A Lightweight PUF-Based Group Authentication Scheme for Privacy-Preserving Metering Data Collection in Smart Grid

Ya-Nan Cao[1], Yujue Wang[2], Yong Ding[3,4], Zhenwei Guo[2], Changsong Yang[3], and Hai Liang[3(✉)]

[1] Guangdong University of Science and Technology, Dongguan, China
[2] Hangzhou Innovation Institute of Beihang University, Hangzhou, China
[3] Guangxi Key Laboratory of Cryptography and Information Security, School of Computer Science and Information Security, Guilin University of Electronic Technology, Guilin, China
lianghai@guet.edu.cn
[4] Institute of Cyberspace Technology, HKCT Institute for Higher Education, Hong Kong SAR, China

Abstract. With the development of information and communication technologies, the services provided by smart grid attract more users to join smart grid. However, with the explosive growth of the number of smart meters, the transmission between the control center and smart meters has brought huge data transmission and computing costs to the smart grid, which is prone to network congestion, untimely power service supply and other network conditions. This paper proposes a Physically Unclonable Function (PUF)-based lightweight group authenticated metering data collection scheme with privacy protection in smart grid (PGAC). The PGAC scheme is designed with lightweight cryptographic primitives, which is suitable for resource-constrained devices. In addition, the PGAC scheme divides the users into groups and uses the gateway as the repeater and aggregator of the communication data of each group, which reduces signaling and communication costs for activating additional request messages from a large number of devices. Security analysis shows that the PGAC scheme maintains the security and privacy of the data collection process for large-scale smart meters. Functional analysis, theoretical analysis and performance analysis show that PGAC scheme has better authentication function and low communication cost.

Keywords: Physically unclonable function · Security · Privacy · Authentication · Aggregation · Encryption · Smart grid

1 Introduction

Smart grid is a new type of intelligent network that optimizes power service and facilitates the user's participation in the operation and management of the power

H. Gao et al. (Eds.): CollaborateCom 2023, LNICST 562, pp. 321–340, 2024.
https://doi.org/10.1007/978-3-031-54528-3_18

system [22]. At present, electricity mainly comes from large power companies [20]. The control center, as the management organization of the power company, makes two-way communication with the power grid equipments, collects the electricity consumption information of users in real time, and provides timely and accurate electricity demand data for the implementation of smart grid pricing and power supply. Therefore, smart grid has achieved efficient and reliable power management [19,26], achieved dynamic balance of power supply, and simplified the cumbersome traditional electricity information collection method.

The metering data collection process of the control center (CC) for smart meters (SMs) is carried out in the public network, and the integrity and confidentiality of the metering data may be threatened [1,10]. The adversary may affect the user's billing bill or even disrupt the supply and demand balance of the grid by tampering with the metering data transmitted in the communication [14]. The adversary may be able to infer the daily behavior of users by analyzing their power consumption information over time. Smart meters are installed on the external grid equipment, users may change the configuration of the smart meter to reduce billing. In addition, the control center needs to communicate with a large number of smart meters at the same time, and a large number of messages are transmitted within the network at the same time may cause network congestion and other situations.

1.1 Related Works

Many solutions have been proposed to solve the problem of identity authentication and key agreement. Kumar et al. [18] proposed an identity authentication scheme based on elliptic curve cryptosystem, which establishes secret session keys through mutual authentication between power grid equipment and control center, so as to securely exchange electricity demand information. Khan et al. [17] established a lightweight smart grid authentication and key negotiation protocol between users and servers based on elliptic curve encryption mechanism and biometrics technology, which does not have key escrow problems and improves the security and confidentiality of the protocol. Zhang et al. [27] designed a lightweight anonymous authentication and key protocol scheme for smart grid, which not only ensures the anonymity and non-traceability of smart meters, but also realizes the rapid mutual authentication between smart meters and service providers.

There are a large number of users and data in smart grid, and efficient data aggregation solutions for privacy protection has become a hot research direction in recent years. Lu et al. [21] discussed the edge layer data security and privacy issues faced by data aggregation schemes, and designed a three-layer privacy protection data aggregation scheme combining edge computing and blockchain architecture. This scheme combines Paillier homomorphic encryption and one-way hash chain technology, and uses edge servers to aggregate data from the same region, and can filter false data in advance. Fan, Liu and Zeng [9] proposed a privacy-protecting data aggregation scheme based on blockchain, which adopted the leader election algorithm and Paillier encryption algorithm to ensure that

data privacy would not be disclosed in the process of metering data collection. In order to solve the problem of communication security, Guan et al. [12] proposed a smart grid privacy protection aggregation authentication scheme, which realizes data source authentication and data aggregation efficiently and flexibly.

Many PUF-based secure authentication and communication schemes have been proposed. Gope and Sikdar [11] introduced an identity authentication key protocol scheme based on PUF to protect privacy and ensure the physical security of smart meters. Cao et al. [2] proposed a PUF-based lightweight identity authentication and privacy protection data collection scheme in smart grid, which can simultaneously solve the security and privacy problems faced in the process of smart grid metering data collection. Ren et al. [24] proposed an Internet of Things (IoT) packet authentication and data transmission scheme based on PUFs. This scheme generates session keys based on the output of PUFs, and sets group leaders to aggregate and forward authentication information, thus reducing the communication cost of activating additional request messages from a large number of devices.

1.2 Our Contributions

In order to solve the security, privacy and efficiency problems of smart grid data collection for multiple users at the same time, a lightweight PUF-based privacy-preserving group authentication metering data collection scheme (PGAC) is proposed. In the PGAC scheme, gateway is introduced as a repeater and aggregator, and PUF is used to realize group authentication. Compared with the public key cryptosystem, the computing cost, communication cost and storage cost of the PGAC scheme are significantly reduced, and can better meet the requirements of resource-constrained devices. The security analysis shows that the scheme can not only guarantee the unforgetability of the identity of the control center, gateway and smart meters, but also protect the physical security of smart meters. Experimental analysis show that the scheme is efficient and can be widely used in multi-user data acquisition scenarios.

1.3 Paper Organization

The rest of this article is organized as follows. Section 2 briefly introduces the knowledge of PUF and fuzzy extractor. In Sect. 3, the system model and system requirements of PGAC are presented. Section 4 presents a concrete PGAC construction in detail. Section 5 includes security analysis, theoretical comparision, and experimental performance analysis. In Sect. 6, the paper is concluded.

2 Preliminaries

This section summarizes some basic knowledge of PUF and fuzzy extractor.

