Int. J. Appl. Math. Comput. Sci., 2021, Vol. 31, No. 2, 353–364
DOI: 10.34768/amcs-2021-0024

A DECENTRALIZED GROUP SIGNATURE SCHEME FOR PRIVACY PROTECTION IN A BLOCKCHAIN

S. DEVIDAS [a,*], SUBBA RAO Y.V. [a], N. RUKMA REKHA [a]

[a]School of Computer and Information Sciences
University of Hyderabad
Hyderabad, Telangana, India
e-mail: {devidas13,yvsrcs,rukmarekha}@uohyd.ac.in

Group signature schemes play a vital role in protecting identity privacy of a member of a group who signs a message using the group signature. However, in the existing group signature schemes the centralized group manager has control over all the participants, and these managers can be malicious. They may take a biased decision when there is a dispute among the group members or while revealing the identity of a group member. To overcome the trust issues related to centralized group managers and to improve user privacy, a decentralized group signature scheme (DGSS) is proposed by decentralizing the role of the group manager. The proposed scheme will be more suitable for decentralized environments like a blockchain. Security analysis along with the proof of correctness is also provided for the proposed scheme. A framework for a blockchain-based e-auction protocol using the DGSS is also proposed in this paper.

Keywords: blockchain, decentralization, e-auction, group signature, privacy, smart contract.

1. Introduction

Nakamoto and Bitcoin (2008) pointed out that although today's commerce on the Internet is totally relied on a trusted third party and is working well enough, it suffers from the inherent problems of trust-based models. The authors proposed a peer-to-peer electronic cash system by the name of bitcoin, which allows online payments to be sent from one party to another without the need for any trusted party. Today, the decentralized public ledger technology in peer-to-peer networks (Li *et al.*, 2019; Kobusińska *et al.*, 2016) is becoming popular and is called the blockchain technology. It has received considerable attention recently with the continuous development in financial and non-financial domains (Fernandez-Vazquez *et al.*, 2019) and its security features. This led to a flurry of advancements in various applications using blockchains. The blockchain offers various security features such as transparency, immutability, or traceability in business transactions (Al Jawaheri *et al.*, 2020). Although all blockchain systems possess these security features, few of business applications like e-auction, crowd funding, etc., emphasize user privacy.

*Corresponding author

Blockchains are majorly categorized (Feng *et al.*, 2019) as (i) public blockchains, which are open to anyone with read and write permissions and any participant can join the consensus process for decision making; (ii) consortium blockchains, where more than one organization can come together to form a network, and read permission is open to all within the network but certain constraints are placed on write permissions where only identified participants from different organizations can participate in the consensus process; (iii) private blockchains, where only one organization can form a network, and read permission is open to all within the network or organization but write permissions are restricted to only identified members of that organization who can participate in the consensus process.

Transactions play a vital role in any blockchain system (Zheng *et al.*, 2020). They are first created by various users and broadcast on the network, and then validated by the network, and subsequently all such validated transactions form a block to be finally added into the blockchain. The transaction data structure can encode the transfer of value from one party to another in the system, and every transaction is a public entry (Androulaki *et al.*, 2013) in a publicly available global ledger blockchain. In order to transact on a blockchain, the user requires a public-private key pair where the public key is used for account identification and the private key to sign the transaction. Since every transaction in the blockchain network is publicly available, anyone in the network can inspect and analyse it.

In the recent past, numerous papers have been published on the blockchain, and several have focused on security and privacy. Kong *et al*. (2018) proposed a personal identification system based on a brain network of EEG signals. Bonneau *et al*. (2015) analyzed the anonymity problem and reviewed privacy enhancing methods. Karame (2016) analyzed the risks and possible attacks in bitcoin systems. The author even proposed a few mitigation strategies to address the issues. Conti *et al*. (2018) reviewed the existing security loopholes in bitcoin. Li *et al*. (2017) surveyed the security risks, possible attacks and vulnerabilities that can be exploited for various blockchain systems. There are studies showing the possibilities to de-anonymize users by creating a transaction graph for various transactions done on a blockchain (Reid and Harrigan, 2013; Ron and Shamir, 2013). The emerging blockchain technology has been solving many problems and entering into almost all the emerging fields (Gao *et al*., 2018), and one such a field is e-auctions.

Over the years, several kinds of auctions have been invented (Krishna, 2009). The most well-known is the English auction, in which the buyer who offers the highest price will win the auction. Eventually, auctions began to take place online, as e-auctions. The e-auctions market is huge, as demonstrated by websites like eBay, which had more than 170 million active buyers in 2018.[1] A buyer who wants to purchase goods or services from sellers need to submit bids. The buyer wants the lowest price, while each seller hopes to get competitive prices from the buyers. To facilitate this mechanism, a trusted third party is required to host the auction and to achieve the privacy of the participants and fairness exchange. But the trusted third party holds a lot of important information about the users. So, this may lead to potential threats (Jouini *et al*., 2014) from single-point attacks to collusion attacks all the time, and it is also difficult to find a fully trusted third party to play such a role in reality.

Recently, many auction protocols have been deployed on top of the blockchain to take advantage of the decentralization, transparency, immutability and verifiability properties of the blockchain, and to get rid of the above shortcomings that were brought by the third party (Chen *et al*., 2018). But as every transaction in the blockchain network is publicly available, transaction analysis can reveal the original identity of the user and their monetary values. The blockchain can be very much regarded as a trusted party for correctness and availability but not for privacy. Considering the public nature of transactions in the network and the privacy challenge of the identity of a user in blockchains, we propose a novel decentralized group signature scheme (DGSS) that can be utilized in blockchains.

The rest of this paper is organised as follows. Section 2 describes the related work. The correctness and security definitions are discussed in Section 3. We analyse proposed decentralized group signature scheme in Section 4. In Section 5 a numerical example of the proposed DGSS is given. In Section 6 the proof of correctness and security analysis of the DGSS is discussed. Section 7 demonstrates a blockchain-based e-auction protocol using DGSS. Finally, Section 8 concludes the paper.

2. Related work

The digital signature is a cryptographic technique used to verify the authenticity of a message and the identity of the sender. A valid signature ensures the integrity of the message as well as non-repudiation. The idea of digital signatures was extended for groups by Chaum and Van Heyst (1991), who first grouped the signature scheme. Any member of the group can anonymously sign a message using group signature without revealing his/her real identity, and the identity of the signer can be revealed by a designated manager of the group. Another group signature scheme that allows members to join and leave the group dynamically was proposed by Chen and Pedersen (1994). Later, Kim *et al*. (2000) proposed a group signature scheme that allows revoking group members efficiently. Some open challenges and new research directions in the group signature scheme were discussed by Ateniese *et al*. (2002), like coalition attacks and deleting group members .

