# A Group Blind Signature Scheme for Privacy Protection of Power Big Data in Smart Grid

Xiao Li, Xueqing Sun, and Fengyin Li(✉)

School of Computer Science, Qufu Normal University, Rizhao 276826, China

**Abstract.** With the continuous expansion of the scale of smart grid, a large amount of data is generated during the operation of smart grid. These massive electricity data can be analyzed in detail by using big data technology to promote the construction and development of smart grid. In addition to the massive data processing generated in smart grid, one of the key factors restricting the development of smart grid is the issue of privacy leakage. In order to solve the problem of user identity information and user electricity data privacy in smart grid, we propose a conditional anonymous group blind signature scheme in smart grid, and use homomorphic tag mechanism to verify the integrity of electricity data. From the safety analysis, the results show that our scheme is safe and effective.

**Keywords:** Smart grid · Group blind signature · Anonymous authentication · Traceability

## 1 Introduction

The development of Internet technology promotes the development of various industries toward automation and intelligence. The intelligent development of the electric power industry gives birth to the smart grid. "Smart grid" is defined as the automatic transmission network that can supervise and control each node. It can ensure the two-way flow of information and power throughout the transmission and distribution process from the power plant to the end user. At this stage, the scale of the smart grid is constantly expanding, and a large amount of data is generated during the operation of the smart grid. Through the analysis of these massive user data, so as to provide users with more intelligent and rational power services is the key to distinguishing smart grids from traditional power grids. The goal of the smart grid is to provide residents and commercial users with more reliable, efficient and controllable power services. However, there are many security and privacy issues in the transmission of user data in the smart grid, such as malicious tampering of electricity consumption data, eavesdropping on user privacy data, and so on. These privacy leakage issues have affected people's information on the smart grid and seriously hindered the development of the smart grid. Therefore, how to protect the privacy of user data in the smart grid has become a top priority.

It is necessary to propose a privacy protection scheme that guarantees the conditional anonymity of user identities in the smart grid, and the control center can obtain fine-grained data on the user's electricity consumption, and at the same time verify the

Y. Tan et al. (Eds.): DMBD 2021, CCIS 1453, pp. 23–34, 2021.
https://doi.org/10.1007/978-981-16-7476-1_3

integrity of the data. The group-blind signature technology [1–3] provides a new idea for us to realize the conditional anonymity and privacy protection of users in the smart grid. It has the characteristics of group signature and blind signature at the same time. Due to the high anonymity of the group-blind signature scheme and the traceability that can guarantee conditional anonymity, more and more new practical schemes are proposed by domestic and foreign scholars [4, 5], and they are applied in various fields to ensure security [6–9].

In this paper, we apply the group-blind signature scheme to the smart grid, and propose a group-blind signature scheme that realizes privacy protection in the smart grid, ensuring the conditional anonymity of user identity information and the privacy protection of consumer data. At the same time, we use the homomorphic tag mechanism to verify the integrity of user data and achieve fine-grained data aggregation.

## 2 Preliminaries

### 2.1 Group Blind Signature

A. Lysyanskaya and Z. Ramzan combined group signature and blind signature for the first time in 1998 to design the first group-blind signature scheme-Lys98 scheme [10], and used this scheme to construct an online and anonymous electronic Cash system. The scheme usually contains three entities, including group manager, group members, and external users. A standard group-blind signature scheme consists of the following five algorithms.

(1) Setup: A probabilistic polynomial algorithm is used to generate the group public key y and the group manager's management private key $S_{GM}$.
(2) Join: The new member interacts with the group manager, and a probabilistic polynomial algorithm generates member keys and member certificates.
(3) Sign: Group members interact with an external user, through the message m input by the external user and the signer's private key, a probabilistic polynomial algorithm generates the signature $\sigma$.
(4) Verify: Input (m, $\sigma$, y), a probabilistic polynomial algorithm to determine the correctness of the signature $\sigma$ on the message m and the group public key y.
(5) Open: A probabilistic polynomial algorithm that output identify of the signer by inputting the signature $\sigma$ and group manger's private key.

### 2.2 Schnorr Identification Protocol

Schnorr identification protocol was proposed by Claus Schnorr in [11] and its security is based on the discrete logarithm problem. We assume that Prover(P) interactive with Verifier(V) in three-rounds protocol to prove that he owns w such that $W = g^{-w} \bmod q$. The flowchart of Schnorr identification is shown in Fig. 1.