2.1 Physical Unclonable Function

PUF is first proposed by Pappu [23] in 2002. A PUF is a physical entity that is hard to clone based on random physical factors introduced during its production [5]. With a specific challenge $Chal$, the PUF can generate a random, unique and unclonable response R according to its production variability, i.e. $R = PUF(Chal)$. PUF can be used for privacy protection, identity authentication and key generation for resource-constrained devices because it is unique, non-clonable, lightweight and resistant to physical attacks. It should also be noted that PUF has a drawback in that it is difficult to reproduce the original response for the same input due to bit errors caused by factors such as temperature or aging effects [4].

2.2 Fuzzy Extractor

Fuzzy extractor allows input data with a certain amount of noise (or error), as long as the inputs are close enough to extract the same uniform random string. Therefore, fuzzy extractor is able to convert unevenly distributed data such as biometrics that have bias at the time of input into uniformly distributed random numbers required by the cryptosystem. Fuzzy extractor consists of two algorithms: $FE.Gen$ and $FE.Rec$ [8,16]. $FE.Gen$ is the key generation algorithm, which takes as input a random string R, and outputs a key K and a help string hd, i.e., $(K, hd) = FE.Gen(R)$. $FE.Rec$ is the key recovery algorithm, which takes as input a random string R containing noise and a help string hd, and outputs the key K, i.e., $K = FE.Rec(R, hd)$.

3 System Model and Requirements

This section defines the architecture of PGAC and formalizes the system requirements.

3.1 System Model

The PGAC system structure is shown in Fig. 1, which consists of three types of entities, including smart meters (SMs), gateway (GW) and control center (CC), and the communication between different entities is through a public network. During the metering data collection process, CC generates a set of data collection request information and sends it to the target GW. The GW forwards the set of information to the group of SMs after verifying the source of the message. Then each smart meter verifies the source of the request information, encrypts the collected metering data and sends it to the GW. After that, the GW verifies the source of these responses, aggregates the authentication code of the SMs into one aggregated authentication code, and aggregates the response and the authentication code into one aggregated response. Finally, the CC verifies the source of the aggregated response and the authenticity of the aggregated authentication code, decrypts the encrypted metering data, and obtains the metering data collected by the SMs.

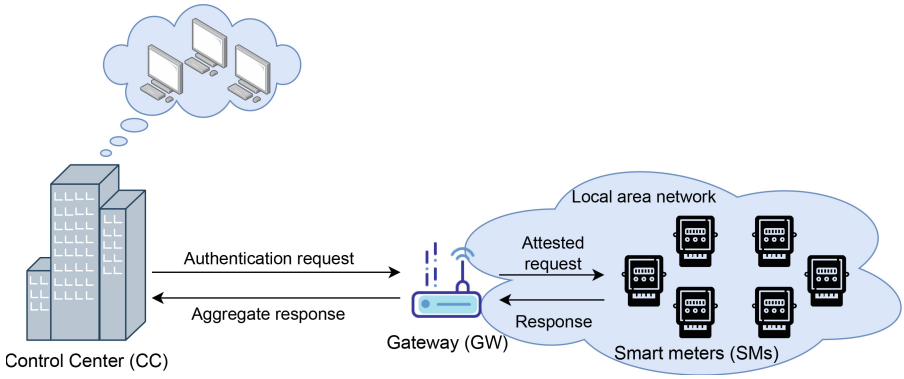

Fig. 1. PGAC system model in smart grid.

- SM is installed on the user side and embedded with a PUF component, which is responsible for collecting the user's power consumption information. After receiving the collection request from CC, the SM transmits the encrypted metering data to CC through the GW, which has the functions of data collection, authentication and encryption in the PGAC system.
- CC is the largest power center of the PGAC system, which is responsible for regulating the balance of power supply and demand and power billing. It has the functions of initiating communication, authentication, encryption and decryption and devices registration in the PGAC system.
- GW is also embedded with a PUF, and is installed by CC between the CC and the SMs based on the geographical range and number of nearby users, which is responsible for forwarding messages to users of its group and aggregating data. It has the functions of authentication, encryption and aggregation in the PGAC system.

3.2 System Requirements

In the PGAC system, a large number of smart meters communicate with the control center through the public network. Besides the security of communication, privacy and physical security of smart meters are vulnerable to threats, the power system often faces high computing overhead and bandwidth occupation due to the massive data transmission at the same time. Therefore, the PGAC system needs to meet the following requirements.

Confidentiality of metering data: Only CC can obtain the real information related to the metering data sent by smart meters, and the regional GW cannot effectively analyze the data sent by smart meters.

Resistance to man-in-the-middle attacks: In the whole communication process, no malicious entity can tamper with or replace the transmitted data without being discovered by the receiver.

Resistance to impersonation attacks: Any entity posing as the identity of CC, GW and smart meter will be detected by the receiver.

Resistance to replay attacks: Any entity sending duplicate tuples can be detected by the receiver.

Resistance to physical attacks: Any tampering with the smart meter configuration can be detected by GW and CC, and any tampering with GW can be detected by CC.

Lightweight: There is not a lot of complex computing and there is no resource-intensive computing at the smart meter side.

Low bandwidth consumption: In the process of collecting massive users' metering data, the data length should be compressed and the number of messages should be reduced.

4 A Concrete PGAC Construction

This section describes the proposed PGAC construction.

4.1 PGAC Construction

System Setup. With the security parameter γ, CC randomly selects a large prime q, chooses a collision-resistant hash function $H : \{0,1\}^* \rightarrow \{0,1\}^\delta$ and a secure symmetric AES encryption scheme Π with encryption algorithm ENC and decryption algorithm DEC, where δ is determined by γ. CC publishes the parameters $pubparam = \{q, H, ENC, DEC\}$.

After that, CC divides the power consumption range into multiple areas according to the information such as geographical location and number of users, and assigns a GW to each area. All smart meters in this area communicate with CC through the gateway they have. Then CC generates a unique identity GID and a challenge $C_{GW,CC}$ for the GW, and sends GID and $C_{GW,CC}$ to the GW through a secure channel.

With the challenge $C_{GW,CC}$, the GW computes the PUF response $R_{GW,CC}$

$$R_{GW,CC} = PUF_{GW,CC}(C_{GW,CC}) \tag{1}$$

and sends $R_{GW,CC}$ to CC over a secure channel.

Then, CC calculates the shared key $K_{GW,CC}$ with the GW and the help string $hd_{GW,CC}$

$$(K_{GW,CC}, hd_{GW,CC}) = FE.Gen(R_{GW,CC}) \tag{2}$$

CC stores $\{C_{GW,CC}, K_{GW,CC}, GID\}$ in the database and sends $hd_{GW,CC}$ to the GW through a secure channel.

At last, the GW stores $\{C_{GW,CC}, hd_{GW,CC}, GID\}$ securely at local.

