mathematics

MDPI

Article

Privacy Preservation Authentication: Group Secret Handshake with Multiple Groups

Dong Han [1,2], Zhen Li [1,3], Mengyu Wang [4,5,*], Chang Xu [4,*] and Kashif Sharif [1]

[1] School of Computer Science & Technology, Beijing Institute of Technology, Beijing 100081, China
[2] Beijing CCID Software Testing Engineering Technology Center Co., Ltd., Beijing 100048, China
[3] Southeast Institute of Information Technology, Beijing Institute of Technology, Putian 351100, China
[4] School of Cyberspace Science & Technology, Beijing Institute of Technology, Beijing 100081, China
[5] National Computer Network Emergency Response Technical Team/Coordination Center of China, Beijing 100029, China
* Correspondence:alex52811@163.com (M.W.); xuchang@bit.edu.cn (C.X.)

Abstract: The technique of group secret handshake (GSH) has been used to help the members affiliated with the same group in achieving private authentication. After executing GSH protocols, the participants affiliated with the group can compute a shared secret key, or generate a public encryption key while the true participants can self-compute their decryption keys. This paper presents a concrete GSH protocol with Multiple Groups. Only a legitimate member can prove that it belongs to a set of legitimate affiliations, but which affiliation it belongs to will not be leaked. The Group Authority can reveal the real identities of the fellows in the proposed scheme after analyzing the flow of communication. The proposed scheme can provide affiliation-hiding and detectability. In addition, it achieves Perfect Forward Secrecy.

Keywords: ring group signature; authentication; group key exchange; privacy

MSC: 94A60

Citation: Han, D.; Li, Z.; Wang, M.; Xu, C.; Sharif, K. Privacy Preservation Authentication: Group Secret Handshake with Multiple Groups. *Mathematics* **2023**, *11*, 532. https://doi.org/10.3390/math11030532

Academic Editor: Antanas Cenys

Received: 30 November 2022
Revised: 15 January 2023
Accepted: 16 January 2023
Published: 18 January 2023

1. Introduction

Secret handshakes for identifying users within a group are an efficient mechanism, first introduced by Balfanz et al. [1]. Unlike traditional authentication protocols, secret handshake protocols help two participants affiliated with the same group privately identify each other. Only when these protocols are executed successfully the two participants can authenticate each other. Even if the participants have identified each other, they cannot learn the partner's details (e.g., the real identity) besides the affiliation information. The secret handshake protocols are also called affiliation-hiding protocols. These protocols can be applied in many scenarios. For example, if a secret agent A wants to authenticate another agent B, then B can conclude whether A is affiliated to the same group only if B is affiliated to that group. Traditionally, two devices execute an ID-based authenticated key exchange to create a session key. However, as pointed out in [2], these protocols are not secure. In contrast, if two devices perform secret handshakes, one party can identify if the other is authorized. Only if both are authorized, then they will establish a shared session key. Anonymous routing can also be achieved using secret handshake protocols [3].

In Balfanz et al.'s scheme, a group manager generates users' pseudonyms, creates certificates based on the pseudonyms and group secret, and then sends the certificates to the members through an authenticated channel. The group manager generates the certificates for the users just like PKG creates the private keys.

Ideally, a secret handshake protocol should achieve Impersonation Resistance, i.e., a non-group member cannot impersonate a group member to execute the protocol. Moreover, non-group members cannot identify group members (Detection resistance). Traceability should also be provided, i.e., if group members are corrupted, they should be traced.

The concept of group secret handshake (GSH) protocols was introduced in [4]. It also proposed two concrete GSH protocols: an RSA signature-based protocol and a Schnorr signature-based protocol. However, the work in [5,6] by Xu et al., successfully executed attacks in the two GSH schemes. Furthermore, they proposed two new GSH protocols to counter the defined attacks. GSH protocols help the participants of the same groups to authenticate each other. After successfully executing the protocol, the players will generate a shared or public encryption key and their decryption keys [7]. Only honest players can calculate the decryption keys, and each player's decryption key is different. The scheme in [7] needs only one round, but it does not hold detectability. Moreover, executing this protocol requires $O(k)$ pairing operations. The schemes in [5,6] require $O(k)$ multiplication operations, $O(k)$ exponentiation operations, and 1 pairing operation. In addition, they need two rounds. Xu et al., also proposed the first protocol with semi-trusted group authority in [8]. Most of the existing solutions rely on fully trusted authority; however, in this new protocol, the group authority can trace the corrupted users. Unfortunately, the group authority can not impersonate the current honest group members to run the protocol. This scheme needs four pairings.

This paper proposes a two-round GSH based on ring group signatures for multiple groups. Our major contributions are listed below.

1 Unlike existing affiliation-hiding (AH) protocols in real-world organizations with multiple groups, in our proposed GSH protocol, players from different groups can calculate public encryption and secret decryption keys. When the protocol is executed successfully, a player A cannot identify which group the other party B is affiliated to, but A can learn that B is affiliated to one of the groups. No matter if the protocol is executed successfully, the adversary cannot learn any sensitive information.
2 Our protocol can provide Perfect Forward Secrecy, i.e., the previously generated session keys remain secure even when the participants' long-term secrets are leaked. Apart from AH property and perfect forward secrecy, this scheme also holds detectability, impersonation resistance, and traceability.
3 We prove that our scheme provides Perfect Forward Secrecy based on a formal security model. We also prove that this new scheme achieves AH property based on a formal privacy model.

The rest of the article is organized into the following sections. Related works and the building blocks are introduced in Sections 2 and 3. In Section 4, we define the security model and privacy model. In Section 5, we give the details of our GSH protocol along with the security analysis. Section 6 concludes the article.