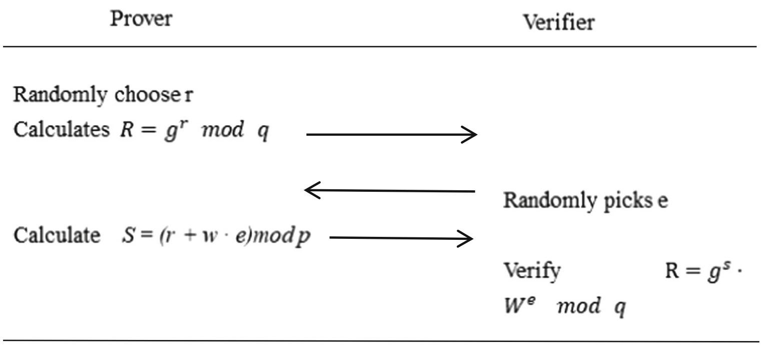
Prover
Verifier
Randomly choose r
Calculates $R = g^r \ mod \ q$
Randomly picks e
Calculate $S = (r + w \cdot e) mod p$
Verify $R = g^s \cdot W^e \ mod \ q$

**Fig. 1.** The flowchart of Schnorr identification

(1) P randomly chooses number r $\in Z_q^*$ and calculates $R = g^r \ mod \ q$ then sends R to V.

(2) V randomly picks e $\in [0, 2^t - 1]$, security of the protocol is based on the parameter t, which means the protocol will be safer with the increase of t, and send e to P.

(3) P calculates $S = (r + w \cdot e) \ mod \ p$ and sends it to V.

V will verify whether the Eq. $R = g^s \cdot W^e$ mod q is set up and accept that P knows w only if the equation holds.

## 3 System Model and Adversary Model

### 3.1 System Model

The system model of the scheme in this paper involves the working relationship of the three entities. Control Center (CC), which can generate system parameters, entity registration, data validation, and conditional tracking of other entities. Smart Substation (SS), which can directly interact with the user, verify the user's identity, and generate blind signatures. Smart meters (SM), which record data in real time, regularly send a whole period consumption data, so there is a threat of data tampering. The relationship between the three entities is shown in Fig. 2. In addition, the scheme in this paper has traceability, and CC can obtain the identity of the signer or revoke the anonymity of the user when signature verification or message validation fails.

### 3.2 Adversary Model

The adversary of the scheme in this paper can not only eavesdrop on the channel between the user and SS, but also attempt to tamper with the data and destroy the stability of the system. There are two main types of adversaries in the adversary model, one is an external adversary who is not in the data collection model, and the other is an internal adversary who has user's identity:

(1) External adversaries can obtain user consumption information by eavesdropping on the channel between SM and SS. The malicious forgery and replacement of consumer information by the adversary will threaten the integrity of the data.
(2) There are two types of internal adversaries. One is honest but curious which they want to get the consumption information of other users, but they don't want to change any data. The other is malicious users who try to tamper with their consumption data.

## 4 Our Scheme

The scheme proposed in this paper includes five stages: (1) system initialization, (2) user anonymous authentication and data reporting, (3) message signatures, (4) verifying correctness of signature and integrity of data, (5) trace the signer or users. We plan to use $SM_i/U_i$ to distinguish the different user and use the $S_i$ to indicate the number of SS. In the phase of data reporting, for simplicity, we simulate only one case where the user reports consumption data.

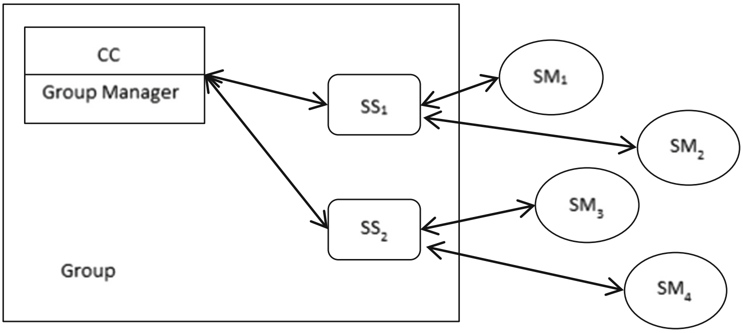

**Fig. 2.** Relationship between the three entities

### 4.1 System Initialization

(1) System parameter generation and releasing:

Step 1: CC chooses two big distinct prime p and q which satisfying $p|q-1$ and computes $n = pq$.
Step 2: CC calculates the RSA public key pair (e, d) which satisfies $ed \equiv 1(\bmod \varphi(n))$, where $\varphi(n)$ is Euler function. The group public key and group private key are e and d respectively.
Step 3: CC chooses a cyclic group $G < g >$ which is subgroup of $Z_q^*$. While, CC randomly chooses the element x and calculates $y = g^x \bmod n$. Hence, group manager's public key and private key are y and x respectively.