Registration. When the smart meter SM_i newly connected to the smart grid, it should perform the registration process with CC and the GW. Above all, CC randomly generates a unique identity ID_i and a set of challenges $C_i =$

$\{C_{i,1}, C_{i,2}, ..., C_{i,s}\}$ for SM_i. CC also assigns a GW as its group leader for SM_i based on its area. Then CC sends $\{ID_i, C_i, GID\}$ to SM_i and ID_i to GW through secure channels.

With the challenges C_i, SM_i computes the PUF responses $R_i = \{R_{i,1}, R_{i,2}, ..., R_{i,s}\}$ as follows

$$R_{i,j} = PUF_i(C_{i,j}), j = 1, 2, ..., s \tag{3}$$

and sends these PUF responses to CC through a secure channel.

Then, CC calculates a set of shared keys $K_i = \{K_{i,1}, K_{i,2}, ..., K_{i,s}\}$ with SM_i and help strings $hd_i = \{hd_{i,1}, hd_{i,2}, ..., hd_{i,s}\}$ as follows

$$(K_{i,j}, hd_{i,j}) = FE.Gen(R_{i,j}), j = 1, 2, ..., s \tag{4}$$

CC stores $\{C_i, K_i, ID_i, GID\}$ in the database and sends hd_i to SM_i through a secure channel.

After receiving ID_i sent by CC, the GW generates a challenge $C_{i,GW}$ and sends it to SM_i through a secure channel.

Then SM_i computes the PUF responses $R_{i,GW}$ as follows

$$R_{i,GW} = PUF_i(C_{i,GW}) \tag{5}$$

and sends $R_{i,GW}$ to the GW through a secure channel.

Next, the GW calculates the shared keys $K_{i,GW}$ with SM_i and help strings $hd_{i,GW}$ as follows

$$(K_{i,GW}, hd_{i,GW}) = FE.Gen(R_{i,GW}) \tag{6}$$

The GW stores $\{C_{i,GW}, K_{GW,CC}, ID_i\}$ securely at local and sends $hd_{i,GW}$ to SM_i through a secure channel.

At last, SM_i stores $\{hd_i, ID_i, GID, hd_{i,GW}\}$ securely at local.

Request. CC retrieves the information $\{C_{GW,CC}, K_{GW,CC}, ID_{GW,CC}, GID\}$ for the GW with the identity GID and the group members SM_i $(i = 1, 2, ..., n)$ information $\{(C_{i,j}, K_{i,j}, ID_i) : i = 1, 2, ..., n\}$. Then CC generates a request information Req, a timestamp t and a random number $r \in Z_q^*$. Then CC calculates

$$r_i^* = H(K_{i,j} \,\|\, t) \oplus r, i = 1, 2, ..., n \tag{7}$$

Also, CC calculates the authentication codes for SM_i $(i = 1, 2, ..., n)$

$$AUTH_{CC-i} = H(Request \| C_{i,j} \| ID_i \| GID \| t \| r_i^* \| K_{i,j}) \tag{8}$$

and the authentication code for the GW

$$AUTH_{CC-GW} = H(ID_{GW} \| GID \| t \| Req \| K_{GW,CC} \| C_{GW,CC} \| IDEN_{CC-SMs}) \tag{9}$$

where $IDEN_{CC-SMs} = \{(ID_i, C_{i,j}, r_i^*, AUTH_{CC-i} : i = 1, 2, ..., n\}$. At last, CC sends the request data tuple $DT_1 = (ID_{GW}, t, AUTH_{CC-GW}, IDEN_{CC-SMs}, Req)$ to the GW.

Forwarding. After receiving the request data tuple DT_1, the GW retrieves ($C_{GW,CC}$, $hd_{GW,CC}$) from its memory, then generates the PUF response

$$R'_{GW} = PUF_{GW}(C_{GW,CC}) \tag{10}$$

and recovers the shared key with CC by invoking $FE.Rec$ algorithm

$$K_{GW,CC} = FE.Rec(R'_{GW}, hd_{GW,CC}) \tag{11}$$

Next, the GW verifies the authentication code of CC as follows

$$H(ID_{GW}\|GID\|t\|Req\|K_{GW,CC}\|C_{GW,CC}\|IDEN_{CC-SMs}) \stackrel{?}{=} AUTH_{CC-GW} \tag{12}$$

If it holds, the GW broadcasts the forwarding data tuple $DT_2 = (IDEN_{CC-SMs}, t, Req)$ to SM_i $(i = 1, 2, ..., n)$, otherwise the protocol terminates.

Encryption. After receiving the request information broadcast by the GW, SM_i generates the PUF response as follows

$$R'_{i,j} = PUF_i(C_{i,j}) \tag{13}$$

Next, SM_i retrieves $hd_{i,j}$ from its memory and invokes $Fe.Rec$ to recover the key

$$K_{i,j} = FE.Rec(R'_{i,j}, hd_{i,j}) \tag{14}$$

Then SM_i retrieves GID and verifies the correctness of CC's authentication code $AUTH_{CC-i}$ as follows

$$H(Req\|C_{i,j}\|ID_i\|GID\|t\|r_i^*\|K_{i,j}) \stackrel{?}{=} AUTH_{CC-i} \tag{15}$$

If it holds, SM_i calculates

$$r = H(K_{i,j} \,\|\, t) \oplus r_i^* \tag{16}$$

$$SK_{i,j} = H(r\|K_{i,j}\|ID_i\|GID\|t) \tag{17}$$

SM_i encrypts the metering data M_i using the session key $SK_{i,j}$

$$E_i = ENC(SK_{i,j}, M_i) \tag{18}$$

and SM_i calculates the authentication code

$$AUTH_{i-CC} = H(ID_i\|GID\|E_i\|r\|K_{i,j}) \tag{19}$$

Next, SM_i retrives $\{C_{i,GW}, hd_{i,GW}\}$ and calculates

$$R'_{i,GW} = PUF_i(C_{i,GW}) \tag{20}$$

$$K_{i,GW} = FE.Rec(R'_{i,GW}, hd_{i,GW}) \tag{21}$$

Also, SM_i generates a timestamp t_i and calculates

$$AUTH_{i-GW} = H(ID_i\|E_i\|t_i\|GID\|K_{i,GW}\|AUTH_{i-CC}) \tag{22}$$

At last, SM_i sends the response data tuple $DT_3 = (E_i, t_i, ID_i, AUTH_{i-CC}, AUTH_{i-GW})$ to the GW.

