of users increases, the number of public keys created by users increases, and the number of public keys registered by Manager also increases. This is why the computation cost and communication size for the user and Manager increases if the number of users increases seen in Table 3.

In the unlinkable post phase, a user generates both a commitment value and a ring signature, and Manager executes a ring signature verification. The computation costs of generating and verifying ring signatures depend on the number of users.

Let us investigate results on #(users, posts) = (10, 100) and (100, 1000). These results correspond to cases in which the number of users is 10 times but the number of posts for each user is the same. Then, from Table 4, the computation cost of user and Manager for #(users, posts) = (100, 1000) is about 10 and 100 times of that for #(users, posts) = (10, 100), respectively. In the case of (100, 1000), the computation cost is about 10 times larger than in the case of (10, 100) because the number of users is 10 times larger and therefore the computation required to create a ring signature for each user is also 10 times larger. On the other hand, since both the number of users and the posts are 10 times larger, the computation cost for a Manager is about $10 * 10$ times larger than that of (10, 100). For the same reason as the above, the communication size of the unlinkable post for #(users, posts) = (10, 100) is around 77 times of that for #(users, posts) = (10, 100) from Table 4.

Let us investigate results on #(users, posts) = (100, 1000) and (100, 10000). These results correspond to cases in which

[2] The implementation is available at https://github.com/ENLINKER/commitment

[3] https://github.com/bartvm/pyring

TABLE 1. Theoretical analysis of all algorithms in Computation cost and communication size, and Storage costs (we assume there exist *n* users and each user publishes *p* posts).

	computation cost		total communication size	storage cost for
	Manager	user U_i	TOTAL	Manager
Algorithm 1 (Registration phase)	-	$\mathcal{O}(1)$	$\mathcal{O}(n)$	$\mathcal{O}(n)$
Algorithm 2 (Unlinkable post phase)	$\mathcal{O}(n^2p)$	$\mathcal{O}(np)$	$\mathcal{O}(n^2p)$	$\mathcal{O}(n^2p)$
Algorithm 3 (Disclose post phase)	$\mathcal{O}(np)$	-	$\mathcal{O}(np)$	$\mathcal{O}(np)$
Algorithm 4 (Read phase)	-	$\mathcal{O}(p)$	$\mathcal{O}(np)$	-

TABLE 2. Comparison of PPSMD with ZSZZH.

SM	prevent user-linkage attacks	unlinkability	disclosure
ZSZZH [26]	Depends on the parameter	Depends on the parameter	No
Ours	✓	✓	✓

TABLE 3. Computation cost and communication size in Register phase.

#user	computation costs (ms)		communication sizes (KB)
	Manager	user U_i	
10	0.02138	0.07792	4.4
100	0.15081	0.1311	404

the number of a user is the same but the number of posts for each user is 10 times. Comparing #(users, posts)=(100, 1000) to (100, 10000), the total number of posts increases 10 times, so the theoretical value of the ratio of computation cost and communication size from (100, 1000) to (100, 10000) is 10. From Table 4, the ratio of computation cost and communication size for #(users, posts)=(100, 1000) and (100, 10000) is as follows:

- Ratio of computation costs of Manager: $570642.56/57947.75 = 9.84$,
- ratio of computation costs of one user: $5794.33/590.18 = 9.818$, and
- ratio of communication sizes: $142690/14269 = 10$.

Consequently, Table 4 reflects the execution well since the experimental and theoretical values are close. Remark that, the time required for one post by a user is 57 ms when #(users, posts) = (100, 10000).

In the disclose post phase, a user has only to send the necessary data to open a commitment value, and a Manager only needs to verify the commitment value. Remark that the disclose post phase does not use a ring signature. Let us investigate results on #(users, posts) = (10, 100) and (100, 1000). These results correspond to cases in which the number of users is 10 times but the number of posts for each user is the same. Then, from Table 5, the computation cost of user for #(users, posts) = (100, 1000) is almost the same as that for #(users, posts) = (10, 100). Since disclose post phase is independent to ring signature, only the number of posts influence the computation cost for a user. On the other hand, since the total posts are 10 times larger, the computation cost for Manager in #(users, posts) = (100, 1000) is about 10 times larger than that of (10, 100).

Let us investigate results on #(users, posts) = (100, 1000) and (100, 10000). These results correspond to cases in which the number of a user is the same but the number of posts for each user is 10 times. Comparing #(users, posts) = (100, 1000) to (100, 10000), the total number of posts increases 10 times, so the theoretical value of the ratio of computation cost and communication size from (100, 1000) to (100, 10000) is 10. From Table 5, the ratio of computation cost and communication size for #(users, posts) = (100, 1000) and (100, 10000) is as follows:

- Ratio of computation costs of Manager: $218.06973/25.077197 = 8.696$,
- ratio of computation costs of one user: $0.1815/0.02023 = 8.972$, and
- ratio of communication sizes: $1620/162 = 10$.