Lee *et al*. (2009) proposed a new group signature scheme that can achieve authenticity, integrity and non-repudiation with confidentiality by using authenticated encryption. Using this new group signature scheme, they designed a sealed-bid auction protocol where confidentiality of the bids is maintained till the bids are opened. Sun *et al*. (2013) proposed another group signature scheme by adding one more random number to Lee *et al.* (2009) group signature to improve security weaknesses. Tsai *et al*. (2018) claimed that their group signature scheme is based on the discrete logarithm problem that addresses security and efficiency concerns.

The origin of the blockchain has begun with removal of the trusted third-party and further bringing trust among the untrusted group (Wang *et al*., 2019). In the literature all researchers have focused on various security issues

[1]https://in.ebay.com/.

of group signature, schemes and their designated group manager has always remained a trusted third party only. In this paper, as our first contribution, we decentralize the role of the designated group manager of a static group signature scheme to address the trust issues of centralized group manager.

Recently, many researchers have been focusing on integrating blockchains with e-auction. Kosba *et al*. (2016) presents Hawk, a framework for creating an Ethereum smart contract on the blockchain. Hawk utilizes a zero knowledge proof (ZKP) to prove the honesty of the manager, but it will take a long time to produce the proof and deploying the ZKP in a smart contract is complex (Zhang *et al*., 2019). Blass and Kerschbaum (2018) present the Strain protocol to implement sealed-bid auction on the blockchain that protects bid privacy against fully malicious parties. But the protocol requires multiple interactions between each participant, and the communication and computation overheads are very large for individual users. Sánchez (2018) proposed Raziel, a system that combines multi-party computation (MPC) and ZKP cryptographic primitives to guarantee the privacy, correctness and verifiability of smart contract. Galal and Youssef (2018) presented a protocol for running sealed-bid auctions on Ethereum. This protocol ensures public verifiability, privacy of bids, and fairness. Lafourcade *et al*. (2019) showed that a bidder's privacy in a blockchain-based e-auction protocol is still a big challenge, because every transaction within the blockchain system is open to all and can be inspected and analysed to link real identities. In this paper, as our second contribution, we utilize our proposed decentralized group signature scheme (DGSS) to enhance the bidders' privacy in a private blockchain-based e-auction protocol.

3. Correctness and security definitions

In this paper, we adopt the definition of group signature schemes and security definitions from the work of Bellare *et al*. (2003).

Definition 1. (*Decentralized group signature scheme*) A decentralized group signature scheme $DGSS = (Init, Sign, Verify, Identify)$ is a collection of four polynomial-time algorithms defined as follows:

Init: The initiation algorithm $Init$ takes the secret key of group managers x_j, random integer k_{ij} in Z_q^* chosen by group managers, public key y_i $(1 \leq i \leq m)$ of the group members as an input and returns (r_{ij}, sij) $(1 \leq j \leq n)$.

Sign: The signing algorithm $Sign$ takes message $M_{original}$, attachment M_{check}, two random integers N_1, N_2 in Z_q^* as input and returns group signature DGSS.

Verify: The verification algorithm $Verify$ takes group signature $\{A, B, C, D, M_{check}\}$, private key of the receiver x_l, collision-resistant hash function $h(\cdot)$ as input and returns message M.

Identify: The identifying algorithm $Identify$ takes public keys of the group members $y_i (1 \leq i \leq n)$, random integers $k_{ij} (1 \leq j \leq m)$ of each group member, parameter B as input and returns the public key y_i of the actual signer.

Definition 2. (*Correctness*) The signature produced by the honest group member should always be accepted, i.e., the $Verify(\cdot)$ algorithm should return 1. The $Identify(\cdot)$ algorithm should always identify the actual signer of the message for the given valid message and group signature.

Definition 3. (*Unforgeability*) It is computationally difficult for any unauthorized member to produce a valid signature on behalf of the group. Only an authorized member of the group can produce a valid signature on behalf of the group.

Definition 4. (*Anonymity*) It is computationally difficult for anyone to determine the actual signer of the message for a given valid group signature.

Definition 5. (*Unlinkability*) It is computationally difficult for anyone to determine whether or not the two valid group signatures are produced by the same user.

Definition 6. (*Traceability*) It is computationally difficult for anyone except group managers to track the identity of the actual singer. If there is any dispute among the group members or as per requirement, all the group managers together can identify the actual signer.

4. Proposed decentralized group signature scheme

To address the user identity privacy challenges in a blockchain, a decentralized group signature scheme (DGSS) is proposed that is based on the difficulty of the discrete logarithm problem. In the literature most of the existing group signature schemes (Agarwal and Saraswat, 2013) contain a designated group manager as a centralized party to reveal the identity of the group member upon requirement. The proposed scheme is based on the assumption that a group manager may be malicious. A malicious manager carries the risk of revealing identities and of a collusion attack. On the other hand, the origin of a blockchain has begun to bring trust among the untrusted party, where the individual party can behave maliciously but as a group they cannot. The existing group signature schemes available in the literature are not suitable to address the privacy leakage of the e-auction protocol because of the centralized group manager. In this section we decentralize the designated group manager of the discrete logarithm-based group signature scheme (Lee *et al.*, 2009) to address the privacy issue of the blockchain-based e-auction protocol. The proposed DGSS consists of four polynomial-time algorithms: the Initiation algorithm, the Signing algorithm, the Verification algorithm, and the Identification algorithm. The DGSS is described as follows.