Step 4: CC publicly chooses secure anti-collision hash function $H:\{0,1\}^* \to \{0,1\}^k$,$H_1:\{0,1\}^* \to Z_q^*$.

Step 5: CC releases the public parameter P = {n, e, G, g, y, H, $H_1$}.

(2) The phase of registering:

Step 1: If new group member (SS) wants to join the group, SS randomly chooses the number $x_i \in Z_q^*$ and sends it to the group manager (CC). CC randomly chooses $y_i \in Z_q^*$ and calculates $C = y^{y_i}x_i \bmod n$, and $C_1 = g^{y_i} \bmod n$. The group member signed private key is $y_i$, the group member signed private key is $C_1$, Group manager Storage Group Member Certificate $x_i$, and returns PK = ($C$,$C_1$) group member's certificate $x_i$ and $y_i$ to SS.

Step 2: $User_i$ opens an account in CC and gets $infor_i = (ID_i||address||timestack)$, CC encrypts the user information $infor_i$ to $(infor_i)^e$ with the group public key e and stores it in its own database, $User_i$ will hold the public value $gt_i = (H(infor_i)^x) \bmod n$. CC installs smart meter at user's home. $SM_i$ randomly chooses $z_i$ to calculate $I_i = g^{z_i} \bmod n$ as his own id information and sends $I_i$ to CC, and CC sends the value of $I_i$ to SS.

## 4.2 User Anonymous Authentication and Data Reporting

(1) User anonymous authentication:

Each SM tries to convince SS that he is valid user by using Schnorr identification protocol and then send the encrypted message to SS. The detailed process is shown in the Fig. 3.

Step 1: $SM_i$ random chooses $t_i \in Z_q^*$ and calculates $T = g^{t_i} \bmod n$ and sends to SS.

Step 2: SS calculates $c_b = H(T||timestack)$ and sends $c_b$ to user.

Step 3: User calculates $s_i = t_i - c_b z_i$ and sends $S_i$ to SS.

Step 4: SS verifies the $c_b = H(g^{S_i} I_i^{c_b}||timestack)$.

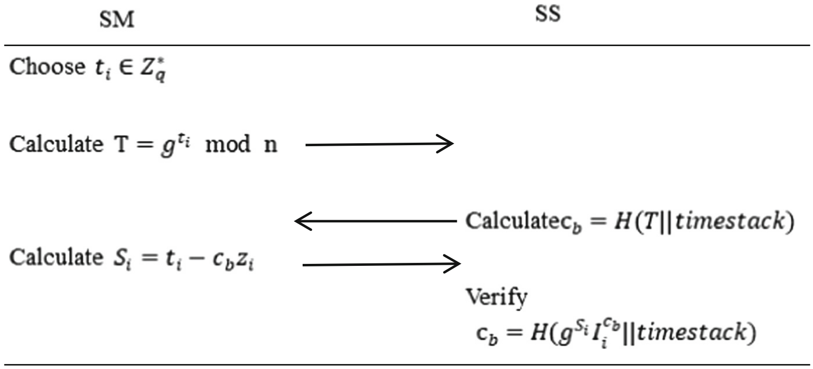
SM
SS
Choose $t_i \in Z_q^*$
Calculate $T = g^{t_i}$ mod n
Calculate$c_b = H(T||timestack)$
Calculate $S_i = t_i - c_b z_i$
Verify
$c_b = H(g^{S_i} I_i^{c_b}||timestack)$

**Fig. 3.** User anonymous authentication

(2) Data reporting:
SS will receive the user's encrypted data after verifying the meter. Take a user $\mathrm{User_k}$ as an example, the electricity bill data generated by one day is m.