Consequently, Table 5 reflects the execution well since the experimental and theoretical values are close. Remark that, the time required for sending one post by a user is 0.0018 ms when #(users, posts) = (100, 10000) in disclose post phase.

Table 6 shows the total storage of Manager for all public keys $U_{\mathtt{rpk}}$ and all the post in each case of (10 users, 100 posts), (100 users, 1000 posts), and (100 users, 10000 posts). From Table 6, the cost of post storage linearly increase depending on the increase of the total posts. On the other hand, the cost of all public keys $U_{\mathtt{rpk}}$ linearly increases depending on the increase of the user. The maximum cost of the disclosure post is the same as the storage cost of all posts, so the additional storage for the disclosure of 10000 posts are 143.66 MB.

VI. DISCUSSION

This section summarizes a brief discussion of the issues that need to be addressed in contemporary research and future works beyond the scope of this paper.

Privacy is important in SM, but privacy protection makes it difficult to claim the thoughts of the user. In the same way, privacy protection is essential for any digital technology, but privacy protection can degrade the benefit of these data technologies. For example, Privacy and usability, privacy and copyright are in a trade-off relationship, and it will be important to realize these in a by-design manner. In this paper, we proposed a method to claim the copyright of contents by using a commitment scheme. It will be increasingly important to achieve a balance between privacy protection and trade-offs by using simple security technologies.

TABLE 4. Computation cost and communication size in Unlinkable post phase.

#(user,post)	computation costs (ms)		communication sizes
	Manager	user U_i	(KB)
10 users, 100 posts (each user publishes 10 posts)	587.54	57.87	183.7
100 users, 1000 posts (each user publishes 10 posts)	57947.75	590.18	14269
100 users, 10000 posts (each user publishes 100 posts)	570642.56	5794.33	142690

TABLE 5. Computation cost and communication size in Disclose post phase.

#(user,post)	computation costs (ms)		communication sizes
	Manager	user U_i	(KB)
10 users, 100 posts (each user publishes 10 posts)	2.298494	0.01553	16.2
100 users, 1000 posts (each user publishes 10 posts)	25.077197	0.02023	162
100 users, 10000 posts (each user publishes 100 posts)	218.06973	0.1815	1620

TABLE 6. Storage costs of Manager.

	$U_{\mathtt{rpk}}$ storage (KB)	post storage (KB)	total storage cost (MB)
10 users and 100 posts (each user publishes 10 posts)	0.4	193.4	0.1938
100 users and 1000 posts (each user publishes 10 posts)	4	14366	14.37
100 users and 10000 posts (each user publishes 100 posts)	4	143660	143.664

VII. CONCLUSION

In this paper, we propose privacy-preserving social media with disclosure PPSMD, which satisfies the properties of privacy-preserving unlinkable posting, disclosure, anonymous authentication, security against username recovery attacks, and security against spoofing attacks. Our platform is based on fundamental security technology such as a commitment scheme and ring signature. Moreover, PPSMD is demonstrated to work as a practical application.

Currently, PPSMD is implemented using the P256 elliptic curve ElGamal, which can be employed for any other commitment schemes. For example, we may use a commitment scheme based on the prime factorization problem [9], the discrete logarithm problem [21], or a hash function [12]. Moreover, our platform can also be implemented with a commitment scheme based on lattice cryptography [1], [4], [20], which represents a post-quantum cryptographic approach. Because PPSMD is extendable to a variety of commitment schemes, it is highly versatile in practice.

ACKNOWLEDGMENT

An earlier version of this paper was presented in part at the CANDAR 2022 [DOI: 10.1109/CANDARW57323.2022.00010].

REFERENCES

[1] C. Baum, I. Damgård, V. Lyubashevsky, S. Oechsner, and C. Peikert, "More efficient commitments from structured lattice assumptions," in *Security and Cryptography for Networks* (Lecture Notes in Computer Science), vol. 11035, D. Catalano and R. D. Prisco, Eds. Amalfi, Italy: Springer, Sep. 2018, pp. 368–385.

[2] G. Beigi, "Social media and user privacy," *CoRR*, vol. abs/1806.09786, pp. 1–3, Jun. 2018.

[3] G. Beigi and H. Liu, "A survey on privacy in social media: Identification, mitigation, and applications," *ACM/IMS Trans. Data Sci.*, vol. 1, no. 1, pp. 1–38, Feb. 2020.