4.1. Initiation algorithm. Let p and q be two large prime numbers such that $q|p-1$, and g be a generator with order q in $GF(p)$. Each group member U_i ($1 \le i \le m$) selects the private key x_i and computes the public key $y_i = g^{x_i} \bmod p$. Receiver l chooses his/her private key x_l randomly and computes public key $y_l = g^{x_l} \mod p$. Each group manager T_j ($1 \le j \le n$) selects his/her private key x'_j and computes the public key $y'_j = g^{x'_j} \bmod p$. For each group member U_i, each group manager T_j randomly chooses an integer k_{ij} in Z_q^* and computes

$$r_{ij} = (y_i \times k_{ij} - x'_j) \mod q, \tag{1}$$

$$s_{ij} = y_i^{k_{ij}} \mod p. \tag{2}$$

Now, each group manager T_j sends pair (r_{ij}, s_{ij}) to the group member U_i. After receiving (r_{ij}, s_{ij}) pairs from all the group managers, the group member U_i computes the certificate as follows:

$$R_i = \sum_{j=1}^{n} r_{ij}, \tag{3}$$

$$S_i = \prod_{j=1}^{n} s_{ij}. \tag{4}$$

After computing (R_i,S_i), the group member U_i can verify the validity of the certificate by checking the following equation:

$$S_i^{y_i} \mod p = (g^{R_i} \times \prod_{j=1}^{n} y'_j)^{x_i} \mod p. \tag{5}$$

The proof of validity is given in Section 6.1.

4.2. Signing algorithm. In a DGSS, a short message M_{check} is added as a test. Group member U_i generates a group signature for message $M_{original}$ by computing the following:

1. Compute $M = M_{check} \| M_{original}$, where the symbol $\|$ stands for concatenation.
2. Group member U_i selects two random numbers N_1,N_2 in Z_q^*.
3. U_i computes four parameters A, B, C, D as follows:

$$A = x_i \times N_1 \times N_2 \mod q, \tag{6}$$

$$B = S_i^{N_1 \times N_2 \times y_i} \mod p, \tag{7}$$

$$C = M \times y_l^{-N_1 \times A \times h(B)} \mod p, \tag{8}$$

$$D = N_1 - R_i \times h(C) \mod q. \tag{9}$$

4. Group signature for message M is $\{A, B, C, D, M_{check}\}$.

4.3. Verification algorithm. The receiver can now reconstruct and check the validity of message M in the following steps:

1. Reconstruction of the message M is computed as follows:

$$M = C \times \Big[g^{D \times A} \times \prod_{j=1}^{n} y_j'^{-h(C) \times A} \times B^{h(C)} \Big]^{x_l \times h(B)} \mod p. \tag{10}$$

2. Message M is valid if and only if

$$M_{check} \stackrel{?}{=} \text{head}(M, s) \tag{11}$$

where $h(\cdot)$ is a collision-resistant hash function, M_{check} is a binary string with s bits, and head(M, s) is a function which returns the first s bits of binary string M. The signature is valid if and only if the above equation holds. The proof of validity is given in Section 6.

4.4. Identification algorithm. When there is a dispute among the group members, the group signature must be opened to reveal the real identity of the actual signer. As group manager T_j has access to (y_i, k_j) of each group member U_i, group manager T_j acquires the (y_i, k_{ij}) of U_i and looks for the signature that satisfies the following equation:

$$B = g^{A \times \sum_{j=1}^{n} k_{ij} \times y_i} \mod p \tag{12}$$

for $i = 1, 2, 3, \ldots, n$, where n is the size of group. Thereby, the group manager can determine the signer.

5. DGSS example

In this section, a numerical example of the proposed DGSS is discussed in detail.

5.1. Initiation algorithm.

- Let $p = 227$, $q = 113$ such that $q|p-1$
- $GF(227) = \{0, 1, 2, \ldots, 226\}$ and 4 is a generator with order q.
- Group member U_1 chooses private key $x_1 = 3$ and computes public key $y_1 = 64$ (i.e., $y_i = g^{x_i}$ mod$p$).
- Each group manager T_j $(1 \leq j \leq 3)$ computes their corresponding (r_{1j}, s_{1j}) pairs as follows:

Group manager T_1:

- T_1 chooses his/her private key $x'_1 = 4$ and computes public key $y'_1 = 29$
- T_1 randomly chooses an integer $k_{11} = 3$ from z^*_{113} for group member U_1 and computes pair (r_{11}, s_{11}) as follows:

$$r_{11} = (64 \times 3 - 4) \mod 113 = 75, \quad \text{(from (1))}$$
$$s_{11} = 64^3 \mod 227 = 186. \quad \text{(from (2))}$$

- T_1 sends $(r_{11}, s_{11}) = (75, 186)$ to U_1.

Group manager T_2:

- T_2 chooses his/her private key $x'_2 = 5$ and computes public key $y'_2 = 116$.
- T_2 randomly chooses an integer $k_{12} = 7$ from z^*_{113} for group member U_1 and computes (r_{12}, s_{12}) pair as follows:

$$r_{12} = (64 \times 7 - 5) \mod 113 = 104, \quad \text{(from (1))}$$
$$s_{12} = 64^7 \mod 227 = 213, \quad \text{(from (2))}$$

- T_2 sends $(r_{12}, s_{12}) = (104, 213)$, to U_1.

Group manager T_3:

- T_3 chooses his/her private key $x'_3 = 6$ and computes public key $y'_3 = 10$.
- T_3 randomly chooses an integer $k_{13} = 8$ from z^*_{113} for group member U_1 and computes (r_{13}, s_{13}) pair as follows:

$$r_{13} = (64 \times 8 - 6) \mod 113 = 54, \quad \text{(from (1))}$$
$$s_{13} = 64^8 \mod 227 = 12. \quad \text{(from (2))}$$

- T_3 sends $(r_{13}, s_{13}) = (54, 12)$ to U_1.
- Now, group member U_1 computes his/her (R_1, S_1) pair as follows:

$$R_1 = (75 + 104 + 54) \mod 113 = 7, \quad \text{(from (3))}$$
$$S_1 = (186 \times 213 \times 12) \mod 227 = 78. \quad \text{(from (4))}$$

- U_1 verifies the correctness of his/her (R_1, S_1) pair using Eqn. (5):

$$78^{64} \mod 227 = \left[4^7(29 \times 116 \times 10)\right]^3 \mod 227,$$
$$82 = 82.$$

- Equation (5) holds. Hence, pair (R_1, S_1) is valid.