Aggregation. After receiving the data $DT_{3,i}$ from the group's smart meter SM_i, the GW retrieves the stored $K_{i,GW}$, and verifies the authentication code $AUTH_{i-GW}$ as follows

$$H(ID_i \| E_i \| t_i \| GID \| K_{i,GW} \| AUTH_{i-CC}) \stackrel{?}{=} AUTH_{i-GW} \tag{23}$$

If authentication codes $AUTH_{i-GW}$ $(i = 1, 2, ..., n)$ are all verified successfully, the GW calculates the aggregated authentication code

$$AUTH = AUTH_{1-CC} \oplus AUTH_{2-CC} \oplus ... \oplus AUTH_{n-CC} \tag{24}$$

Then the GW generates a timestamp t_{GW} and calculates

$$AUTH_{GW-CC} = H(AUTH \| GID \| t_{GW} \| K_{GW,CC} \| CT_{SMs}) \tag{25}$$

where $CT_{SMs} = \{(ID_i, E_i) : i = 1, 2, ..., n\}$. In the end, the GW aggregates the messages into one message $MES = (GID, t_{GW}, AUTH_{GW-CC}, AUTH, CT_{SMs})$ and sends it to the CC.

Decryption. After receiving the aggregated response MES from the GW, CC verifies the identity authentication code $AUTH_{GW-CC}$

$$H(AUTH \| GID \| t_{GW} \| K_{GW}) \stackrel{?}{=} AUTH_{GW-CC} \tag{26}$$

If it holds, CC calculates the identity authentication codes for the group of smart meters SM_i $(i = 1, 2...n)$ as follows

$$AUTH'_{i-CC} = H(ID_i \| GID \| E_i \| r \| K_{i,j}) \tag{27}$$

and verifies the authenticity of the group authentication code $AUTH$

$$AUTH'_{1-CC} \oplus AUTH'_{2-CC} \oplus ... \oplus AUTH'_{n-CC} \stackrel{?}{=} AUTH \tag{28}$$

If it holds, CC calculates the session keys with each SM_i $(i = 1, 2, ..., n)$

$$SK_{i,j} = H(r \| K_{i,j} \| ID_i \| GID \| t) \tag{29}$$

then uses the session keys $SK_{i,j}$ $(i = 1, 2, ..., n)$ to decrypt the corresponding metering data ciphertext respectively

$$M_i = DEC(SK_{i,j}, E_i) \tag{30}$$

and stores the metering data M_i $(i = 1, 2, ..., n)$ in the database to provide data support for subsequent power services.

Theorem 1. *The proposed PGAC construction is correct.*

Proof. To prove the correctness of the proposed PGAC construction, it is only necessary to prove that all equations in Eq. (12), Eq. (15), Eq. (23), Eq. (26), Eq. (28) and Eq. (30) hold.

1) The GW generates the response R'_{GW} through Eq. (10), recovers the key $K_{GW,CC}$ through Eq. (11), and then verifies Eq. (12) according to the data tuple from CC as follows

$$H(ID_{GW}\|GID\|t\|Req\|IDEN_{CC-SMs}) = AUTH_{CC-GW}$$

Therefore, the GW can successfully verify the authenticity of the data tupleDT_1 sent by CC.

2) SM_i $(i = 1, 2, ..., n)$ generates the responses $R'_{i,j}$ $(i = 1, 2, ..., n)$ through Eq. (13), recovers the keys $K_{i,j}$ $(i = 1, 2, ..., n)$ through Eq. (14), and then verifies Eq. (15) according to the data tuple $DT_2 = (t, Req, IDEN_{CC-SMs})$ from the GW as follows

$$H(Req\|C_{i,j}\|ID_i\|GID\|t\|r_i^*\|K_{i,j}) = AUTH_{CC-i}$$

Therefore, SM_i $(i = 1, 2, ..., n)$ can successfully verify the authenticity of the request information sent by CC.

3) The GW verifies Eq. (23) based on the data tuple $DT_{3,i} = (E_i, t_i, ID_i, AUTH_{i-CC}, AUTH_{i-GW})$ from SM_i and the key $K_{i,GW}$ as follow

$$H(ID_i\|E_i\|t_i\|GID\|K_{i,GW}\|AUTH_{i-CC}) = AUTH_{i-GW} \tag{31}$$

Therefore, the GW can successfully verify the authenticity of the data tuples $DT_{3,i}$ sent by SM_i $(i = 1, 2, ..., n)$.

4) CC verifies Eq. (26) based on the aggregated response $MES = (GID, t_{GW}, AUTH_{GW-CC}, AUTH, CT_{SMs})$ from the GW and the key $K_{GW,CC}$ as follows

$$H(AUTH\|GID\|t_{GW}\|K_{GW,CC}\|CT_{SMs}) = AUTH_{GW-CC}$$

Therefore, CC can successfully verify the authenticity of the aggregated response MES sent by the GW.

5) CC verifies Eq. (28) based on the aggregated response MES, random number r, and the keys $K_{i,j}$ $(i = 1, 2, ..., n)$ from the GW as follows

$$AUTH'_{i-CC} = H(ID_i\|GID\|E_i\|r\|K_{i,j}) = AUTH_{i-CC}, i = 1, 2, ..., n$$

$$AUTH'_{1-CC} \oplus AUTH'_{2-CC} \oplus ... \oplus AUTH'_{n-CC} = AUTH$$

Therefore, CC can successfully verify the authenticity of ciphertext E_i $(i = 1, 2, ..., n)$ of the metering data in the aggregate response MES.

6) For ciphertext E_i of metering data from SM_i $(i = 1, 2, ..., n)$, CC can calculate the session key $SK_{i,j}$ through Eq. (29), which meets

$$DEC(SK_{i,j}, E_i) = DEC(SK_{i,j}, ENC(SK_{i,j}, M_i)) = M_i$$

Therefore, CC can correctly decrypt ciphertext E_i of the metering data E_i $(i = 1, 2, ..., n)$.

Therefore, the proposed PGAC scheme is correct.

4.2 Construction Optimization

Note that if CC fails to verify $AUTH$ during the decryption phase, it needs to find out the source of the error. To this end, the truncation code technology can be adopted to optimize the above PGAC construction. The truncation code enables CC to locate the source of faulty data effectively, and avoids the waste of computing resources, communication cost and time of grid devices caused by the repeated transmission and computation of a large number of data. Furthermore, the truncation code replaces long data transmission with short data, which greatly reduces communication transmission costs and bandwidth consumptions. The optimized PGAC construction has the same system setup, registration, request, forwarding and encryption phases as in Sect. 4.1, thus they are omitted here.

Aggregation. This phase is the same as in Sect. 4.1 before aggregating the data into the message MES. The GW truncates the first eight bits of $AUTH_{i-CC}$, which is represented as $AUTH_{i-trun}$. Then the GW aggregates the response data tuples from SM_i $(i = 1, 2, ..., n)$ into one message $MES' = (GID, t_{GW}, AUTH_{GW-CC}, AUTH, \{(ID_i, E_i, AUTH_{i-trun}) : i = 1, 2, ...n\})$ and sends it to CC.