Step 1: The number of blocks of consumption data reported per day is limited by the security parameter λ. Set the safety parameter λ to 24, SM should generate 24 data blocks a day. Structure of 24 blocks generated by $\mathrm{SM_k}$ shows in Fig. 4. Each block represents an hour of electricity consumption data, containing attribute values for the l dimension.
Step 2: SM generates another random number $\mathrm{stk} \in \mathrm{Z}_q^*$ as the secret tag key. Then SM outputs the public tag key $ptk = gt_k^{stk} \ mod \ n.$
Step 3: $\mathrm{SM_k}$ will generate l random values $\{\mathrm{mx_1}, \mathrm{mx_2}, \mathrm{mx3}, \ldots \ldots, \mathrm{mx_l}\}$ and calculate $\mathrm{u_j} = \mathrm{gt_k^{mx_j}}$ mod n for $\mathrm{j} \in [1, \mathrm{l}]$. For each data block $\mathrm{m_i}$, it computes a data $\mathrm{tag_i}$. $\mathrm{SM_k}$ will generate tag for every data block (24/λ/day) by calculating the $\mathrm{tag_i} = (\mathrm{H(MID}\|\mathrm{i}) \cdot \prod_{\mathrm{j}=1}^{\mathrm{l}} \mathrm{u_j^{m_{ij}}})^{\mathrm{stk}}$, in which MID is the abstract of data and i is the block number of $\mathrm{m_i}$, $\mathrm{SM_k}$ outputs the set of data tags Tag = $\{\mathrm{tag_1}, \mathrm{tag_2}, \mathrm{tag_3}, \ldots \ldots, \mathrm{tag_i}\}$, $\mathrm{i} \in [1, 24/\lambda]$.
Step 4: SM encrypts (m‖tag) by using public key e. By doing so, we ensure that no other entity can learn the consumption information, other than group private key owner CC. Then $\mathrm{SM_k}$ calculates M = $(\mathrm{m}\|\mathrm{tag})^{\mathrm{e}}$ and $\mathrm{H_1(m)}$.

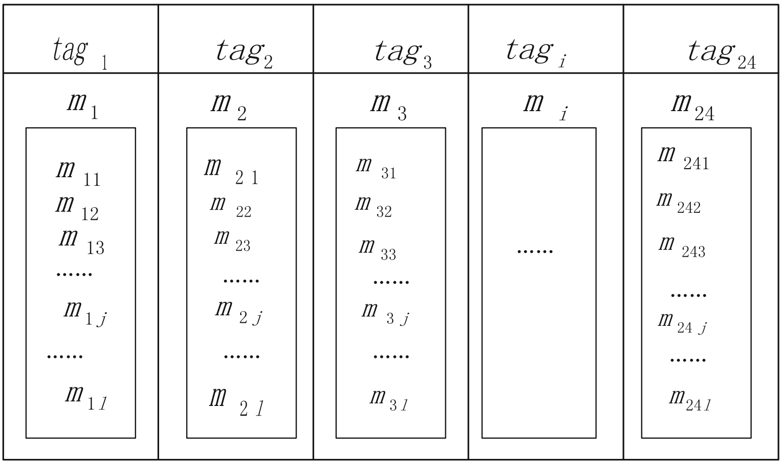

**Fig. 4.** Structure of 24 blocks generated by $\mathrm{SM_k}$

### 4.3 Blindly Signature on the Message

Generate the signature: The process is shown in the Fig. 5.

Step 1: Signer randomly chooses k $\in Z_p^*$, calculates r' = $g^k$ mod n, and sends the r' to $SM_k$, $SM_k$ randomly chooses a, b and calculates the blind factor r = $r'^a g^b$ mod n = $g^{ak}$+b mod n. Then calculates $H_1(m)' = a^{-1} r H_1(m) \bmod n - 1$, sending the $H_1(m)$ to SS.
Step 2: Signers SS to calculate the signature $\sigma^* = (r, s^*, C, C_1)$ for blind messages, where $s^* = H_1(m)' y_i + k \bmod n - 1$, $r = r'^a g^b \bmod n = g^{ak+b} \bmod n$, $C = y^{y_i} x_i$, $C_1 = g^{y_i}$.
Step 3: The $SM_k$ extracts the $H_1(m)$'s signature from the signature of the blinded message, then gets the signature σ = (r, s, $C_1$), and sends the M and the σ together to the CC by calculating the $s = as^* + b \bmod n - 1$.