2. Related Works

To improve the efficiency of the scheme in [1], Castelluccia et al. [9] designed a new SH protocol using public key encryption technology. It is efficient since it is not constructed based on the bilinear map technique. Xu and Yung presented the first two-party secret handshake protocol in [10]. This scheme is designed without a one-time certificate. However, their scheme can only support weak anonymity: k-anonymity [11]. The work in [12] presented two two-party secret handshake protocols. However, it was pointed out by [13] that the protocols do not hold affiliation-hiding property. After Oblivious Signature-Based Envelope (OSBE) was proposed [14], Nasserian and Tsudik presented the ElGamal signature [15] based OSBE scheme [16]. They combined two OSBE schemes to construct a new secret handshake protocol. Zhou et al. [17] pointed out that there exist some attacks in [16] and presented an ElGamal signature-based secret handshake protocol and a DSA [18] signature secret handshake scheme. However, this scheme requires three rounds.

Hoepman [19] presented an SH protocol in which each participant can belong to multiple organizations. The Group Authority (GA) cannot trace the real identities in their schemes according to the communication manuscripts between the two parties. Yamashita and Tanaka [20] also proposed a two-party secret handshake protocol. If the two partici-

pants belong to the same organization, they can execute the protocol successfully. However, Ref. [21] found Yamashita and Tanaka's scheme attackable. That is, the attacks can find that Alice belongs to the groups $G_1, G_2, \ldots, G_n$, even if the attacker does not belong to the same groups $G_1, G_2, \ldots, G_n$.

Ateniese et al. [22] proposed a fuzzy-matched SH protocol based on Fuzzy Identity-Based Encryption technology. However, their scheme does not hold traceability. If the group members are corrupted, GA cannot recover the real identity of the corrupted members. Although a series of secret handshake schemes [23–26] have been proposed, these schemes did not consider multiple players.

In IoT, a device needs to discover other devices around it. However, some devices have private information. Therefore, the devices must identify each other in a privacy-preserving way. Zhou et al. [27] presented a secret handshake method to help a device identify other devices nearby. Their scheme satisfies sensitive attribute secrecy. In their scheme, the objects holding sensitive attributes form a group. A device concludes if the other holds sensitive information by confirming if they have the same group membership. The proposed scheme can support large-scale information updating. For instance, a dismissed employee cannot access the devices anymore. Therefore, the scheme can provide efficient addition and revocation. Tian et al. [28] proposed a new SH protocol based on blind signatures, which can be used in intelligent transport systems. The scheme can provide publicly traceable property. Specifically, a user's membership can be deleted publicly, if it uses its certificate more than k times. Their scheme exhibited linkable AH property. Afterwards, they designed an unlinkable SH scheme. Tian et al. [29] constructed a novel scheme SH based on ID-based signature and ID-based encryption technology. Their scheme holds unlinkability and AH for an untrusted group authority. If a member is corrupted, it will be deleted. The malicious members are deleted by using a secret sharing algorithm. Wen et al. [30] constructed an SH scheme that considers multiple attributes. The proposed scheme can be used in multi-keyword search scenarios. Panja et al. [31] proposed an SH scheme that can provide deniability. If a user has executed the protocol, the communication manuscript can prove it, and the users cannot deny it. They designed a deniable secret handshake scheme based on blind signature technology. Chow et al. [32] designed a secret sharing scheme, and they pointed out that their scheme can be used to realize SH since the users who can recover the common secret will have the same AH information. An et al. [33] proposed a lattice-based SH scheme. In their scheme, key exposure is considered. In addition, An et al. [34] also proposed a novel lattice-based SH protocol, which is not designed without one-time certificates. Instead, the users' certificates can be reused. Based on physical unclonable functions (PUF), Qureshi and Munir [35] designed a novel authenticated key establishment scheme. Lee et al. [36] proposed an anonymous authenticated key exchange protocol based on PUF to achieve efficient user join and exit. Sun et al. [37] constructed an efficient scheme based on realistic tamper-proof devices to achieve key exchange for VANETs. Guo et al. [38] proposed an authenticated key exchange method which holds anonymity for wearable computing environments. Chen and Lee proposed an anonymous key exchange scheme which orients groups based on chaotic maps [39].

3. Building Blocks

As used in this work, we will outline the **Bilinear map** and the n-BDHE Assumption.

Bilinear map : Assume $\mathbb{G}_1$ and $\mathbb{G}_2$ are both multiplicative groups. $\mathbb{G}_1$'s generator is α. The order of $\mathbb{G}_1$ and $\mathbb{G}_2$ is some large prime q.

A bilinear map $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ can provide the properties below:

- Bilinearity: $\hat{e}(\alpha^m, \alpha^n) = \hat{e}(\alpha, \alpha)^{mn}$, for all $m, n \in \mathbb{Z}_p^*$.
- Non-degeneracy: There exist $\omega, \chi \in \mathbb{G}_1$ such that $\hat{e}(\omega, \chi) \neq 1$.
- Computability: For any $\omega, \chi \in \mathbb{G}_1$, $\hat{e}(\omega, \chi)$ can be efficiently computed.

***n*-BDHE Assumption** [40]: Let $\alpha_i = \alpha^{(t^i)} \in \mathbb{G}_1$. We say an algorithm $\mathcal{E}$ has advantage $\mathrm{Adv}(\mathcal{E})$ in solving n-BDHE problem in $\mathbb{G}_1$, where

$$\mathrm{Adv}(\mathcal{E}) = \Pr[\mathcal{E}(\alpha, \beta, \alpha^t, \ldots, \alpha^{(t^n)}, \alpha^{(t^{n+2})}, \ldots, \alpha^{(t^{2n})}) = \hat{e}(\alpha, \beta)^{t^{n+1}}].$$

The n-BDHE assumption holds, if $\mathrm{Adv}(\mathcal{E})$ is negligible for $\mathcal{E}$. Here, $\mathcal{E}$ is a probabilistic polynomial time (PPT) algorithm.

The proposed scheme is designed by using asymmetric group key agreement as given in [41] and the ring group signature technique as given in [42].

4. Models and Definitions

An organization uses the GSH protocol in scenarios where n groups $\{G_1, G_2, \ldots, G_n\}$ working within the organization. An organization authority (OA) manages the groups, while an individual group G_i is managed by a group authority GA_i. GAs can register and revoke membership. The users in group G_i are referred to as group members of G_i. Furthermore, the GSH protocol participants are referred to as players or participants. Moreover, we call the group members legitimate participants. We give the system model in Figure 1.