Decryption. This phase has the same authentication process for SM_i $(i = 1, 2, ..., n)$ and GW as in Sect. 4.1. If Eq. (28) does not hold, CC verifies whether $AUTH'_{i-trun} = AUTH_{i-trun}$ $(i = 1, 2, ..., n)$ holds one by one. If any truncation code is not satisfied, CC re-requests the metering data of the owner of the wrong truncation code. Otherwise, CC starts the metering data collection process for all smart meters SM_i $(i = 1, 2, ..., n)$ in the group again.

5 Scheme Analysis

This section analyzes the security and performance of the proposed PGAC construction.

5.1 Security Analysis

Theorem 2. *Assuming that the symmetric encryption scheme* Π *is secure, the proposed PGAC scheme can protect the privacy of users, that is, other entities cannot infer any metering data information of users from the data transmitted during the communication.*

Proof. In the proposed PGAC construction, the confidentiality of metering data depends on the confidentiality of the session key. The legitimate CC has the shared key $K_{i,j}$ and can compute $r_i^* = H(K_{i,j} \| t) \oplus r$ and the session key $SK_{i,j} = H(r \| K_{i,j} \| ID_i \| GID \| t)$. SM_i is able to calculate $R'_{i,j} = PUF_i(C_{i,j})$, thus recovering $K_{i,j} = FE.Rec(R'_{i,j}, hd_{i,j})$, and then calculate $r = r^* \oplus H(K \| t_u)$

and the decryption key $SK_{i,j} = H(r\|K_{i,j}\|ID_i\|GID\|t)$. Because PUF is inseparable from the microprocessor of the smart meter [13], $K_{i,j}$ and $SK_{i,j}$ cannot be obtained by other entities. In addition, random elements such as the one-time key $K_{i,j}$ and the random number r are used in the generation of session key $SK_{i,j}$, so that the confidentiality of session key is not affected by past or future session key leakage. Therefore, the proposed PGAC scheme can ensure the confidentiality and integrity of the session key, and ensure that the privacy of users is not leaked.

Theorem 3. *The proposed PGAC scheme can resist man-in-the middle attacks. That is, any malicious entity cannot deduce any information about metering data by intercepting the communication data transmitted during metering data collection, nor can it make the GW, SM_i $(i = 1, 2, ..., n)$ or CC accept tampered or forged communication data.*

Proof. According to Theorem 2, entities other than CC and SM_i cannot destroy the confidentiality of metering data M_i through sniffer attacks. When a malicious entity attempts to tamper with the intercepted tuple DT_1 or the aggregated response MES, a valid authentication code $AUTH_{CC-GW}$ or $AUTH_{GW-CC}$ must be generated. According to Theorem 2, only legitimate CC and GW can obtain the key $K_{GW,CC}$, thus no other entity can generate valid authentication codes $AUTH_{CC-GW}$ or $AUTH_{GW-CC}$. When a malicious entity attempts to tamper with the intercepted data tuple $DT_2 = (t, Req, IDEN_{CC-SMs})$, a valid authentication code $AUTH_{CC-i}$ $(i = 1, 2, ..., n)$ must be generated. By the same token, only legitimate CC and M_i $(i = 1, 2, ..., n)$ can obtain the key $K_{i,j}$, other entities cannot generate valid authentication code $AUTH_{CC-i}$. When a malicious entity attempts to tamper with the intercepted tuple $DT_{3,i}$, a valid authentication code $AUTH_{i-GW}$ must be generated. Similarly, only SM_i and the legitimate GW can obtain key $K_{i,GW}$, while other entities cannot generate valid $AUTH_{i-GW}$. Therefore, the proposed PGAC scheme can resist man-in-the-middle attacks.

Theorem 4. *The proposed PGAC construction is resistant to impersonation attacks. That is, any malicious entity cannot impersonate the identity of SM_i $(i = 1, 2, ..., n)$, the GW or CC without being detected.*

Proof. To impersonate CC, the malicious entity must compute a valid $AUTH_{CC-GW}$ and a valid $AUTH_{CC-i}$ $(i = 1, 2, ..., n)$. In addition, if the adversary tries to impersonate the GW, she/he must send a valid $AUTH_{GW-CC}$ to CC. Also, if the malicious entity tries to impersonate SM_i, she/he must compute a valid $AUTH_{i-GW}$ and $AUTH_{i-CC}$. According to Theorem 2, the malicious entity cannot compute valid $AUTH_{CC-GW}$, $AUTH_{CC-i}$, $AUTH_{GW-CC}$, $AUTH_{i-GW}$ and $AUTH_{i-CC}$. Thus, the proposed PGAC construction is resistant to impersonation attacks.

Theorem 5. *The proposed PGAC construction is resistant to replay attacks, which means any malicious entity cannot make CC, the GW or SM_i accept a data tuple that has been accepted before.*

Proof. In the proposed PGAC construction, the timestamp t is employed in generating data tuples DT_1 and DT_2, the timestamp t_i $(i = 1, 2, ..., n)$ are employed in generating data tuples $DT_{3,i}$, the timestamp t_{GW} is employed in generating the data tuple MES. Thus, when the data tuple DT_1, DT_2, DT_3 or MES is resent, it would be recognized as invalid according to the freshness of timestamps. Thus, the proposed PGAC construction can resist replay attacks.

Theorem 6. *The proposed PGAC construction is resistant to physical attacks. That is, the user cannot modify metering data by changing the configuration of smart meters.*

Proof. PUF is unclonable, and no entity can recreate the same PUF [25]. In addition, the PUF will lose its function after being tampered with. When PUF_{GW} and PUF_i are tampered with, the behaviors of the GW and SM_i will be changed [3]. That is, the PUF would not be able to correctly generate R'_{GW}, $R'_{i,j}$ and $R'_{i,GW}$, which implies the keys $K_{GW,CC} = FE.Rec(R'_{GW}, hd_{GW,CC})$, $K_{i,j} = FE.Rec(R'_{i,j})$ and $K_{i,GW} = FE.Rec(R'_{i,GW}, hd_{i,GW})$ could not be recovered. This causes SM_i and the GW to be unable to calculate valid authentication codes, which will be detected by the receivers. Thus, the proposed PGAC construction is resistant to physical attacks.

5.2 Functional Analysis

This section compares the functions implemented by Ding et al.'s scheme [6], Cao et al.'s scheme [2], Ren et al.'s scheme [24], and our PGAC scheme. Ding et al. [6] proposed an identity-based metering data aggregation scheme for the security of metering data collection in industrial smart grids. In [6], the collector can collect and aggregate metering data of users in their respective management domains, and supports the collector to perform batch verification of user signatures in the management domains, thus maintaining the confidentiality and integrity of metering data. However, the solution does not employ any authentication mechanism to verify the data collection request from the service provider before the smart meter performs the signature process.