5.2. Signing algorithm.

- Group manager U_1 generates a group signature for the message $M_{Original} = 27$ by concatenating $M_{Check} = 5$ using the following steps:

1. $M = 27||5$ ($M = M_{Original}||M_{Check}$) $M = 221$.
2. U_1 selects two random integers $N_1 = 4$, $N_2 = 5$ in Z^*_{113}.
3. U_1 computes four parameters A, B, C, D as follows:

$$A = (3{\times}4{\times}5) \mod 113, \quad \text{(from (6))}$$
$$A = 60,$$

$$B = 78^{4\times5\times64} \mod 227, \quad \text{(from (7))}$$
$$B = 7,$$

$$C = 221{\times}16^{-4\times60\times h(7)} \mod 227, \quad \text{(from (8))}$$
$$C = 203,$$

$$D = 4 - 7 \times h(203) \mod 113, \quad \text{(from (9))}$$
$$D = 61.$$

Note: Let $h(7) = 3$ and $h(203) = 8$.

4. The group signature for the message 221 is $\{60, 7, 203, 61, 5\}$.

5.3. Verification algorithm.

- Receiver can now reconstruct the message using the group signature $\{60, 7, 203, 61, 5\}$ and check the validity of the message using his/her private key $x_{11} = 2$ and public $y_{11} = 16$.

  1. Reconstruction of the message is computed using Eqn. (10):

$$\begin{aligned} &203 \times \Big[4^{61\times 60} \times (29 \times 116 \times 10)^{-8\times 60} \\ &\quad \times 7^8\Big]^{2\times 4} \mod 227 \\ &= 203 \times \Big[4^{3660} \times (33640)^{-480} \\ &\quad \times 7^8\Big]^{8} \mod 227 \\ &= 203 \times \Big[4^{3660} \times (33640)^{198} \\ &\quad \times 7^8\Big]^{8} \mod 227 \\ &= 203 \times \Big[4^{44} \times (33640)^{198} \\ &\quad \times 7^8\Big]^{8} \mod 227 \\ &= 221. \end{aligned}$$

  2. The message 221 is valid if Eqn. (11) holds:

$$\begin{aligned} \mathrm{head}(221, 3) &= \mathrm{head}(11011101, 3) \\ &= 101 = 5 \end{aligned}$$

- Equation (11) holds. Hence, the message is valid.

5.4. Identification algorithm.

- All the group managers use public keys of the group members and their random integer k_{ij} to identify the actual signer of the message.
- The public key of group member U_1 is $y_1 = 64$ and random integers of all the group managers are $k_{11} = 3$, $k_{12} = 7$ and $k_{13} = 8$.
- If (12) holds then the user with public key $x_1 = 3$ is the actual signer

$$\begin{aligned} &4^{60\times((3\times 64)+(7\times 64)+(8\times 64))} \mod 227 \\ &= 4^{60\times(192+448+512)} \mod 227 \\ &= 4^{60\times 1152} \mod 227 \\ &= 4^{69120} \mod 227 \\ &= 4^{190} \mod 227 \\ &= 7. \end{aligned}$$

- Equation (12) holds. Hence the user with public key $y_1 = 64$ is the actual signer of the message.

6. Proof of correctness and security analysis

The security analysis and the proof of correctness for the proposed DGSS is discussed in this section. The DGSS is holding all the security properties of the group signature scheme even after decentralizing the group manager Lee *et al.* (2009).

6.1. Correctness for pair (R_i, S_i).

After computing pair (R_i, S_i), group member U_i can verify the validity of the certificate as follows:

$$\begin{aligned} &(g^{R_i} \times \prod_{j=1}^{n} y'_j)^{x_i} \mod p \\ &= (g^{\sum_{j=1}^{n} r_{ij}} \times \prod_{j=1}^{n} y'_j)^{x_i} \mod p \qquad \text{(from (3))} \\ &= (g^{\sum_{j=1}^{n}(y_i\times k_{ij} - x'_j)} \times \prod_{j=1}^{n} y'_j)^{x_i} \mod p \qquad \text{(from (1))} \\ &= (g^{y_i \times \sum_{j=1}^{n} k_{ij} - \sum_{j=1}^{n} x'_j} \times \prod_{j=1}^{n} g^{x'_j})^{x_i} \mod p \\ &= (g^{y_i \times \sum_{j=1}^{n} k_{ij} - \sum_{j=1}^{n} x'_j} \times g^{\sum_{j=1}^{n} x'_j})^{x_i} \mod p \\ &= g^{x_i \times y_i \times \sum_{j=1}^{n} k_{ij}} \mod p \\ &= (g^{x_i \times \sum_{j=1}^{n} k_{ij}})^{y_i} \mod p \\ &= (\prod_{j=1}^{n} g^{x_i \times k_{ij}})^{y_i} \mod p \\ &= (\prod_{j=1}^{n} y_i^{k_{ij}})^{y_i} \mod p \\ &= (\prod_{j=1}^{n} s_{ij})^{y_i} \mod p \qquad \text{(from (2))} \\ &= S^{y_i}. \end{aligned}$$

It is discussed in Section 4.3 that for the group signature on message M $=\{A, B, C, D,\ M_{check}\}$ from Eqn. (10), it can be as follows:

$$\begin{aligned} &C \times \Big[g^{D\times A} \times \prod_{j=1}^{n} y_j'^{-h(C)\times A} \\ &\times B^{h(C)}\Big]^{x_l \times h(B)} \mod p \\ &= C \times \Big[g^{(N_1 - R_i \times h(C))\times A} \times g^{-\sum_{j=1}^{n} x'_j \times h(C) \times A} \\ &\times S_i^{N_1\times N_2 \times h(C)}\Big]^{x_l\times h(B)} \mod p \qquad \text{(from (9))} \end{aligned}$$

$$= C \times \Big[g^{N_1\times A - R_i\times h(C)\times A} \times g^{-\sum_{j=1}^{n} x'_j \times h(C)\times A} \times (g^{R_i} \times \prod_{j=1}^{n} y'_j)^{x_i\times N_1\times N_2\times h(C)} \Big]^{x_l\times h(B)} \mod p$$

(from (5))

$$= C \times \Big[g^{N_1\times A - R_i\times h(C)\times A} \times g^{-\sum_{j=1}^{n} x'_j \times h(C)\times A} \times g^{R_i\times x_i\times N_1\times N_2\times h(C)} \times g^{\sum_{j=1}^{n} x'_j\times x_i\times N_1\times N_2\times h(C)} \Big]^{x_l\times h(B)} \mod p$$

$$= C \times \Big[g^{N_1\times A - R_i\times h(C)\times A} \times g^{-\sum_{j=1}^{n} x'_j \times h(C)\times A} \times g^{R_i\times A\times h(C)} \times g^{\sum_{j=1}^{n} x'_j\times A\times h(C)} \Big]^{x_l\times h(B)} \mod p \qquad \text{(from (6))}$$

$$= C \times \Big[g^{N_1\times A - R_i\times h(C)\times A - \sum_{j=1}^{n} x'_j \times h(C)\times A} \times g^{+R_i\times A\times h(C) + \sum_{j=1}^{n} x'_j\times A\times h(C)} \Big]^{x_l\times h(B)} \mod p$$

$$= M \times y_l^{-N_1\times A\times h(B)} \times g^{N_1\times A\times x_l\times h(B)} \mod p$$