[4] F. Benhamouda, S. Krenn, V. Lyubashevsky, and K. Pietrzak, "Efficient zero-knowledge proofs for commitments from learning with errors over rings," in *Computer Security—ESORICS 2015* (Lecture Notes in Computer Science), vol. 9326, G. Pernul, P. Y. A. Ryan, and E. R. Weippl, Eds. Vienna, Austria: Springer, Sep. 2015, pp. 305–325.

[5] J. Bonneau, J. Anderson, and G. Danezis, "Prying data out of a social network," in *Proc. Int. Conf. Adv. Social Netw. Anal. Mining*, N. Memon and R. Alhajj, Eds. Athens, Greece: IEEE Computer Society, Jul. 2009, pp. 249–254.

[6] G. D. Crescenzo, J. Katz, R. Ostrovsky, and D. A. Smith, "Efficient and non-interactive non-malleable commitment," in *Proc. Int. Conf. Theory Appl. Cryptograph. Techn.*, Innsbruck, Austria, May 2001, pp. 40–59.

[7] L. A. Cutillo, R. Molva, and T. Strufe, "Safebook: A privacy-preserving online social network leveraging on real-life trust," *IEEE Commun. Mag.*, vol. 47, no. 12, pp. 94–101, Dec. 2009.

[8] F. Elmendili, A. Moustir, and Y. E. B. E. Idrissi, "Privacy preserving on social networks: New policies and approaches," in *Proc. 3rd Int. Conf. Smart City Appl.*, Tetouan, Morocco, Oct. 2018, pp. 45:1–45:9.

[9] S. Goldwasser, S. Micali, and R. L. Rivest, "A digital signature scheme secure against adaptive chosen-message attacks," *SIAM J. Comput.*, vol. 17, no. 2, pp. 281–308, 1988.

[10] R. Gross and A. Acquisti, "Information revelation and privacy in online social networks," in *Proc. ACM Workshop Privacy Electron. Soc.*, V. Atluri, S. D. C. di Vimercati, and R. Dingledine, Eds., Alexandria, VA, USA, Nov. 2005, pp. 71–80.

[11] I. Haitner, M.-H. Nguyen, S. J. Ong, O. Reingold, and S. Vadhan, "Statistically hiding commitments and statistical zero-knowledge arguments from any one-way function," *SIAM J. Comput.*, vol. 39, no. 3, pp. 1153–1218, Jan. 2009.

[12] S. Halevi and S. Micali, "Practical and provably-secure commitment schemes from collision-free hashing," in *Proc. 16th Annu. Int. Cryptol. Conf.*, Santa Barbara, CA, USA, Aug. 1996, pp. 201–215.

[13] H. Hu, R. Lu, C. Huang, and Z. Zhang, "TripSense: A trust-based vehicular platoon crowdsensing scheme with privacy preservation in VANETs," *Sensors*, vol. 16, no. 6, p. 803, 2016.

[14] S. Ji, W. Li, M. Srivatsa, J. S. He, and R. Beyah, "General graph data deanonymization: From mobility traces to social networks," in *Proc. ACM Trans. Intell. Syst. Technol.*, 2016, pp. 249–254.

[15] H.-Y. Lin and W.-G. Tzeng, "An efficient solution to the millionaires' problem based on homomorphic encryption," in *Proc. 3rd Int. Conf. ACNS*, in Lecture Notes in Computer Science, vol. 3531, J. Ioannidis, A. D. Keromytis, and M. Yung, Eds., Jun. 2005, pp. 456–466.

[16] Z. Liu, J. Weng, J. Guo, J. Ma, F. Huang, H. Sun, and Y. Cheng, "PPTM: A privacy-preserving trust management scheme for emergency message dissemination in space–air–ground-integrated vehicular networks," *IEEE Internet Things J.*, vol. 9, no. 8, pp. 5943–5956, Apr. 2022.

[17] R. Lu, X. Lin, T. Luan, X. Liang, and X. Shen, "Pseudonym changing at social spots: An effective strategy for location privacy in VANETs," *IEEE Trans. Veh. Technol.*, vol. 61, no. 1, pp. 86–96, Jan. 2012.

[18] X. Ma, J. Hancock, and M. Naaman, "Anonymity, intimacy and self-disclosure in social media," in *Proc. CHI Conf. Hum. Factors Comput. Syst.*, J. Kaye, A. Druin, C. Lampe, D. Morris, and J. P. Hourcade, Eds., San Jose, CA, USA, May 2016, pp. 3857–3869.

[19] H. Miyaji, P.-C. Hsu, and A. Miyaji, "Privacy-preserving social media with a disclosure," in *Proc. 10th Int. Symp. Comput. Netw. Workshops (CANDARW)*, Nov. 2022, pp. 337–343.