Ren et al. [24] proposed a group authentication and data transmission scheme based on PUF to solve the large-scale concurrent access authentication problem in narrowband IoT. In [24], the data management server communicates with all narrowband IoT devices within the group through a specified gateway to perform efficient authentication and data transfer processes. However, the scheme does not use any authentication mechanism to verify the authenticity of the data request from the data management server before the gateway forwards the data request to the narrowband IoT devices in its region. In addition, when the gateway aggregates the responses of all narrowband IoT devices it manages, the scheme similarly does not employ any authentication mechanism to verify the authenticity of those responses.

Cao et al.'s scheme [2] mainly aims at the mutual identity authentication and data privacy protection problems faced by real-time data collection between

the control center and a single smart meter, and does not take into account the measurement data collection of large-scale smart meters at the same time. In our PGAC scheme, control center uses the gateway to collect metering data of all smart meters in the area and realize group authentication function. In addition, the PGAC scheme uses the gateway to verify the source of data collection request and ciphertext of metering data respectively in the forwarding phase and the aggregation phase, which improves the security of the system and reduces the waste of computing resources caused by invalid data to the control center and smart meters Table 1.

Table 1. Function comparison

Scheme	Forwarding authentication	Request authentication	Response authentication	Aggregate response	Group authentication	Batch authentication
[6]	×	×	√	√	×	√
[2]	×	√	√	×	×	×
[24]	×	√	×	√	√	×
PGAC	√	√	√	√	√	×

Notes: √ indicates support, × indicates no support.

5.3 Theoretical Comparision

In this section, the proposed PGAC scheme, Ding et al.'s scheme [6], Cao et al.'s scheme [2] and Ren et al.'s scheme [24] are analyzed theoretically and compared in terms of computational cost. Table 2 summarizes in detail the computing costs required by system setup phase, registration phase, request phase, forwarding phase, encryption phase, aggregation phase and decryption phase in these four schemes. Among them, the number of smart meters is n, T_{Mod} represents the operation time of a modular exponentiation operation, T_E represents the operation time of a bilinear pair operation, T_{Mp} represents the operation time of a multiplication operation, and T_{Log} represents the operation time of a discrete logarithm operation. T_H indicates the operation time of a hash operation, T_{Gen} indicates the operation time of a $FE.Gen$ algorithm, T_{Rec} represents the run time of a $FE.Rec$ algorithm, T_{PUF} denotes the run time of a PUF operation, and T_S indicates the run time of a encryption or decryption algorithm of symmetric encryption scheme Π.

Cao et al.'s scheme [2], Ren et al.'s scheme [24] and our PGAC scheme only carry out lightweight operations such as hash operation and PUF operation, while Ding et al.'s scheme [6] has complex bilinear pair operation and a large number of modular exponentiation operation, discrete logarithm operation and multiplication operation, thus it has more computational time costs.

Since Ren et al.'s scheme [24] assumes that PUF is idealized, it is not equipped with a fuzzy extractor, so the following analysis does not take into account the extra time spent on the $FE.Gen$ and $FE.Rec$ algorithms by Cao et al.'s scheme [2] and our PGAC scheme. For the system setup procedure, since

Table 2. Computing cost comparison

Phase	Scheme [6]	Scheme [2]	Scheme [24]	PGAC
System setup	$2T_{Mod} + T_H + T_E$	$\triangle$	T_{PUF}	$T_{PUF} + T_{Gen}$
Registration	$(n+1)T_H + (n+1)T_{Mod}$	$nT_{PUF} + nT_{Gen}$	nT_{PUF}	$2nT_{PUF} + 2nT_{Gen}$
Request	$\triangle$	nT_H	$T_{PUF} + (4n + 2)T_H$	$2nT_H$
Forwarding	–	–	$\triangle$	$T_{PUF} + T_{Rec} + T_H$
Encryption	$nT_H + 3nT_{Mod} + 2nT_{Mp}$	$nT_{PUF} + nT_{Rec} + 4nT_H + nT_s$	$nT_{PUF} + 5nT_H + nT_s$	$2nT_{PUF} + 2nT_{Rec} + 5nT_H + nT_s$
Aggregation	$(2n+1)T_H + (2n + 2)T_{Mod} + (6n-1)T_{Mp}$	–	$\triangle$	$(n+1)T_H$
Decryption	$T_{Mod} + 2T_H + 3T_{Mp} + T_E + nT_{Log}$	$3nT_H + nT_S$	$nT_H + nT_S$	$(2n+1)T_H + nT_S$

Notes: $\triangle$ denotes that the execution time of lightweight operations not defined in this phase, such as $\oplus$ and random number generation.

Cao et al.'s scheme [2] does not have an aggregator, the computing time of Cao et al.'s scheme [2] and Ren et al.'s scheme [24] can be regarded as consistent with our PGAC scheme. For the registration procedure, because Cao et al.'s scheme [2] and Ren et al.'s scheme [24] lack the identity authentication mechanism between the gateway and the smart meter, our PGAC scheme costs nT_{PUF} more than Ren et al.'s scheme [24]. For the request procedure, Ren et al.'s scheme [24] costs $T_{PUF} + (2n+2)T_H$ more than our PGAC scheme, and our PGAC scheme costs nT_H more than Cao et al.'s scheme [2].

For the forwarding procedure, our PGAC scheme costs $T_{PUF} + T_{Rec} + T_H$ more than Cao et al.'s scheme [2] and Ren et al.'s scheme [24], but Cao et al.'s scheme [2] does not have an aggregator, and the gateway in Ren et al.'s scheme [24] does not have the function of gateway to verify the identity of the control center. For the encryption procedure, Ren et al.'s scheme [24] costs nT_H more than Cao et al.'s scheme [2], and our PGAC scheme costs $nT_{PUF} + nT_{Rec} + nT_H$ more than Cao et al.'s scheme [2]. However, Cao et al.'s scheme [2] and Ren et al.'s scheme [24] lack the function of gateway to verify the identity of each smart meter. For the aggregation procedure, our PGAC scheme costs $(n+1)T_H$ more than Cao et al.'s scheme [2] and Ren et al.'s scheme [24], but Cao et al.'s scheme [2] does not have the aggregation function and Ren et al.'s scheme [24] does not realize the function of gateway to verify the identity of each smart meter. For the decryption procedure, Cao et al.'s scheme [2] costs $(n-1)T_H$ more than our PGAC scheme, and our PGAC scheme costs $(n+1)T_H$ more than Ren et al.'s scheme [24].