(from (8))

$$= M \times g^{-N_1\times A\times h(B)\times x_l} \times g^{N_1\times A\times x_l\times h(B)} \mod p$$
$$= M \times g^{-N_1\times A\times h(B)\times x_l + N_1\times A\times x_l\times h(B)} \mod p$$
$$= M.$$

6.2. Security analysis. The security of the proposed DGSS is based on the difficulty of the discrete logarithm problem. The DGSS satisfies all the security properties as follows.

6.2.1. Unforgeability. An attacker can generate a valid group signature if and only if he have a valid (R_i, S_i) and x_i. Even with the assumption that the attacker has a valid (R_i, S_i), in order to generate a valid group signature, he or she first needs to compute the value of B by Eqn. (7), which is not feasible as N_1, N_2 are not known, and then the values of parameters A, C, D by Eqns. (6), (8) and (9). As for the proposed scheme, the attacker does not have the secret key x_i. Hence he or she can never be able to forge the group signature.

6.2.2. Anonymity. Given a valid group signature $\{A, B, C, D,\ M_{check}\}$ it is difficult for anyone except the group managers to identify the actual signer. All the private information inside the group signature is protected by random parameters. In group signature $\{A, B, C, D, M_{check}\}$, only A and B have the identity information. So, whether the scheme has anonymity by A and B or not is discussed.

Attack 1: Given a valid group signature $\{A, B, C, D, M_{check}\}$ and the equation $A = x_i \times N_1 \times N_2 \mod q$, one can compute

$$g^A = g^{x_i\times N_1\times N_2} \mod p$$
$$= y_i^{N_1\times N_2} \mod p.$$

If the attacker has N_1, N_2, then he/she can compute y_i and find the actual signer's identity. But the random integers N_1, N_2 are unknown and thus it is not feasible to find the actual signer. Therefore, the proposed DGSS has anonymity by parameter A.

Attack 2: Given a valid group signature $\{A, B, C, D, M_{check}\}$ and the equation $B = S_i^{N_1\times N_2\times y_i} \mod p$, one can compute

$$S_i^{N_1.N_2.y_i}$$
$$= (g^{R_i} \times \prod_{j=1}^{n} y'_j)^{x_i\times N_1\times N_2} \mod p$$

(from (5))

$$= (g^{\sum_{j=1}^{n} r_{ij}} \times \prod_{j=1}^{n} g^{x'_j})^{x_i\times N_1\times N_2} \mod p$$

(from (3))

$$= (g^{\sum_{j=1}^{n}(y_i\times k_{ij} - x'_j)} \times g^{\sum_{j=1}^{n} x'_j})^{x_i\times N_1\times N_2} \mod p$$

(from (1))

$$= g^{\sum_{j=1}^{n} k_{ij}\times y_i\times x_i\times N_1\times N_2} \mod p$$
$$= y_i^{\sum_{j=1}^{n} k_{ij}\times N_1\times N_2\times y_i} \mod p.$$

If the attacker has $\sum_{j=1}^{n} k_{ij}$, N_1, N_2, then he/she can compute y_i and find the actual signer's identity. But $\sum_{j=1}^{n} k_{ij}$, N_1, N_2 are unknown and hence no one can find the actual signer. Therefore, the proposed DGSS has anonymity by B. Because of anonymity of A and B, the proposed DGSS has anonymity by C and D, respectively by Eqns. (8) and (9). Hence, the entire group signature $\{A, B, C, D, M_{check}\}$ has anonymity.

6.2.3. Unlinkability.

Lemma 1. *To determine whether the two group signatures* $\{A, B, C, D,\ M_{check}\}$ *and*

$\{A', B', C', D', M'_{check}\}$ are generated by the same user, the following equation should hold:

$$\frac{B}{B'} = \left(\frac{g^A}{g^{A'}}\right)^{\sum_{j=1}^{n} k_{ij} \times y_i} \mod p. \tag{13}$$

Corollary 1. *It is computationally infeasible to determine that two group signatures were generated by the same user.*

Proof. We have

$$\frac{B}{B'} = \frac{S_i^{N_1 \times N_2 \times y_i}}{S_i^{N'_1 \times N'_2 \times y_i}} \mod p \qquad \text{(from (7))}$$

$$= \frac{\left(\prod_{j=1}^{n} s_{ij}\right)^{N_1 \times N_2 \times y_i}}{\left(\prod_{j=1}^{n} s_{ij}\right)^{N'_1 \times N'_2 \times y_i}} \mod p \qquad \text{(from (4))}$$

$$= \frac{\left(\prod_{j=1}^{n} {y_i}^{k_{ij}}\right)^{N_1 \times N_2 \times y_i}}{\left(\prod_{j=1}^{n} {y_i}^{k_{ij}}\right)^{N'_1 \times N'_2 \times y_i}} \mod p \qquad \text{(from (2))}$$

$$= \frac{\left(y_i^{\sum_{j=1}^{n} k_{ij}}\right)^{N_1 \times N_2 \times y_i}}{\left(y_i^{\sum_{j=1}^{n} k_{ij}}\right)^{N'_1 \times N'_2 \times y_i}} \mod p$$

$$= \frac{\left(g^{x_i \sum_{j=1}^{n} k_{ij}}\right)^{N_1 \times N_2 \times y_i}}{\left(g^{x_i \sum_{j=1}^{n} k_{ij}}\right)^{N'_1 \times N'_2 \times y_i}} \mod p$$

$$= \frac{\left(g^{x_i \times N_1 \times N_2}\right)^{\sum_{j=1}^{n} k_{ij} \times y_i}}{\left(g^{x_i \times N'_1 \times N'_2}\right)^{\sum_{j=1}^{n} k_{ij} \times y_i}} \mod p$$

$$= \left(\frac{g^A}{g^{A'}}\right)^{\sum_{j=1}^{n} k_{ij} \times y_i} \mod p. \qquad \text{(from (6))}$$

■

Corollary 1 holds true because the attacker does not have knowledge of $\sum_{j=1}^{n} k_{ij} \times y_i$, and solving Eqn. (13) boils down to a DLP hard problem along with unknown random parameter k_{ij}.