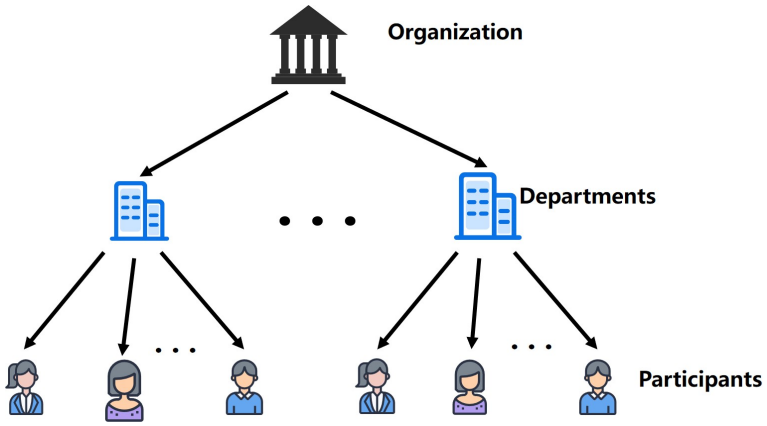

Figure 1. System model.

Definition 1. *We define the GSH protocol with the following algorithms.*

- `Setup`: *The input is a set of security parameters, and the output is public parameters.*
- `CreateOrganization`: *Given public parameters, OA outputs $\mathcal{SK}_i$, which is the group secret key, and gpk_i, which is the group public key for G_i. It also creates a certificate revocation list. The certificate revocation list is originally public and empty. $\mathcal{SK}_i$ and gpk_i are sent to GA_i through an authenticated private channel.*
- `AddUser`: *GA_i inputs $\mathcal{SK}_i$ and U, and outputs a certificate* `cert`*, then sends it to U through an authenticated private channel.*
- `Handshake`: *It is an authentication protocol, which is performed by n participants $\{U_1, \ldots, U_n\}$ where $n \geq 2$. We assume U_i is a member of* G*. Given U_i's certificate and* G*'s certificate revocation list, U_i aborts or U_i generates the encryption and decryption key pair.*
- `RevokeUser`: *GA_i revokes U by updating $\mathcal{RL}$. Only GAs can update the certificate revocation list.*

4.1. Participants and Notations

Let Π_i^t present the instance t of U_i with its partner players. sid_i^t denotes the instance Π_i^t's session identifier. sid_i^t is the concatenation of the messages sent and received by Π_i^t. pid_i^t is a set that contains all the players' identifiers corresponding to Π_i^t. In pid_i^t, according to dictionary order, the identifiers are ordered. The encryption key and decryption key generated by Π_i^t are represented as ekey_i^t and dkey_i^t. ms_i^t is the concatenation of all the

messages received and sent by Π_i^t. In ms_i^t, according to the identifiers' order, all the messages are sorted in each round and ordered by round. Moreover, we also give the definitions of the notations used in our scheme in Table 1.

Definition 2 (**Accepting**). *The instance* Π_i^t *has accepted if it has* $\mathrm{ekey}_i^t (\neq null)$, $\mathrm{dkey}_i^t (\neq null)$, pid_i^t, *and* sid_i^t.

Definition 3 (**Partnering**). *The instances* Π_i^t *and* Π_j^s *(where* $i \neq j$*) are partnered iff (a) they have accepted; (b)* $\mathrm{pid}_i^t = \mathrm{pid}_j^s$*; and (c)* $\mathrm{sid}_i^t = \mathrm{sid}_j^s$.

Definition 4. *A GSH scheme is correct if, assuming all certificates,* $\mathcal{SK}$*, and* $\mathcal{RL}$ *are generated by executing the algorithms given earlier (except* `Handshake`*). For any instance* Π_i^t *and any of* Π_i^t*'s partners* Π_j^s*, whenever* Π_i^t *has accepted for any message* $m \in \{0,1\}^\tau$*, it holds that* $D(E(m, ekey_j^s), dkey_i^t) = m$ *and* $D(E(m, ekey_i^t), dkey_j^s) = m$.

Table 1. List of Notations.

Notation	Definition
GM_i	group manager for the i-th group
gpk_i	group public key for the i-th group
H_1, H_2	cryptographic hash functions
$sig_i(m_i)$	U_i's identity based signature on m_i using its pseudo-ID ID_i
V_{RGS}	ring signature verification algorithm
σ_i	the ring signature generated by U_i
(P, Q)	encryption key
z_i	decryption key
m	a plaintext
c	the ciphertext for m
γ_i	the issuer secret key
(ξ_i, ζ_i)	the opener secret key

4.2. Privacy Model

In this work, we define the privacy model as a game. This game is between an adversary $\mathcal{A}$ and challenger $\mathcal{C}^{ah}$. $\mathcal{A}$'s goal is to get the participants' affiliation information. The adversary should be able to distinguish between two executions in order to learn the affiliations. The two executions are; (a) where $\mathcal{C}^{ah}$ normally executes the protocols as legitimate participants, and (b) where it interacts with a simulator.

During the initialization phase, the challenger creates an organization that includes m groups. Specifically, it generates the group secret keys, the group public keys, and the members' certificates for each group. Then the challenger selects corrupted players and gives their certificates to $\mathcal{A}$. Afterward, the challenger executes `RevokeUser`, i.e., prune the corrupted members and update $\mathcal{RL}$.

$\mathcal{A}$ issues a polynomial number of $\mathtt{Start}(\Pi_i^t, G)$, $\mathtt{Send}(\Pi_i^t, \Delta)$, $\mathtt{Ekey.Reveal}(\Pi_i^t)$, $\mathtt{Dkey.Reveal}(\Pi_i^t)$, and $\mathtt{Corrupt}(U_i)$ queries adaptively. The challenger uniformly chooses a bit $b \in \{0,1\}$ randomly. If b equals 1, C^{ah} replies as legitimate players, honestly. If b equals 0, C^{ah} answers the queries using the simulator. If b equals 0, C^{ah} replies to the queries as below.