To sum up, compared with Ding et al.'s scheme [6], our PGAC scheme has obvious advantages in computational time cost, but spends more on computational time cost than Cao et al.'s scheme [2] and Ren et al.'s scheme [24]. However, our PGAC scheme implements more comprehensive authentication function and only rely on lightweight operations.

5.4 Experimental Performance

In this section, the experimental performance of the proposed PGAC system is compared with Cao et al.'s scheme [2] and Ren et al.'s scheme [24]. This section uses the Python HashLib module to implement the SHA-256 algorithm and the Crypto library to implement the 256-bit AES-CBC encryption algorithm. The simulation experiments are run on Microsoft Windows 11 operating system with Intel(R) Core(TM)i7-11800H CPU@2.30 GHz and 32GB RAM. Moreover, the simulation results of a 128-bit arbiter PUF circuit on an MSP430 microcontroller machine with a CPU of 798MHz in [15] are used to evaluate the performance of PUF operations. The proposed PGAC system uses the BCH code migration mechanism [7] to implement the FE.Gen and FE.Rec operations of fuzzy extractor, where T_{Gen} is 1.17ms and T_{Rec} is 3.28ms. Table 3 shows the parameter length in the simulation.

Table 3. Parameter length

Parameter	Size (bits)
Number of user n	500
Challenge C	128
Response R	128
Shared key K	128
Identity code ID	128
Random number r	128
Timestamp t	64
Request information Req	512
Metering data M	2048

This section sets up a group of 500 users and uses the same metering data to fairly test the performance of each scheme. Figure 2 shows the computing time of Cao et al.'s scheme [2], Ren et al.'s scheme [24] and our PGAC scheme in the five procedures of request, forwarding, encryption, aggregation and deception. In the request procedure, Cao et al.'s scheme [2] takes about one-third as long as Ren et al.'s scheme [24], and our PGAC scheme takes about half as long as Ren et al.'s scheme [24]. In the forwarding procedure, only the PGAC scheme has the capability of the gateway authentication request and costs 3.402ms. In the encryption and aggregation procedure, Ren et al.'s scheme [24] did not set a fuzzy extractor, so it took about twice as long as Cao et al.'s scheme [2], while our PGAC scheme designed a mutual authentication mechanism with gateway, so it took about twice as long as Cao et al.'s scheme [2] and Ren et al.'s scheme [24]. In the decrption procedure, Ren et al.'s scheme [24] and our PGAC scheme perform group authentication in about half the time of Cao et al.'s scheme [2]. The proposed PGAC scheme takes more total time than the other two ones, but implements a more complete authentication mechanism.

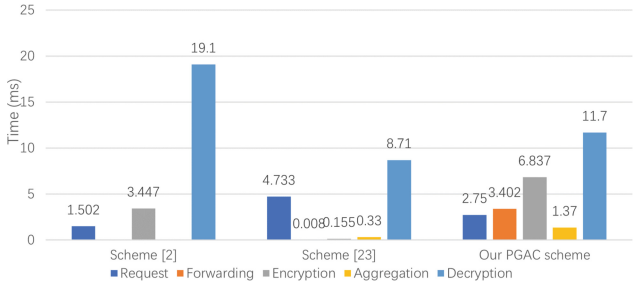

Fig. 2. Comparison of execution times for each procedure of the communication process.

Figure 3 shows the total number of messages sent by Cao et al.'s scheme [2], Ren et al.'s scheme [24], and our PGAC scheme in the four procedures of request, forwarding, encryption, and aggregation. In Ren et al.'s scheme [24] and our PGAC scheme, only one message is sent between the control center and a group of smart meters in mutual communication, while 500 messages are sent between smart meters and control center in Cao et al.'s scheme [2].

Figure 4 shows these three schemes to send the ciphertext response to the control center or data management server. Ren et al.'s scheme [24] and our PGAC scheme have basically the same size of data transmitted, while Cao et al.'s scheme [2] has at least 42.41 KB more data than the two schemes. Figure 3 and Fig. 4 show the superiority of group authentication mechanism in the large-scale user data collection scenario of smart grid. This mechanism can effectively reduce the number of message transmission and shorten the total length of transmitted data to a certain extent, thus achieving the design goal of controlling the bandwidth pressure of the main communication network.

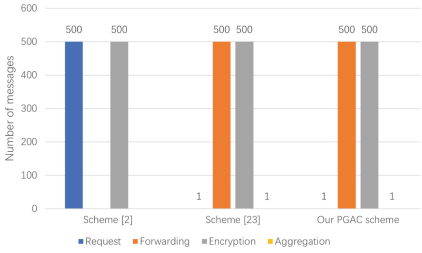

Fig. 3. Number of messages sent in the request, forward, encrypt and aggregate phases.

Then, the computing time of control center, gateway and smart meter in PGAC scheme is analyzed when the number of group users is 100 through 1000. Because the calculation of a single smart meter has nothing to do with the number of groups, the smart meter has maintained a time cost of about 6.837ms. Figure 5 shows the time spent by the control center and gateway in different group users. As shown in Fig. 5, the computing time of the control center and gateway increases linearly with the number of group users.

Fig. 4. Send data volume to the control center or data management server.

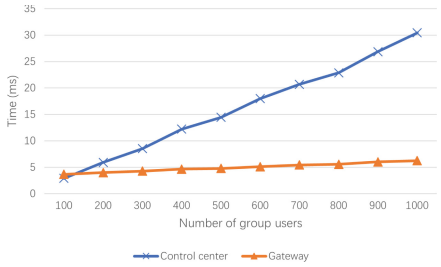

Fig. 5. Time overhead of the control center and gateway.

6 Conclusion

In order to ensure the confidentiality and integrity of metering data and the high efficiency of communication in the process of data acquisition of a large number of users, a multi-user data acquisition scheme (PGAC) based on PUF is proposed in smart grid. The PGAC scheme realizes the group authentication process of the control center for smart meters and the mutual identity authentication of the communication among the control center, gateway and smart meters. The security analysis shows that the PGAC scheme can resist the traditional attacks and physical attacks, and user privacy can be prevented from disclosure. Based on the results of functional analysis, theoretical analysis and efficiency analysis, the proposed PGAC scheme has more advantages in computing cost and communication overhead compared with other schemes.

Acknowledgments. This article is supported in part by the Guangxi Natural Science Foundation under grant 2019GXNSFGA245004 and 2023GXNSFAA026236, the National Natural Science Foundation of China under projects 61962012 and 62303037, the Zhejiang Soft Science Research Program under grant 2023C35081, and the special fund of the High-level Innovation Team and Outstanding Scholar Program for universities of Guangxi.