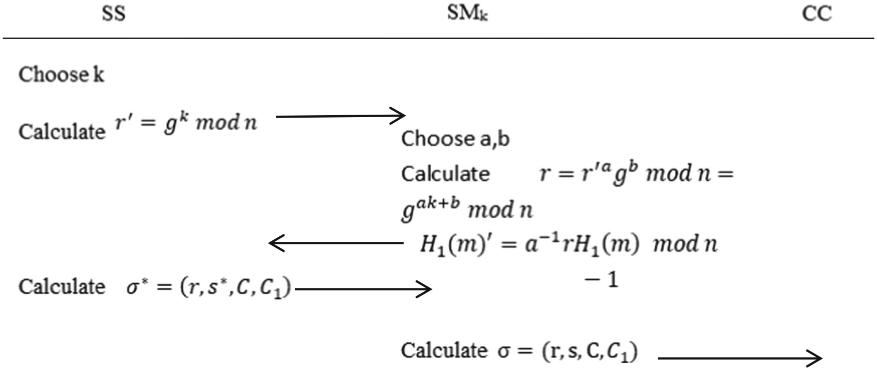
SS
SMk
CC
Choose k
Calculate $r' = g^k \bmod n$
Choose a,b
Calculate $r = r'^a g^b \bmod n = g^{ak+b} \bmod n$
$H_1(m)' = a^{-1} r H_1(m) \bmod n - 1$
Calculate $\sigma^* = (r, s^*, C, C_1)$
Calculate $\sigma = (r, s, C, C_1)$

**Fig. 5.** Generate the signature

## 4.4 Verification and Traceability

(1) Verify the signature's correctness and the data's integrity:

Step 1: CC decrypts M by using group private key d and gets the consumption information (m‖Tag) and computes $H_1(m)$.

Step 2: CC verifies the correctness of the signature by judging whether or not the Eq. (1) is established. If the signature is verified correctly, it is proved that m was not tampered with during the transmission after being signed by SS.

$$g^s = r C_1^{r H_1(m)} \tag{1}$$

Step 3. If the signature is valid, verify M by using the Tag to test whether the data has been modified. CC decrypts M to get Tag, m, $u_j$, so CC can calculate the following values:

$$TG = \prod_{i=1}^{24/\lambda} tag_i \tag{2}$$

$$MG_j = \prod_{i=1}^{24/\lambda} m_{ij} \tag{3}$$

$$DG = \prod_{j=1}^{l} e\left(u_j, ptk\right)^{MG_j} \tag{4}$$

$$HS = \prod_{i=1}^{24/\lambda} h(MID \parallel i) \tag{5}$$

Step 4: The CC then verifies whether Eq. (6) is set up every 24 h:

$$DG \cdot e(HS, ptk) = e(TG, gt_k) \tag{6}$$

Step 5: If Eq. (6) is setting up, we can sure that the consumer's information has not been modified. If not, CC will revoke the anonymity of user to check whether user or external adversary have changed the consumer's information.

(2) Trace the identify of signer and revoke the anonymity of user:

Step 1: If we find that there is controversy when verifying the signature equation, CC can open the signature to verify signer's identity $x_i$ and find the information of SS by using the CC's private key.

$$x_i = C/C_1^x = y^{y_i} x_i / g^{y_i x} \tag{7}$$

Step 2: If Eq. (6) isn't established, we know that the integrity of m has been destroyed, the anonymity of user will be revoked by CC to check whether adversary or user have changed data. Due to different SM has different gt, when CC acquires m‖Tag and corresponding gt. CC uses the group private key d to decrypt $(\mathrm{infor_i})^e$ in the database to obtain $\mathrm{infor_i}$, and uses the decrypted information to calculate $\mathrm{gt_i}$.

$$gt_i = H(infor_i)^x \ mod\ n = H(ID_i||address||timestack)^x \ mod\ n$$

Then, compare $\mathrm{gt_i}$ with the original gt to ensure the identity of user.

## 5 Security Analysis

The security of the proposed scheme is based on the assumption of several difficult problems, including the discrete logarithm problem and the integer decomposition problem. In addition, this scheme is based on Schnorr's identification protocol and the security of RSA encryption. The following is to prove that the proposed scheme has authentication, privacy-preserving, traceability, unforgeability and anonymity. The specific analysis is as follows:

### 5.1 Authenticatability

Authenticatability means that only legal users can upload their consumption information to SS. In our data reporting protocol, SS will verify the validity of consumers' identity. Only verifying successfully, SS can give a blind signature to data and send encrypted data with signature to CC. In the authenticated process, we use Schnorr identification protocol to authenticate the user's identity.