- $\mathtt{Start}(\Pi_i^t)$ and $\mathtt{Send}(\Pi_i^t, \Delta)$ queries: After receiving the queries, C^{ah} replies with the information generated by the simulator. If Δ is incorrect, C^{ah} sets *reject* as *True*, then returns *null*.
- $\mathtt{Ekey.Reveal}(\Pi_i^t)$: If $reject \neq True$, output $ekey_i^t$; otherwise, return *null*.
- $\mathtt{Dkey.Reveal}(\Pi_i^t)$: If $reject \neq True$, output $dkey_i^t$; otherwise, return *null*.
- $\mathtt{Corrupt}(U_i)$: $\mathcal{C}^{ah}$ sends $\mathtt{cert}_i$ to $\mathcal{A}$ and updates the list $\mathcal{RL}$.

At last, a bit b' is returned by $\mathcal{A}$. If $b' = b$, the adversary $\mathcal{A}$ wins the game. We define $\mathcal{A}$'s advantage as

$$\mathtt{Adv}^{ah}(\mathcal{A}) = \left|2 \cdot \Pr[b = b'] - 1\right|.$$

Definition 5. *If for any PPT adversary $\mathcal{A}$, $\mathtt{Adv}^{ah}(\mathcal{A})$ is negligible, then the GSH protocol holds AH property.*

4.3. Security Model

Similar to the previous models, this game is also played between $\mathcal{A}$ (an adversary) and $\mathcal{C}$ (a challenger). In this model, $\mathcal{A}$ has complete control of the communication channel. Moreover, it can corrupt any number of players, including the ones in the test session. $\mathcal{A}$ receives the challenge and then sends start and reveal queries (except for tested instance or any instance partnered with it). We show that $\mathcal{A}$ cannot distinguish a ciphertext that is encrypted by the public key of any fresh instance from a random string. The initialization process is omitted here since it is similar to the privacy model.

$\mathcal{C}$ responds to $\mathcal{A}$'s queries as follows:

- $\mathtt{Start}(\Pi_i^t)$ and $\mathtt{Send}(\Pi_i^t, \Delta)$: Return the answer output by the instance Π_i^t. If Δ is incorrect, output *null*.
- $\mathtt{Ekey.Reveal}(\Pi_i^t)$: Output ekey_i^t.
- $\mathtt{Dkey.Reveal}(\Pi_i^t)$: Output dkey_i^t.
- $\mathtt{Test}(\Pi_i^t)$: The query can be performed only once. Note that Π_i^t should be fresh. $\mathcal{A}$ selects (m_0, m_1) where $(|m_0| = |m_1|)$, and sends (m_0, m_1) to the challenger $\mathcal{C}$. Then $\mathcal{C}$ picks randomly $b \in \{0, 1\}$ uniformly, encrypts m_b using ekey_i^t, and sends $\mathcal{A}$ the ciphertext.
- $\mathtt{Corrupt}(U_i)$: $\mathcal{C}$ updates the list $\mathcal{RL}$, and sends U_i's certificate to $\mathcal{A}$. Even if $\mathcal{A}$ has queried Test(Π_j^s), it can still corrupt U_j.

At last, a bit b' is returned by $\mathcal{A}$. Here, $\mathcal{A}$ wins with an advantage as

$$\mathrm{Adv}(\mathcal{A}) = |2 \cdot \Pr[b = b'] - 1|.$$

In order to describe perfect forward security, Freshness is defined below.

Definition 6. *If $\mathcal{A}$ has not sent any of the queries, i.e., $\mathtt{Corrupt}(U_i)$, $\mathtt{Corrupt}(U_j)$, $\mathtt{Dkey.Reveal}(\Pi_i^t)$, or $\mathtt{Dkey.Reveal}(\Pi_j^s)$, where Π_i^t is partnered with Π_j^s, we say the instance Π_i^t is fresh.*

Definition 7. *If for any PPT adversary $\mathcal{A}$, $\mathrm{Adv}(\mathcal{A})$ is negligible in the above game, we say the GSH protocol is secure against semantically indistinguishable chosen plaintext attacks (IND-CPA).*

5. The Proposed Scheme

This section gives the details of the proposed scheme. Following it, a detailed security analysis and additional features will be given. Our scheme includes the `Setup`, `CreateOrganization`, `AddUser`, and `Handshake` algorithms:

- `Setup`: Choose a bilinear group pair $(\mathbb{G}_1, \mathbb{G}_2)$ of prime order p with a computable isomorphism ψ, where $g_1 = \psi(g_2)$. g_1 and g_2 are the respective generators of $\mathbb{G}_1$ and $\mathbb{G}_2$. For bilinear map $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$, the Strong Diffie–Hellman (SDH) assumption holds on $(\mathbb{G}_1, \mathbb{G}_2)$, and the Linear assumption holds on $\mathbb{G}_1$. Also select two hash functions $H_1 : \{0,1\}^* \to \mathbb{G}_1$, and $H_2 = \mathbb{G}_2 \to \{0,1\}^\tau$.
- `CreateOrganization`: Suppose a secret organization created by an organization authority (OA) includes n departments (groups), and all of them support the SDH+ group signature scheme. The i-th group is managed by a group manager GM_i. OA generates gpk, gpk_i, γ_i, and (ξ_i, ζ_i), and sends them to GM_i through a secure authenticated channel, where $\{gpk, l\} = \{gpk_i, l_i\}_{\{i \in [n]\}}$ are the lists of the n group public keys

and sizes, $gpk_i = (g_{1i}, g_{2i}, w_i, d_i, h_i, u_i, v_i)$ and its issuer secret key is γ_i and opener secret key is (ξ_i, ζ_i).