6.2.4. Traceability. Group managers T_j have access to $(y_i, \sum_{j=1}^{n} k_{ij})$ of each group member U_i. So, they can acquire (y_i, k_j) of U_i satisfying the equation

$$B = g^{A \times \sum_{j=1}^{n} k_{ij} \times y_i} \mod p$$

for $i = 1, 2, \ldots, m$, where m is the number of group members. Therefore, the set of group managers together can determine the actual signer, thereby making the proposed DGSS traceable if required.

7. Blockchain-based e-auction protocol using the DGSS

In this section, we introduce different types of roles used in the proposed blockchain-based e-auction protocol.

7.1. Roles. There are mainly four roles in the proposed protocol: the bidder, registration manager, auction manager and identity manager.

Bidder: The user/bidder U_i with unique identity ID_i chooses a private key x_i and computes public key $y_i = g^{x_i} \mod p$. The user/bidder with a valid key pair can bid for the goods.

Registration manager: The registration manager (RM) with private key x_{RM} and public key $y_{RM} = g^{x_{RM}} \bmod p$ is responsible for registering each bidder and computes respective key pairs for the bidders.

Auction manager: The auction manager (AM) with private key x_{AM} and public key $y_{AM} = g^{x_{AM}} \bmod p$ is responsible for maintaining the goods information. The AM is also responsible for determining the winning bid and also opens it to other bidders to check the validity of the winning bid.

Identity manager: The AM can only determine the winning bid without knowing the real identity of the winner. Hence, the AM sends the winning bid to the RM and the RM can find the real identity of the winner, but takes more time for determining the winner's real identity. For reducing the winner identity determination time, the AM sends winning bid and its information to the IM. The IM with private key x_{IM} and public key $y_{IM} = g^{x_{IM}} \bmod p$ processes it and sends its corresponding information to the RM. Finally, the RM can determine the winner's identity in a short time.

7.2. Proposed e-auction protocol. The proposed blockchain-based e-auction protocol comprises three phases: the bidder registration phase, bidding phase and the winner identification phase. Each phase of the proposed protocol is described as follows.

7.2.1. Bidder registration phase. Bidder U_i ($1 \le i \le m$) secretly sends (ID_i, y_i) to all the RM_j's ($1 \le j \le n$) for the registration. After receiving the registration request, each RM_j chooses a random integer k_{ij} such that $\gcd(k_{ij}, q) = 1$, where p and q are large prime numbers, and another random integer RN_{ij} for each U_i, and computes the certificate for the bidder as follows:

$$r_{ij} = y_i \times k_{ij} - x_{RM} \mod q, \tag{14}$$

$$s_{ij} = y_i^{k_{ij}} \mod p. \tag{15}$$

Now, each RM_j sends the corresponding (r_{ij}, s_{ij}) pair to the bidder U_i. The bidder collects all the (r_{ij}, s_{ij}) pairs from all the group managers and computes his/her summarized pair as follows:

$$R_i = \sum_{j=1}^{n} r_{ij}, \tag{16}$$

$$S_i = \prod_{j=1}^{n} s_{ij}. \tag{17}$$

After computing (R_i, S_i) and $\sum_{j=1}^{n} RN_{ij}$, bidder B_i can verify the validity of the certificate by the following equation:

$$S_i^{y_i} \mod p = (g^{R_i} \times y_{RM})^{x_i} \mod p. \tag{18}$$

The certificate is valid for bidder U_i if (18) holds. $\sum_{j=1}^{n} RN_{ij}$ is a linking value, and RM_j can use it to reveal the real identity of the winning bidder. In the meantime, RM_j stores the bidder's information as off-chain storage as in Table 1.

7.2.2. Bidding phase. If bidder U_i wants to participate in the auction, then he/she needs to compute the following:

1. Bidder U_i sends his/her random number RN_i and identity of the goods kept for auction GNO_i to the IM.
2. The IM selects a random integer d_i and computes $NO_i = GNO_i || d_i$
3. The IM signs NO_i and RN_i using x_{IM} as $S = \text{sign}_{x_{AM}}[NO_i, RN_i]$. The IM sends signature S and NO_i to the bidder.
4. The IM maintains an off-chain storage database for linking values as shown in Table 2.
5. The bidder can verify whether the $\sum_{j=1}^{n} RN_{ij}$ of the decryption is equal to the bidders $\sum_{j=1}^{n} RN_{ij}$, and this step protects anyone from modifying NO_i.
6. The bidder computes $M = (GNO_i || T_i, NO_i, P_i)$, where P_i is the price of his/her bid, and T_i the timestamp.
7. Now, bidder U_i chooses two random numbers N_1, N_2 in Z_q^* and computes the signature of the bid as follows:

$$A = x_i \times N_1 \times N_2 \mod q, \tag{19}$$

$$B = S_i^{N_1 \times N_2 \times y_i} \mod p, \tag{20}$$

$$C = M \times y_{AM}^{-N_1 \times A \times h(B)} \mod p, \tag{21}$$

$$D = N_1 - R_i \times h(C) \mod q. \tag{22}$$

8. Finally, bidder U_i sends his/her bid signature $\{A, B, C, D, GNO_i\}$ to the AM.
9. After receiving all the bids, the AM maintains auction information as in Table 3 in his/her off-chain storage.

7.2.3. Winner identification phase. After the bidding process, the AM, IM and RM will cooperate to find and publish the identity of winner U_w as follows.

Table 1. Bidder's information at RM_j's off-chain storage.

Identity	Public key	Integer	Linking value
ID_1	y_1	$\sum_{j=1}^{n} k_{1j}$	$\sum_{j=1}^{n} RN_{1j}$
ID_2	y_2	$\sum_{j=1}^{n} k_{2j}$	$\sum_{j=1}^{n} RN_{2j}$
ID_3	y_3	$\sum_{j=1}^{n} k_{3j}$	$\sum_{j=1}^{n} RN_{3j}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$
ID_m	y_m	$\sum_{j=1}^{n} k_{mj}$	$\sum_{j=1}^{n} RN_{mj}$

Table 2. Linking value of bidders at the IM's off-chain storage.