**Theorem 1.** The Schnorr identity protocol is an interactive protocol with the participation of certifier A and honest verifier B. If A and B successfully run the protocol, B is always convinced of A's identity.

Proof.

$$c_b = H(T||timestack)$$

$$= H(g^{t_i}||timestack)$$

$$= H(g^{S_i + c_b z_i}||timestack)$$

$$= Hg^{S_i} I_i^{c_b}||timestack)$$

SS can be calculated $g^{S_i} I_i^{c_b}$ and compared with T. Therefore, SS will accept SM's proof of identity as long as SS and SM can follow the protocol.

## 5.2 Privacy Protection

**Theorem 2.** The adversary cannot obtain the consumption information of the user in the initial and intermediate stages.

Proof. In the two phrases of data reporting and blind signature, the adversary and SS have the ability to obtain the encrypted user's consumption information M, but cannot directly obtain the private key of the CC. So the possible way is to divide the large prime number into p and q, assuming that factoring N into the correct p and q has a non-negligible possibility $\epsilon$ in the polynomial time algorithm. However, there is no effective algorithm to solve the problem of prime number factorization. Therefore, our scheme can effectively protect the privacy of user's consumption information.

## 5.3 Anonymity

If M is compromised, the scheme is anonymous and the adversary cannot obtain the identity of the owner of the information.

**Theorem 3.** Even if the adversary can crawl into the private database of CC and steal the decrypted information m and Tag, A can't infer the identity of the user by analysing the consumption information m.

Proof. If Adversary tries to infer the identify of data owner, the only way is to get $\mathrm{gt_k}$ from $\mathrm{tag_i} = \left(\mathrm{H(MID}||\mathrm{i}) \cdot \prod_{\mathrm{j}=1}^{\mathrm{l}} \mathrm{u_j^{m_{ij}}}\right)^{\mathrm{stk}}$ and compare to $\mathrm{H_1(infor_i)^x \bmod n}$. However, Adversary has no capacity for getting $\mathrm{gt_k}$ because of solving the discrete logarithm problem is hard. Adversary cannot link m to user identity information, so anonymity is guaranteed.

## 5.4 Unforgeability

Unforgeability refer to the fact that external adversary can't forge or tamper with the file. In our scheme of data reporting, the meter will set security parameter λ in advance to control the times of reporting. If the security parameter is λ, the frequency of sending report is 24h/λ. At the same time, we introduce the homomorphic tag mechanism to verify whether the original data has been modified.

**Theorem 4.** If our scheme has been correctly performed by all entities, the Eq. (6) will hold when the CC executes the verification.

Proof. The correctness of our verification Eq. (6) is elaborated as follows:

$$Left = DG \cdot e(HS, ptk) = \prod\nolimits_{j=1}^{l} e\left(u_j, ptk\right)^{MG_j} \cdot e(HS, ptk)$$

$$\begin{aligned} Right &= e(TG, gt_k) = e\left(\prod\nolimits_{i=1}^{24/\lambda} tag_i, gt_k\right) \\ &= e\Bigg(\Big(\prod\nolimits_{i=1}^{24/\lambda} (\mathcal{H}(MID||i) \cdot \prod\nolimits_{j=1}^{l} u_j^{m_{ij}}\Big)^{stk}, gt_k\Bigg) \\ &= e\Bigg(\Big(\prod\nolimits_{i=1}^{24/\lambda} (\mathcal{H}(MID||i)\Big)^{stk}, gt_k\Bigg) \cdot e\Bigg(\prod\nolimits_{j=1}^{l} \prod\nolimits_{i=1}^{24/\lambda} u_j^{m_{ij}stk}, gt_k\Bigg) \end{aligned}$$

$$= e\left(HS, gt_k^{stk}\right) \cdot e(\prod_{j=1}^{l} u_j^{\sum_{i=1}^{24/\lambda} m_{ij}}, gt_k^{stk})$$

$$= e\left(HS, gt_k^{stk}\right) \cdot e(\prod_{j=1}^{l} u_j, gt_k^{stk})^{\sum_{i=1}^{24/\lambda} m_{ij}}$$

$$\to \prod\nolimits_{j=1}^{l} e\left(u_j, ptk\right)^{MG_j} = e(\prod\nolimits_{j=1}^{l} u_j^{\sum_{i=1}^{24/\lambda} m_{ij}}, gt_k^{stk})$$

$$= \prod_{j=1}^{l} e(u_j^{\sum_{i=1}^{24/\lambda} m_{ij}}, gt_k^{stk})$$

$$= \prod_{j=1}^{l} e(u_j, gt_k^{stk})^{\sum_{i=1}^{24/\lambda} m_{ij}}$$

$$= e(\prod_{j=1}^{l} u_j, gt_k^{stk})^{\sum_{i=1}^{24/\lambda} m_{ij}}$$

$$\text{Left} = \text{Right}$$

Hence, we ensure that the equation DG•e(HS, ptk) = e(TG, $gt_k$) will establish through the formula if all participants follow as our scheme.