- `AddUser`: GM_i adds the j-th member to the i-th group ($(i,j) \in ([n], [l_i])$) and sends the membership secret key $msk(i,j)$, gpk, and the registration value $reg_{(i,j)}$ to it through a private authenticated channel.
  `RevokeUser`: To remove a user U_i, the GMs add ID_i into the certificate revocation list.
- `Handshake`: Executed by some set $\mathcal{U} = \{U_1, \ldots, U_{n'}\}$ of players. Let S and D be empty sets (initially) of integers. Let $W = \{1, \ldots, n'\}$. In the first round, U_i broadcasts M_i to other participants. In the second round, U_i broadcasts $(c_i, \mathrm{sid}_i, sig_i(c_i, \mathrm{sid}_i))$. The details are shown as follows:

  Round 1:

  1. U_i selects $r_i \in \mathbb{Z}_p^*$ and $T_i \in \mathbb{G}_1 \setminus 1$ randomly, then computes $P_i = g^{-r_i}$ and $Q_i = \hat{e}(T_i, g)$.
  2. For $j \in [1, n']$, U_i computes $f_j = H_1(j)$ and $z_{i,j} = T_i f_j^{r_i}$.
  3. Set $m_i = (P_i, Q_i, \{z_{i,j}\}_{j \in [1,n'], j \neq i}, \mathrm{sid}_i)$.
  4. To sign m_i with respect to gpk, U_i computes a ring group signature $\sigma_i = (\mathbf{e}_{i,0}, \ldots, \mathbf{e}_{i,n-1}, c_{i,0}, \mathbf{s}_{i,0}, \ldots, \mathbf{s}_{i,n-1})$. Moreover, U_i generates an ID-based signature $sig_i(m_i)$ on m_i using its pseudo-ID ID_i. We assume all the participants are registered with the same Private key Generator.
  5. U_i broadcasts $M_i = (m_i, ID_k, sig_i(m_i), \mathbf{e}_{i,0}, \ldots, \mathbf{e}_{i,n-1}, \mathbf{s}_{i,0}, \ldots, \mathbf{s}_{i,n-1})$. *Here $c_{i,0}$ is not included in M_i .*

  Round 2:

  1. If any two received messages include the same ID_j, U_i aborts. For $j \in [1, n']$ and $j \neq i$ there exists an invalid $\mathrm{sig}_j(m_j)$, then $W = W \setminus j$. For any $j \in W$ there exists ID_j that is on the certification revocation list, then $W = W \setminus j$. U_i calculates the encryption key (P', Q'), where $P' = \prod_{j \in W} P_j$, $Q' = \prod_{j \in W} Q_j$.
  2. U_i uses (P', Q') to encrypt $c_{i,0}$ and generates the ciphertext $c_i = (c_{1,i}, c_{2,i}, c_{3,i})$, where $t_i \leftarrow \mathbb{Z}_p, c_{1,i} = g^{t_i}, c_{2,i} = P'^{t_i}, c_{3,i} = c_{i,0} \oplus H_2(Q'^{t_i})$.
  3. Set $\mathrm{sid}_i = [M_1 || \ldots || M_{n'}]$. Broadcast $(c_i, \mathrm{sid}_i, sig_i(c_i, \mathrm{sid}_i))$.
  4. U_i computes $z_i' = T_i f_i^{r_i} \prod_{j \in W, j \neq i} z_{j,i}$, and uses z_i' to decrypt ciphers. Since $Q' = \hat{e}(z_i, g)\hat{e}(f_i, P')$, U_i can compute $c_{j,0} = c_{3,j} \oplus H_2(\hat{e}(z_i', c_{1,j})\hat{e}(f_i, c_{2,j}))$.
  5. To verify the ring group signature σ_j, a verifier runs $V_{RGS}(m_j, gpk, \sigma_j)$ to check whether $c_{j,0}' = c_{j,0}$ holds. If so, σ_j is valid. If σ_j is invalid, set $S = S \cup \{j\}$. Hence, U_i can deduce that U_j ($j \in S$) is an illegal participant.
  6. Let $S' = W \setminus S$. If $|S'| \geq 3$ then $D \subset S'$ where $|D| \geq 2$. $U_{j,j \in D}$ generates (P, Q) and computes the decryption key z_i, where

$$P = \prod_{j \in D} P_j, \quad Q = \prod_{j \in D} Q_j, and$$

$$z_i = T_i f_i^{r_i} \prod_{j \in D, j \neq i} z_{j,i} = (\prod_{j \in D} X_j) f_i^{\sum_{j \in D} r_j}.$$

  – Encryption. U_i uses (P, Q) to encrypt $m \in \{0,1\}^\tau$ and obtains the ciphertext $c = (c_1, c_2, c_3)$, where $t \leftarrow \mathbb{Z}_p, c_1 = g^t, c_2 = P^t, c_3 = m \oplus H_2(Q^t)$.
  – Decryption. U_i uses z_i to decrypt ciphers. Since $Q = \hat{e}(z_i, g)\hat{e}(f_i, P)$, U_i can compute $m = c_3 \oplus H_2(\hat{e}(z_i, c_1)\hat{e}(f_i, c_2))$. Otherwise, U_i rejects.

Remark. If gpk is leaked, the dishonest participants can verify the ring group signature. However, the private key generator can trace them. In order to make the scheme easier to understand, we omit the generation algorithm and the verification algorithm $V_{RGS}(m_j, gpk, \sigma_j)$ of the ring group signature and the ID-based signature. In order to make the scheme easy to understand, we assume that U_i uses gpk to generate the ring

group signature. In fact, if U_i plans to hide itself in part of the groups, it will choose the corresponding group public keys to generate the ring group signature.

5.1. Security Analysis

Theorem 1. *The proposed GSH scheme satisfies the AH property.*

Proof of Theorem 1. In order to show that our scheme holds an Affiliation-Hiding property, two games G0 (the real game) and G1 (a simulation) are designed.

To prove that $\mathcal{A}$ cannot distinguish between G0 and G1, G1 is defined as follows.

Simulation. C^{ah} maintains list U^{list} which is initially empty. Assume $W = \{1, \ldots, n\} \setminus \{i\}$, D is a set of integers and originally empty, and pid_i^t=$\{U_1, \ldots, U_n\}$.