References

1. Cao, Y.N., Wang, Y., Ding, Y., Guo, Z., Wu, Q., Liang, H.: Blockchain-empowered security and privacy protection technologies for smart grid. Comput. Stand. Interf. **85**, 103708 (2023)
2. Cao, Y.N., Wang, Y., Ding, Y., Zheng, H., Guan, Z., Wang, H.: A puf-based lightweight authenticated metering data collection scheme with privacy protection in smart grid. In: 2021 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BDCloud/SocialCom/SustainCom), New York City, NY, USA, September 30 - Oct. 3, 2021, pp. 876–883. IEEE (2021)
3. Delavar, M., Mirzakuchaki, S., Ameri, M., Mohajeri, J.: Puf-based solutions for secure communications in advanced metering infrastructure (ami): Puf-based solutions for secure communications in ami. Int. J. Commun. Syst. **30**, e3195 (10 2016)
4. Delvaux, J., Peeters, R., Gu, D., Verbauwhede, I.: A survey on lightweight entity authentication with strong pufs. ACM Comput. Surv. **48**(2) (oct 2015)
5. Devadas, S., Suh, E., Paral, S., Sowell, R., Ziola, T., Khandelwal, V.: Design and implementation of puf-based "unclonable" rfid ics for anti-counterfeiting and security applications. In: 2008 IEEE International Conference on RFID, pp. 58–64 (2008)
6. Ding, Y., Wang, B., Wang, Y., Zhang, K., Wang, H.: Secure metering data aggregation with batch verification in industrial smart grid. IEEE Trans. Industr. Inf. **16**(10), 6607–6616 (2020)
7. Dodis, Y., Ostrovsky, R., Reyzin, L., Smith, A.: Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. SIAM J. Comput. 38(1), 97–139 (2008)
8. Dodis, Y., Reyzin, L., Smith, A.: Fuzzy extractors: how to generate strong keys from biometrics and other noisy data. In: Cachin, C., Camenisch, J.L. (eds.) Advances in Cryptology - EUROCRYPT 2004, pp. 523–540. Springer, Berlin Heidelberg, Berlin, Heidelberg (2004)
9. Fan, H., Liu, Y., Zeng, Z.: Decentralizing privacy-preserving data aggregation scheme using blockchain in smart grid. In: Security and Privacy in Digital Economy: First International Conference, SPDE 2020, Quzhou, China, October 30-November 1, 2020, Proceedings 1, pp. 131–142. Springer (2020)
10. Fang, L., Li, M., Liu, Z., Lin, C., Ji, S., Zhou, A., Susilo, W., Ge, C.: A secure and authenticated mobile payment protocol against off-site attack strategy. IEEE Trans. Dependable Secure Comput. **19**(5), 3564–3578 (2022). https://doi.org/10.1109/TDSC.2021.3102099
11. Gope, P., Sikdar, B.: Privacy-aware authenticated key agreement scheme for secure smart grid communication. IEEE Trans. Smart Grid **10**(4), 3953–3962 (2018)
12. Guan, Z., Zhang, Y., Zhu, L., Wu, L., Yu, S.: Effect: an efficient flexible privacy-preserving data aggregation scheme with authentication in smart grid. SCIENCE CHINA Inf. Sci. **62**, 1–14 (2019)
13. Guilley, S., Pacalet, R.: Soc security: a war against side-channels. Annals of Telecommunications-annales des télécommunications (2004)
14. Guo, Z., Qin, B., Guan, Z., Wang, Y., Zheng, H., Wu, Q.: A high-efficiency and incentive-compatible peer-to-peer energy trading mechanism. IEEE Transactions on Smart Grid, pp. 1–1 (2023)

15. Herder, C., Yu, M.D., Koushanfar, F., Devadas, S.: Physical unclonable functions and applications: a tutorial. Proc. IEEE **102**(8), 1126–1141 (2014)
16. Kaur, T., Kaur, M.: Cryptographic key generation from multimodal template using fuzzy extractor. In: 2017 Tenth International Conference on Contemporary Computing (IC3), pp. 1–6 (2017)
17. Khan, A.A., Kumar, V., Ahmad, M., Rana, S.: Lakaf: lightweight authentication and key agreement framework for smart grid network. J. Syst. Architect. **116**, 102053 (2021)
18. Kumar, N., Aujla, G.S., Das, A.K., Conti, M.: Eccauth: a secure authentication protocol for demand response management in a smart grid system. IEEE Trans. Industr. Inf. **15**(12), 6572–6582 (2019)
19. Li, X., et al.: A privacy-preserving lightweight energy data sharing scheme based on blockchain for smart grid. In: Gao, H., Wang, X., Wei, W., Dagiuklas, T. (eds.) Collaborative Computing: Networking, Applications and Worksharing - 18th EAI International Conference, CollaborateCom 2022, Hangzhou, China, October 15–16, 2022, Proceedings, Part II. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, vol. 461, pp. 91–110. Springer (2022). https://doi.org/10.1007/978-3-031-24386-8_6
20. Liu, S., Zhang, Q., Liu, H.: Privacy protection of the smart grid system based on blockchain. J. Phys.: Conf. Series **1744**(2), 022129 (Feb 2021)
21. Lu, W., Ren, Z., Xu, J., Chen, S.: Edge blockchain assisted lightweight privacy-preserving data aggregation for smart grid. IEEE Trans. Netw. Serv. Manage. **18**(2), 1246–1259 (2021)
22. M, N.B., Pushparajesh, V.: Review of internet of things: distributed power in smart grid. IOP Conf. Series: Mater. Sci. Eng. **1055**(1), 012139 (feb 2021)
23. Pappu, R., Recht, B., Taylor, J., Gershenfeld, N.: Physical one-way functions. Science **297**(5589), 2026–2030 (2002)
24. Ren, X., Cao, J., Ma, M., Li, H., Zhang, Y.: A novel puf-based group authentication and data transmission scheme for nb-iot in 3g pp 5g networks. IEEE Internet Things J. **9**(5), 3642–3656 (2022)
25. Tuyls, P., Batina, L.: Rfid-tags for anti-counterfeiting. In: Topics in Cryptology-CT-RSA 2006: The Cryptographers' Track at the RSA Conference 2006, San Jose, CA, USA, February 13–17, 2005. Proceedings. pp. 115–131. Springer (2006). https://doi.org/10.1007/11605805_8
26. Zeng, X., Liu, Q., Huang, H., Jia, X.: A lightweight privacy-preserving scheme for metering data collection in smart grid. In: 2017 IEEE 18th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM), pp. 1–6 (2017)
27. Zhang, L., Zhao, L., Yin, S., Chi, C.H., Liu, R., Zhang, Y.: A lightweight authentication scheme with privacy protection for smart grid communications. Futur. Gener. Comput. Syst. **100**, 770–778 (2019)