Linking value	NO_i
$\sum_{j=1}^{n} RN_1$	NO_1
$\sum_{j=1}^{n} RN_2$	NO_2
$\sum_{j=1}^{n} RN_3$	NO_3
$\vdots$	$\vdots$
$\sum_{j=1}^{n} RN_m$	NO_m

1. AM opens all the bids using the following equation:

$$M_i = C_i \times \Big[g^{D_i \times A_i} \times \prod_{j=1}^{n} y_{RM}^{-h(C_i)\times A_i} \times B_i^{h(C_i)} \Big]^{x_{RM}\times h(B_i)} \mod p. \quad (23)$$

After opening all the bids the AM finds the highest bid M_j by executing his/her smart contract and checks the validity of the bid with $GNO_i = \mathrm{head}(M_j, S)$.

2. The AM selects a random number N_3 and computes $Q_j = X_{RM} \times N_3 \mod q$ and $C'_j = M_j \times (C_j \times M'_j)^{N_3} \mod p$. Then, the AM publishes $\{A_j, B_j, C'_j, D_j, GNO_i\}$ and Q_j such that anyone can verify the validity of the winning bid. Every winning bid satisfies the following equation:

$$M_j = C'_j \times \Big[g^{D_j \times A_j} \times \prod_{j=1}^{n} y_{RM}^{-h(C_j)\times A_j} \times B_i j^{h(C_j)} \Big]^{Q_j\times h(B_j)} \mod p. \quad (24)$$

3. The AM sends the winning bid $\{A_j, B_j, C_j, D_j, GNO_j\}$ and NO_j to the IM. Then, IM finds the corresponding linking value $\sum_{j=1}^{n} RN_j$ of NO_j by looking at Table 2.

4. The IM then sends the winning bid $\{A_j, B_j, C_j, D_j, GNO_j\}$ and linking value $\sum_{j=1}^{n} RN_{ij}$ to the RM. The RM then finds the corresponding ID_j, y_j and $\sum_{j=1}^{n} k_{ij}$ of $\sum_{j=1}^{n} RN_{ij}$ by looking at Table 1. Then the RN checks whether

$$U_j = g^{A_j \times \sum_{j=1}^{n} k_{jj} \times y_j} \mod p$$

holds or not. If so, U_j that has identified ID_j as the winner.

Now, RM_j sends the transaction details of winning bidder to the ordering service. The ordering service collects all such transactions to create a new block. The new block will be sent to all RM_j's. RM_j's verify the block and append it into their blockchain.

Table 3. Auction information table at the AM's off-chain storage.

User	Signature
U_1	$\{A_1, B_1, C_1, D_1, GNO_1\}$
U_2	$\{A_2, B_2, C_2, D_2, GNO_2\}$
U_3	$\{A_3, B_3, C_3, D_3, GNO_3\}$
$\vdots$	$\vdots$
U_m	$\{A_m, B_m, C_m, D_m, GNO_m\}$

8. Conclusion

A novel DGSS was proposed to address the identity privacy challenges in blockchain based applications. Lee *et al.* (2009) group signature scheme was extended to the DGSS by decentralization of the group manager to eliminate the basic requirement of having trust in the group manager and also to improve the identity privacy of group members. The security properties like unforgeability, anonymity, unlinkability and traceability for the proposed DGSS are also discussed. The proposed DGSS were more suitable for permissioned blockchain-based applications. However, the use of anonymous signatures for public blockchains and the mathematical security model of the proposed DGSS were explored in future work. The proof of correctness for the proposed scheme ensures that the original message can still be reconstructed correctly, even after it has been distributed among several group managers. Also a framework of the blockchain-based e-auction protocol with the DGSS is proposed.

Acknowledgment

The first author acknowledges the financial assistance received in the form of a Senior Research Fellowship from the Council of Scientific & Industrial Research (CSIR), Government of India.

References

Agarwal, A. and Saraswat, R. (2013). A survey of group signature technique, its applications and attacks, *International Journal of Engineering and Innovative Technology* **2**(10): 28–35.

Al Jawaheri, H., Al Sabah, M., Boshmaf, Y. and Erbad, A. (2020). Deanonymizing tor hidden service users through bitcoin transactions analysis, *Computers & Security* **89**: 101684.

Androulaki, E., Karame, G. O., Roeschlin, M., Scherer, T. and Capkun, S. (2013). Evaluating user privacy in bitcoin, *International Conference on Financial Cryptography and Data Security, Okinawa, Japan*, pp. 34–51.

Ateniese, G., Song, D. and Tsudik, G. (2002). Quasi-efficient revocation of group signatures, *International Conference on Financial Cryptography, Southampton, Bermuda*, pp. 183–197.

Bellare, M., Micciancio, D. and Warinschi, B. (2003). Foundations of group signatures: Formal definitions, simplified requirements, and a construction based on general assumptions, *International Conference on the Theory and Applications of Cryptographic Techniques, Warsaw, Poland*, pp. 614–629.

Blass, E.-O. and Kerschbaum, F. (2018). Strain: A secure auction for blockchains, *European Symposium* *on Research in Computer Security, Barcelona, Spain*, pp. 87–110.

Bonneau, J., Miller, A., Clark, J., Narayanan, A., Kroll, J.A. and Felten, E.W. (2015). SOK: Research perspectives and challenges for bitcoin and cryptocurrencies, *2015 IEEE Symposium on Security and Privacy, San Jose, USA*, pp. 104–121.

Chaum, D. and Van Heyst, E. (1991). Group signatures, *Workshop on the Theory and Application of of Cryptographic Techniques, Brighton, UK*, pp. 257–265.

Chen, L. and Pedersen, T.P. (1994). New group signature schemes, *Workshop on the Theory and Application of of Cryptographic Techniques, Perugia, Italy*, pp. 171–181.

Chen, Y.-H., Chen, S.-H. and Lin, I.-C. (2018). Blockchain based smart contract for bidding system, *2018 IEEE International Conference on Applied System Invention (ICASI), Chiba, Japan*, pp. 208–211.

Conti, M., Kumar, E.S., Lal, C. and Ruj, S. (2018). A survey on security and privacy issues of bitcoin, *IEEE Communications Surveys & Tutorials* **20**(4): 3416–3452.

Feng, Q., He, D., Zeadally, S., Khan, M.K. and Kumar, N. (2019). A survey on privacy protection in blockchain system, *Journal of Network and Computer Applications* **126**(8): 45–58.