- $\texttt{Start}(\Pi_i^t)$: C^{ah} generates $M_i = (m_i, ID_k, sig_i(m_i))$ by normally executing the protocol, and generates $\mathbf{e}_{i,0}, \ldots, \mathbf{e}_{i,n-1}, \mathbf{s}_{i,0}, \ldots, \mathbf{s}_{i,n-1}$ randomly.
- $\texttt{Send}(\Pi_i^t, \Delta)$: C^{ah} responds to the query as follows:
  1. If any two received messages include the same ID_j, C^{ah} aborts. For $j \in [1, n']$ and $j \neq i$, if there exists $\text{sig}_j(m_j)$ as invalid, $W = W \setminus j$. For any $j \in W$, if there exists ID_j that is on the certification revocation list, $W = W \setminus j$. C^{ah} calculates (P', Q'), where $P' = \prod_{j \in W} P_j$, $Q' = \prod_{j \in W} Q_j$.
  2. C^{ah} generates randomly $c_{i,0}$ and C^{ah} uses (P', Q') to encrypt $c_{i,0}$ and generates the ciphertext $c_i = (c_{1,i}, c_{2,i}, c_{3,i})$, where $t_i \leftarrow \mathbb{Z}_p, c_{1,i} = g^{t_i}, c_{2,i} = P'^{t_i}$, $c_{3,i} = c_{i,0} \oplus H_2(Q'^{t_i})$.
  3. Set $\text{sid}_i = [M_1 || \ldots || M_{n'}]$. Broadcast $(c_i, \text{sid}_i, sig_i(c_i, \text{sid}_i))$.
- $\texttt{Ekey.Reveal}(\Pi_i^t)$: If reject $\neq$ *True*, C^{ah} computes $P = \prod_{l \in \mathcal{D}'}^n P_l$, $Q = \prod_{l \in \mathcal{D}'}^n Q_l$, and returns (P, Q); otherwise, it returns *null*.
- $\texttt{Dkey.Reveal}(\Pi_i^t)$: If reject $\neq$ *True*, C^{ah} recovers the corresponding $z_{i,i}$ corresponding to sid_i^t from U^{list}, then returns $d_i = \prod_{l \in \mathcal{D}'}^n z_{l,i}$; otherwise, it outputs *null*.
- $\texttt{Corrupt}(\mathcal{U}_i)$: C^{ah} gives cert_i to $\mathcal{A}$. Then, C^{ah} inserts id_i to $\mathcal{RL}$.

The difference between G0 and G1 is that the ring group signature is invalid in G1. Therefore, the adversary can distinguish between G0 and G1 if it can determine that the ring group signature in G0 is valid or if the ring group signature in G1 is valid. The adversary does not have gpk, so it can not verify the signatures. Moreover, $c_{i,0}$ is encrypted by the asymmetric encryption algorithm. Therefore, the adversary cannot get all the elements of the ring group signature. Let the event $\mathcal{E}$ denote the adversary guesses the group public key gpk and decrypts the ciphertext for $c_{i,0}$ successfully. We can observe that event $\mathcal{E}$ occurs with negligible probability. Therefore, the adversary cannot distinguish between G0 and G1. That is, the proposed protocol holds affiliation hiding property. □

Theorem 2. *Suppose there is an adversary $\mathcal{A}$ who asks at most q_{H_1} H_1-queries, q_{H_2} H_2-queries, q_{s_1} Start-queries, q_{s_2} Send-queries, q_c Corrupt-queries, q_E Ekey.Reveal-queries, and q_D Dkey.Reveal-queries. Moreover, suppose that it wins the game defined in the security model with* $\text{Adv}(\mathcal{A})$. *Then there exists an algorithm to break the n-BDHE assumption with an advantage*

$$\frac{(1 - n\text{Adv}_{rgs}(\mathcal{A}))}{e(q_D + n)q_{H_2}}\texttt{Adv}(\mathcal{A}).$$

Proof of Theorem 2. The challenger $\mathcal{C}$ aims to solve the n-BDHE problem, i.e., $(\alpha, \beta, \alpha_1, \ldots, \alpha_n, \alpha_{n+2}, \ldots, \alpha_{2n})$. H_1 and H_2 are treated as random oracles. If $\mathcal{C}$ uses $\mathcal{A}$ to break the protocol with $\texttt{Adv}(\mathcal{A})$, then it can break n-BDHE assumption with $\frac{(1-n\text{Adv}_{rgs}(\mathcal{A}))}{e(q_D+n)q_{H_2}}\texttt{Adv}(\mathcal{A})$. We assume that pid_i^t=$\{U_1, \ldots, U_n\}$. We omit the details of the proof since they are similar to that of Theorem 4.1 in [7]. □

5.2. Additional Features

Besides affiliation hiding and perfect forward security, the proposed scheme also holds traceability, impersonation resistance property, and detectability. The honest participants can detect invalid players by verifying the ring group signatures. If the ring group signatures are invalid, PKG can trace the real identity of the invalid players by using its pseudo-ID. The malicious players cannot impersonate others since they cannot forge others' ring group signatures.

The proposed scheme is efficient. Suppose $|W| = k_1$, $|D| = k_2$, "P" represents Pairing, "M_1" and "M_2" represent multiplication in $\mathbb{G}_1$ and $\mathbb{G}_2$, respectively, and "E" represents exponentiation in $\mathbb{G}_1$. Then, $\mathcal{U}_i$ needs $1P + k_1E + (k_1 - 1)M_1$ to perform the first round. $\mathcal{U}_i$ requires $2k_1M_1 + 3E + 2P + (k_1 - 1)M_2$ to execute the second round. After running the protocol, $\mathcal{U}_i$ needs $(k_2 - 1)M_1 + (k_2 - 1)M_2$ to compute the encryption keys and $k_2M_1 + 1E$ to compute the decryption keys.

6. Conclusions

Group member authentication is a challenging task in group communication. We design a novel protocol with multiple groups based on ring group signatures in this work. Only a legitimate member can prove that they belong to a set of legitimate affiliations, but which affiliation they belong to is not leaked. After executing the scheme, the honest players can compute a public encryption key and its decryption key. In the proposed scheme, the honest players can find the illegitimate participants, i.e., the scheme captures detectability. We proved that the scheme exhibits affiliation-hiding and perfect forward secrecy.