[20] H. Miyaji, Y. Wang, and A. Miyaji, "Message-restriction-free commitment scheme based on lattice assumption," in *Information Security Practice and Experience* (Lecture Notes in Computer Science), vol. 13107, R. H. Deng et al., Eds. Nanjing, China: Springer, Dec. 2021, pp. 90–105.

[21] T. P. Pedersen, "Non-interactive and information-theoretic secure verifiable secret sharing," in *Proc. 11th Annu. Int. Cryptol. Conf.*, Santa Barbara, CA, USA, Aug. 1991, pp. 129–140.

[22] A. Pfitzmann and M. Hansen, "Anonymity, unlinkability, unobservability, pseudonymity, and identity management—A consolidated proposal for terminology," Version v0.28, Fac. Comput. Sci., Inst. Syst. Archit., TU Dresden, Dresden, Germany, Anon Terminol. Paper, 2013. [Online]. Available: http://dud.inf.tu-dresden.de/Anon_Terminology.shtml

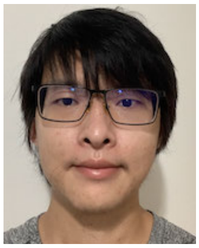
[23] M. Schreiner, T. Fischer, and R. Riedl, "Impact of content characteristics and emotion on behavioral engagement in social media: Literature review and research agenda," *Electron. Commerce Res.*, vol. 21, no. 2, pp. 329–345, Jun. 2021.

[24] K.-A. Shim, "An efficient ring signature scheme from pairings," *Inf. Sci.*, vol. 300, pp. 63–69, Apr. 2015.

[25] N. V. Saberhagen, "Cryptonote V 2.0," Whitepaper_annotated, White Paper, 2013. [Online]. Available: https://www.getmonero.org/ru/resources/research-lab/pubs/whitepaper_annotated.pdf, doi: 10.5281/zenodo.6496895.

[26] J. Zhang, J. Sun, R. Zhang, Y. Zhang, and X. Hu, "Privacy-preserving social media data outsourcing," in *Proc. IEEE Conf. Comput. Commun. (IEEE INFOCOM)*, Honolulu, HI, USA, Apr. 2018, pp. 1106–1114.

HIDEAKI MIYAJI received the B.S. degree from Kanazawa University, in 2018, and the M.Eng. degree from Osaka University, in 2020, where he is currently pursuing the master's degree. His current research interests include cryptography, including lattices, hash functions, and commitment schemes.



PO-CHU HSU received the B.Sc. and M.Sc. degrees in computer science from National Taiwan University, in 2017 and 2020, respectively. He is currently pursuing the Ph.D. degree with Osaka University. His research interests include blockchains and cryptographic protocols.

ATSUKO MIYAJI (Member, IEEE) received the B.Sc., M.Sc., and D.Sc. degrees in mathematics from Osaka University, in 1988, 1990, and 1997, respectively. She joined Panasonic Company Ltd., from 1990 to 1998, and engaged in research and development for secure communication. She was an Associate Professor with the Japan Advanced Institute of Science and Technology (JAIST), in 1998. She joined the Computer Science Department, University of California at Davis, from 2002 to 2003. She has been a Professor with the JAIST, since 2007. She has been a Professor with the Graduate School of Engineering, Osaka University, since 2015. Her research interests include the application of number theory into cryptography and information security. She is a fellow of IPSJ. She is a member of the International Association for Cryptologic Research, the Institute of Electronics, Information and Communication Engineers, the Information Processing Society of Japan, and the Mathematical Society of Japan. She received the Young Paper Award of SCIS'93, in 1993, the Notable Invention Award of the Science and Technology Agency, in 1997, the IPSJ Sakai Special Researcher Award, in 2002, the Standardization Contribution Award, in 2003, the Award for the contribution to Culture of Security, in 2007, the Director-General of the Industrial Science and Technology Policy and Environment Bureau Award, in 2007, the DoCoMo Mobile Science Award, in 2008, the Advanced Data Mining and Applications (ADMA 2010) Best Paper Award, the Prizes for Science and Technology, the Commendation for Science and Technology by the Minister of Education, Culture, Sports, Science and Technology, the International Conference on Applications and Technologies in Information Security (ATIS 2016) Best Paper Award, the 16th IEEE Trustcom 2017 Best Paper Award, the IEICE Milestone Certification, in 2017, the 14th Asia Joint Conference on Information Security (AsiaJCIS 2019) Best Paper Award, the Information Security Applications-20th International Conference (WISA 2020) Best Paper Gold Award, and the Distinguished Educational Practitioners Award, in 2020.