Fernandez-Vazquez, S., Rosillo, R., De La Fuente, D. and Priore, P. (2019). Blockchain in fintech: A mapping study, *Sustainability* **11**(22): 6366.

Galal, H.S. and Youssef, A.M. (2018). Verifiable sealed-bid auction on the Ethereum blockchain, *International Conference on Financial Cryptography and Data Security, Nieuwpoort, Curaçao*, pp. 265–278.

Gao, W., Hatcher, W.G. and Yu, W. (2018). A survey of blockchain: Techniques, applications, and challenges, *2018 27th International Conference on Computer Communication and Networks (ICCCN), Hangzhou, China*, pp. 1–11.

Jouini, M., Rabai, L.B.A. and Aissa, A.B. (2014). Classification of security threats in information systems, *Procedia Computer Science* **32**: 489–496.

Karame, G. (2016). On the security and scalability of bitcoin's blockchain, *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria*, pp. 1861–1862.

Kim, H.-J., Lim, J.I. and Lee, D.H. (2000). Efficient and secure member deletion in group signature schemes, *International Conference on Information Security and Cryptology, Seoul, Korea*, pp. 150–161.

Kobusińska, A., Brzeziński, J., Boroń, M., Inatlewski, Ł., Jabczyński, M. and Maciejewski, M. (2016). A branch hash function as a method of message synchronization in anonymous P2P conversations, *International Journal of Applied Mathematics and Computer Science* **26**(2): 479–493, DOI: 10.1515/amcs-2016-0034.

Kong, W., Jiang, B., Fan, Q., Zhu, L. and Wei, X. (2018). Personal identification based on brain networks of EEG signals, *International Journal of Applied Mathematics and Computer Science* **28**(4): 745–757, DOI: 10.2478/amcs-2018-0057.

Kosba, A., Miller, A., Shi, E., Wen, Z. and Papamanthou, C. (2016). Hawk: The blockchain model of cryptography and privacy-preserving smart contracts, *2016 IEEE Symposium on Security and Privacy (SP), San Jose, USA*, pp. 839–858.

Krishna, V. (2009). *Auction Theory*, Academic Press, San Diego.

Lafourcade, P., Nopere, M., Picot, J., Pizzuti, D. and Roudeix, E. (2019). Security analysis of auctionity: A blockchain based e-auction, *International Symposium on Foundations & Practice of Security FPS 19, Toulouse, France*, pp. 290–307.

Lee, C.-C., Ho, P.-F. and Hwang, M.-S. (2009). A secure e-auction scheme based on group signatures, *Information Systems Frontiers* **11**(3): 335–343.

Li, S., Zhang, Y., Wang, Y. and Sun, W. (2019). Utility optimization–based bandwidth allocation for elastic and inelastic services in peer-to-peer networks, *International Journal of Applied Mathematics and Computer Science* **29**(1): 111–123, DOI: 10.2478/amcs-2019-0009.

Li, X., Jiang, P., Chen, T., Luo, X. and Wen, Q. (2017). A survey on the security of blockchain systems, *Future Generation Computer Systems* **107**: 841–853.

Nakamoto, S. and Bitcoin, A. (2008). A peer-to-peer electronic cash system, `https://bitcoin.org/bitcoin.pdf`.

Reid, F. and Harrigan, M. (2013). An analysis of anonymity in the bitcoin system, *Security and Privacy in Social Networks, Boston, USA*, pp. 197–223.

Ron, D. and Shamir, A. (2013). Quantitative analysis of the full bitcoin transaction graph, *International Conference on Financial Cryptography and Data Security, Okinawa, Japan*, pp. 6–24.

Sánchez, D.C. (2018). Raziel: Private and verifiable smart contracts on blockchains, *arXiv:* 1807.09484.

Sun, Y., Sun, Y., Luo, M., Gu, L., Zheng, S. and Yang, Y. (2013). Comment on Lee *et al.*'s group signature and e-auction scheme, *Information Systems Frontiers* **15**(1): 133–139.

Tsai, C.-Y., Ho, P.-F. and Hwang, M.-S. (2018). A secure group signature scheme., *IJ Network Security* **20**(2): 201–205.

Wang, X., Zha, X., Ni, W., Liu, R.P., Guo, Y.J., Niu, X. and Zheng, K. (2019). Survey on blockchain for internet of things, *Computer Communications* **136**: 10–29.

Zhang, R., Xue, R. and Liu, L. (2019). Security and privacy on blockchain, *ACM Computing Surveys* **52**(3): 1–34.

Zheng, H., Wu, Q., Xie, J., Guan, Z., Qin, B. and Gu, Z. (2020). An organization-friendly blockchain system, *Computers & Security* **88**: 101598.

S. Devidas received his MTech from Jawaharlal Nehru Technological University Hyderabad in 2013. He received his Master of Computer Applications (MCA) degree from Jawaharlal Nehru Technological University in 2010. He is currently pursuing his PhD at the School of Computer and Information Sciences, University of Hyderabad. His research interests are blockchain technology and information security.

Subba Rao Y.V. is an assistant professor in the School of Computer and Information Sciences at the University of Hyderabad, where he has been since 2004. He received an MSc from the University of Hyderabad, an MPhil from the University of Hyderabad and an MTech from the Indian Statistical Institute (ISI). He received his PhD in computer science from the University of Hyderabad. Prior to coming to the University of Hyderabad he was associated with Chennai Mathematical Institute (CMI), Bapatla Engineering College (BEC) and ICFAI University. His research interests span cryptography and related areas. He has worked on two projects with the Defence Electronics Research Laboratory (DLRL) on cracking the famous A5/1 and A5/3 ciphers that are used to provide over-the-air communication privacy in the GSM cellular telephone standard.

N. Rukma Rekha is an associate professor in the School of Computer and Information Sciences at the University of Hyderabad, where she has been since 2007. She received her PhD in computer science from Andhra University in 2014. Her research interests span information security, cryptography, pervasive computing and blockchain technology. Much of her work has been on improving the understanding, design, and performance of cryptographic, information security and blockchain protocols mainly through applications. She has worked on two projects with the Defence Electronics Research Laboratory (DLRL) on cracking the famous A5/1 and A5/3 ciphers that are used to provide over-the-air communication privacy in the GSM cellular telephone standard.

Received: 8 December 2020
Revised: 11 March 2021
Accepted: 26 April 2021