Author Contributions: Conceptualization, C.X.; Methodology, D.H.; Software, Z.L. and M.W.; Validation, M.W.; Formal analysis, C.X.; Writing—original draft, D.H., M.W. and C.X.; Writing—review & editing, Z.L. and K.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research is supported by the National Natural Science Foundation of China (Grant No. 61972037, No. U1804263, No. 62172040, No. U1836212) and National Key Research and Development Program of China under the grant No. 2021YFB2701200, 2022YFB2702402.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Balfanz, D.; Durfee, G.; Shankar, N.; Smetters, D.; Staddon, J.; Wong, H.C. Secret handshakes from pairing-based key agreements. In Proceedings of the 2003 Symposium on Security and Privacy, Berkeley, CA, USA, 11–14 May 2003. IEEE: Berlin/Heidelberg, Germany, 2003; pp. 180–196.
2. Crosby, S.; Goldberg, I.; Johnson, R.; Song, D.; Wagner, D. A cryptanalysis of the high-bandwidth digital content protection system. In *ACM Workshop on Digital Rights Management, Proceedings of the ACM CCS-8 Workshop DRM 2001, Philadelphia, PA, USA, 5 November 2001*; Springer: Berlin/Heidelberg, Germany, 2001; pp. 192–200.
3. Li, S.; Ephremides, A. Anonymous routing: A cross-layer coupling between application and network layer. In Proceedings of the 2006 40th Annual Conference on Information Sciences and Systems, Princeton, NJ, USA, 22–24 March 2006; IEEE: New York, NY, USA, 2006; pp. 783–788.
4. Jarecki, S.; Kim, J.; Tsudik, G. Group secret handshakes or affiliation-hiding authenticated group key agreement. In Proceedings of the Cryptographers' Track at the RSA Conference, San Fancisco, CA, USA, 5–9 February 2007; Springer: Berlin/Heidelberg, Germany, 2006; pp. 287–308.
5. Xu, C.; Guo, H.; Li, Z.; Mu, Y. New construction of affiliation-hiding authenticated group key agreement. *Secur. Commun. Netw.* **2013**, *6*, 723–734. [CrossRef]
6. Xu, C.; Guo, H.; Li, Z.; Mu, Y. Affiliation-hiding authenticated asymmetric group key agreement based on short signature. *Comput. J.* **2014**, *57*, 1580–1590. [CrossRef]
7. Xu, C.; Li, Z.; Mu, Y.; Guo, H.; Guo, T. Affiliation-hiding authenticated asymmetric group key agreement. *Comput. J.* **2012**, *55*, 1180–1191. [CrossRef]
8. Xu, C.; Zhu, L.; Li, Z.; Wang, F. One-round affiliation-hiding authenticated asymmetric group key agreement with semi-trusted group authority. *Comput. J.* **2015**, *58*, 2509–2519. [CrossRef]