### 5.5 Traceability

**Theorem 5.** If the Eq. (1) is not established, the CC executes tracking operation to get the signer's information by using $x_i = C/C_1^x$. Next, CC will revoke the anonymity of user, if the Eq. (6) is not established.

Proof. Signature correctness:

$$g^s = g^{as^* + b}$$

$$= g^{a\left(y_i H_1(m)' + k\right) + b}$$

$$= g^{a\left(y_i a^{-1} r H_1(m) + k\right) + b}$$

$$= g^{y_i r H_1(m)} g^{ak} g^b$$

$$= C_1^{rH_1(m)} r'^a g^b$$

$$= rC_1^{rH_1(m)}$$

By verifying the correctness of Eq. (1), CC can check the signature issuing from SS.

Tracking the Signer identity:

$$x_i = C/C_1^x$$

$$= y^{y_i} x_i / g^{y_i x}$$

$$= x_i$$

CC can get identity of signer by using group private key x which only group manager owns.

$$gt_i = H(infor_i)^x \ mod\ n = H(ID_i||address||timestack)^x \ mod\ n$$

Finally, using the registration information in the database, CC can decrypt the $(infor_i)^e$ post information stored in its own database with its own private key d to get the user information $infor_i$, and then calculate the $H(infor_i)^x$ one by one to match the results with the corresponding $gt_k$.

## 6 Conclusion

This paper proposes a group-blind signature scheme for privacy protection in smart grids. The four stages of the scheme realize the conditional anonymity of user identity information and the privacy protection of consumption data in the process of collecting electricity data. In addition, the homomorphic verifiable tag mechanism is used to ensure the verifiability of the integrity of the consumption data. The subsequent security analysis partly proves that the scheme proposed in this paper has authenticatability, privacy protection, anonymity, unforgeability and traceability.

## References

1. Ramzan, Z.A.: Group blind digital signatures: theory and applications. Dissertations Massachusetts Institute of Technology (1999)
2. Mala, H., Nezhadansari, N.: New blind signature schemes based on the (elliptic Curve) discrete logarithm problem. In: ICCKE 2013, October 2013. https://doi.org/10.1109/iccke.2013.6682844
3. Khater, M.M., et al.: Blind signature schemes based on elgamal signature for electronic voting: a survey. Int. J. Comput. Appl. **975**, 8887
4. Zhao, C., Huifang, Y., Li, J.: Research on the universal composability of group-blind signatures. Appl. Res. Comput. **34**(10), 3109–3111 (2017)
5. Kong, W., et al.: A practical group blind signature scheme for privacy protection in smart grid. J. Parallel Distrib. Comput. **136**, 29–39 (2020)
6. Zhang, J., et al.: Multi-authorization electronic voting system based on group-blind signature. Favorites **8** (2015)
7. Zhang, X., Zhang, J.-Z., Xie, S.-C.: A secure quantum voting scheme based on quantum group blind signature. Int. J. Theor. Phys. **59**(3), 719–729 (2020)
8. Zhang, P., et al.: A new post-quantum blind signature from lattice assumptions. IEEE Access **6**, 27251–27258 (2018)
9. Liu, G., et al.: A novel quantum group proxy blind signature scheme based on five-qubit entangled state. Int. J. Theor. Phys. **58**(6), 1999–2008 (2019)
10. Lysyanskaya, A., Ramzan, Z.: Group blind digital signatures: a scalable solution to electronic cash. Lect. Notes Comput. Sci. 184–197 (1998). https://doi.org/10.1007/bfb0055483
11. Schnorr, C.P.: Efficient identification and signatures for smart cards. In: Brassard, G. (ed.) CRYPTO 1989. LNCS, vol. 435, pp. 239–252. Springer, New York (1990). https://doi.org/10.1007/0-387-34805-0_22