9. Castelluccia, C.; Jarecki, S.; Tsudik, G. Secret handshakes from ca-oblivious encryption. In Proceedings of the International Conference on the Theory and Application of Cryptology and Information Security, Jeju Island, Republic of Korea, 5–9 December 2004; Springer: Berlin/Heidelberg, Germany, 2004; pp. 293–307.
10. Xu, S.; Yung, M. k-anonymous secret handshakes with reusable credentials. In *Acm Conference on Computer & Communications Security, Proceedings of the 11th ACM Conference on Computer and Communications Security, Washington, DC, USA, 25–29 October 2004*; Association for Computing Machinery: New York, NY, USA; p. 158.
11. Sweeney, L. k-anonymity: A model for protecting privacy. *Int. J. Uncertain. Fuzziness-Knowl.-Based Syst.* **2002**, *10*, 557–570. [CrossRef]
12. Vergnaud, D. RSA-based secret handshakes. In Proceedings of the International Workshop on Coding and Cryptography, Bergen, Norway, 14–18 March 2005; Springer: Berlin/Heidelberg, Germany, 2005, pp. 252–274.
13. Jarecki, S.; Kim, J.; Tsudik, G. Beyond secret handshakes: Affiliation-hiding authenticated key exchange. In Proceedings of the Cryptographers' Track at the RSA Conference, San Francisco, CA, USA, 8–11 April 2008; Springer: Berlin/Heidelberg, Germany, 2008; pp. 352–369.
14. Li, N.; Du, W.; Boneh, D. Oblivious signature-based envelope. In Proceedings of the Twenty-Second Annual Symposium on Principles of Distributed Computing, Boston, MA, USA, 13–16 July 2003; pp. 182–189.
15. ElGamal, T. A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE Trans. Inf. Theory* **1985**, *31*, 469–472. [CrossRef]
16. Nasserian, S.; Tsudik, G. Revisiting oblivious signature-based envelopes. In Proceedings of the International Conference on Financial Cryptography and Data Security, Anguilla, Anguilla, 27 February–2 March 2006; Springer: Berlin/Heidelberg, Germany, 2006; pp. 221–235.
17. Zhou, L.; Susilo, W.; Mu, Y. Three-round secret handshakes based on ElGamal and DSA. In Proceedings of the International Conference on Information Security Practice and Experience, Hangzhou, China, 11–14 April 2006; Springer: Berlin/Heidelberg, Germany, 2006; pp. 332–342.
18. Barker, E. *Digital Signature Standard (DSS) [Includes Change Notice 1 from 12/30/1996]*; Digital Signature Standard (DSS): Gaithersburg, MD, USA, 1994.
19. Hoepman, J.H. Private handshakes. In Proceedings of the European Workshop on Security in Ad-hoc and Sensor Networks, Cambridge, UK, 2–3 July 2007; Springer: Berlin/Heidelberg, Germany, 2007; pp. 31–42.
20. Yamashita, N.; Tanaka, K. Secret handshake with multiple groups. In *International Workshop on Information Security Applications, Proceedings of the 7th International Workshop, WISA 2006, Jeju Island, Republic of Korea, 28–30 August 2006*; Springer: Berlin/Heidelberg, Germany, 2006; pp. 339–348.
21. Kawai, Y.; Tanno, S.; Kondo, T.; Yoneyama, K.; Ohta, K.; Kunihiro, N. Extension of secret handshake protocols with multiple groups in monotone condition. *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.* **2010**, *93*, 1122–1131. [CrossRef]
22. Ateniese, G.; Kirsch, J.; Blanton, M. Secret handshakes with dynamic and fuzzy matching. In Proceedings of the Network and Distributed System Security Symposium, NDSS 2007, San Diego, CA, USA, 28 February–2 March 2007; Volume 7, pp. 43–54.
23. Jarecki, S.; Liu, X. Unlinkable secret handshakes and key-private group key management schemes. In Proceedings of the International Conference on Applied Cryptography and Network Security, Zhuhai, China, 5–8 June 2009; Springer: Berlin/Heidelberg, Germany, 2007; pp. 270–287.
24. Sorniotti, A.; Molva, R. Secret handshakes with revocation support. In Proceedings of the International Conference on Information Security and Cryptology, Seoul, Republic of Korea, 2–4 December 2009; Springer: Berlin/Heidelberg, Germany, 2009; pp. 274–299.
25. Jarecki, S.; Liu, X. Affiliation-hiding envelope and authentication schemes with efficient support for multiple credentials. In Proceedings of the International Colloquium on Automata, Languages, and Programming, Reykjavik, Iceland, 7–11 July 2008; Springer: Berlin/Heidelberg, Germany, 2008; pp. 715–726.
26. Sorniotti, A.; Molva, R. Federated secret handshakes with support for revocation. In Proceedings of the International Conference on Information and Communications Security, Barcelona, Spain, 15–17 December 2010; Springer: Berlin/Heidelberg, Germany, 2010; pp. 218–234.
27. Zhou, Q.; Pandey, O.; Ye, F. An Approach for Multi-Level Visibility Scoping of IoT Services in Enterprise Environments. *IEEE Trans. Mob. Comput.* **2020**, *21*, 408–420. [CrossRef]
28. Tian, Y.; Li, Y.; Deng, R.H.; Li, N.; Yang, G.; Yang, Z. A new construction for linkable secret handshake. *Comput. J.* **2020**, *63*, 536–548. [CrossRef]
29. Tian, Y.; Li, Y.; Mu, Y.; Yang, G. Unlinkable and Revocable Secret Handshake. *Comput. J.* **2021**, *64*, 1303–1314. [CrossRef]
30. Wen, Y.; Zhang, F.; Wang, H.; Miao, Y.; Gong, Z. Intersection-policy private mutual authentication from authorized private set intersection. *Sci. China Inf. Sci.* **2020**, *63*, 122101. [CrossRef]
31. Panja, S.; Dutta, S.; Sakurai, K. Deniable secret handshake protocol-revisited. In Proceedings of the International Conference on Advanced Information Networking and Applications, Matsue, Japan, 27–29 March 2019; Springer: Berlin/Heidelberg, Germany, 2019; pp. 1266–1278.
32. Chow, Y.W.; Susilo, W.; Tonien, J.; Vlahu-Gjorgievska, E.; Yang, G. Cooperative secret sharing using QR codes and symmetric keys. *Symmetry* **2018**, *10*, 95. [CrossRef]

33. An, Z.; Pan, J.; Wen, Y.; Zhang, F. Forward-Secure Revocable Secret Handshakes from Lattices. In Proceedings of the International Conference on Post-Quantum Cryptography, Virtual Event, 28–30 September 2022; Springer: Berlin/Heidelberg, Germany, 2022; pp. 453–479.
34. An, Z.; Zhang, Z.; Wen, Y.; Zhang, F. Lattice-Based Secret Handshakes with Reusable Credentials. In Proceedings of the International Conference on Information and Communications Security, Chongqing, China, 19–21 November 2021; Springer: Berlin/Heidelberg, Germany; pp. 231–248.
35. Qureshi, M.A.; Munir, A. PUF-RAKE: A PUF-Based Robust and Lightweight Authentication and Key Establishment Protocol. *IEEE Trans. Dependable Secur. Comput.* **2022**, *19*, 2457–2475. [CrossRef]
36. Lee, T.; Ye, X.; Lin, S. Anonymous Dynamic Group Authenticated Key Agreements Using Physical Unclonable Functions for Internet of Medical Things. *IEEE Internet Things J.* **2022**, *9*, 15336–15348. [CrossRef]
37. Sun, M.; Guo, Y.; Zhang, D.; Jiang, M. Anonymous Authentication and Key Agreement Scheme Combining the Group Key for Vehicular Ad Hoc Networks. *Complex* **2021**, *2021*, 5526412:1–5526412:13. [CrossRef]
38. Guo, Y.; Zhang, Z.; Guo, Y. Anonymous Authenticated Key Agreement and Group Proof Protocol for Wearable Computing. *IEEE Trans. Mob. Comput.* **2022**, *21*, 2718–2731. [CrossRef]
39. Chen, M.; Lee, T. Anonymous Group-Oriented Time-Bound Key Agreement for Internet of Medical Things in Telemonitoring Using Chaotic Maps. *IEEE Internet Things J.* **2021**, *8*, 13939–13949. [CrossRef]
40. Boneh, D.; Boyen, X.; Goh, E.J. Hierarchical identity based encryption with constant size ciphertext. In Proceedings of the Annual International Conference on the Theory and Applications of Cryptographic Techniques, Aarhus, Denmark, 22–26 May 2005; Springer: Berlin/Heidelberg, Germany, 2005; pp. 440–456.
41. Wu, Q.; Mu, Y.; Susilo, W.; Qin, B.; Domingo-Ferrer, J. Asymmetric group key agreement. In Proceedings of the Annual International Conference on the Theory and Applications of Cryptographic Techniques, Cologne, Germany, 26–30 April 2009; Springer: Berlin/Heidelberg, Germany, 2009; pp. 153–170.
42. Chen, L. Ring Group Signatures. In Proceedings of the 2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications, Liverpool, UK, 25–27 June 2012; pp. 409–418. [CrossRef]