A Privacy-Preserving and Real-time Traceable Power Request Scheme for Smart Grid

Qingyou Yang[1], Jianan Hong[1], Kaiping Xue[1*], Weikeng Chen[1], Xiang Zhang[1], Hao Yue[2]
1. The Department of EEIS, University of Science and Technology of China, Hefei, Anhui 230027 China
2. The Department of Computer Science, San Francisco State University, San Francisco, CA 94132 USA
*kpxue@ustc.edu.cn

***Abstract*—Smart grid facilitates reliable and efficient power generation and transmission by integrating information and communication technologies. By collecting users' power demands in advance, the control center (power operator) can adjust the amount of electricity generated to reduce the excess power, which can increase the profit of the power operator. However, on the one hand, user's privacy becomes a critical issue, since it may leak out a user's life habits, which may make user's safety and belongings under threat. On the other hand, the system needs to arm the capability to avoid diverse adversaries' attacks and trace misbehaving users (who request power irresponsibly).**

In this paper, we propose a privacy-preserving and real-time traceable power request scheme to fulfill the security requirements. We utilize aggregator as a proxy between users and the control center, which verifies the messages and aggregates multiple users' requests together to preserve their privacy. More importantly, this privacy-preserving mechanism has no effect for the control center to whether charge each user, or trace the misbehaving users in real time. The performance analysis shows that our scheme is efficient in terms of computation and storage overhead.

***Index Terms*—Smart grid; power request; privacy preservation; traceability.**

I. Introduction

Smart grid is considered as the next generation power supply system [1], which integrates the traditional power grid with the modern information communication technology. With the communication networks, power generation and distribution can be implemented in a more reliable and efficient way. In order to reduce excess power generation, power request schemes are introduced to smart grid [2–8]: the control center periodically collects users' power demands, and determines the overall amount of electricity to be generated by the total power demands at every future moment. The power demands that users predict or project are sent by smart meters [8]. Several prediction methods have been proposed in [9, 10].

Naturally, the correctness of the collected power demands amount affects the load balancing for the power grid seriously. Therefore, it is necessary to confirm the power request messages with some effective authentication methods. Otherwise, a malicious unregistered user can launch an attack to abuse the system by sending fake request messages [5]. Besides the aspect of authentication, user's privacy should be preserved, as the leakage of user's power usage pattern in the request message will disclose the user's daily habits such as when residents leave houses [11]. Additionally, the control center has to trace the misbehaving registered users, as the smart meter is vulnerable to be compromised [12], and they may send power request messages irresponsibly, which will also affect the load balancing for smart grid seriously. Moreover, the traceability has to be real-time to recover a smart grid from imbalance to avoid further damages when it suffers.

Recently, a variety of schemes have been proposed to tackle part of the aforementioned security issues. Some literatures [2, 3] focus on solving the problems of authentication and privacy preservation, but ignore the importance of traceability for smart grid. Without traceability, registered users' misbehaviors cannot be traced by the control center. Furthermore, the schemes [4, 7, 8] provide the traceability function with the assumption of honest control center. And the credential-based schemes [5] and [6] provide a higher security but with a large amount of communication burden and complicated search. However, no existing power request scheme addresses the real-time traceability of misbehaving users with privacy preservation. Therefore, it is necessary to propose a real-time traceable and privacy-preserving power request scheme.

In this paper, we propose a privacy-preserving and real-time traceable power request scheme for smart grid. We utilize the aggregator to gather users' power request messages, which can achieve secure and privacy-preserving data forwarding. We also use the aggregator to cache request messages temporarily for the control center to trace users' misbehaviors of power request. Specifically, the major contributions of our proposed scheme are twofold.

1) We present a new power request scheme in smart grid, in which, a well-designed request aggregation mechanism is utilized to achieve the privacy preservation of users, and the real-time trace of misbehaving power request users, simultaneously. To the best of our knowledge, our scheme is the first to satisfy real-time traceability of misbehaving users without revealing users' privacy.
2) Considering the potential malicious power request from various users, an efficient authentication is implemented to prevent unauthorized entities from forging request messages; and authorized users' misbehaviors can also be reduced equipped with a feasible reward/penalty scheme.

The rest of this paper is organized as follows. We first introduce our network model and security requirements in

978-1-4673-8999-0/17/$31.00 ©2017 IEEE

Section II. After reviewing the Bilinear Pairing and Paillier Cryptosystem in Section III, we propose our power request scheme in Section IV, followed by the security analysis and performance evaluation in Section V and VI, respectively. Finally, we conclude the paper in Section VII.

II. Network Model and Security Requirements

In this section, we describe our network model and the security requirements.

A. Network model

The network model consists of three entities: Smart Meters (SM), Aggregators (AG) and Control Center (CC). Fig. 1 depicts the organization of these entities. Their functions and duties are described as follows:

- **Smart Meter (SM):** SM is an intelligent device which is usually installed in or out of user's house. It helps user to make the power usage plans and submits the plans included in request messages to the Aggregator. For clarity, we treat the smart meter and user as the same component.
- **Aggregator (AG):** AG is responsible for a certain residential area and connects with various SMs. It collects request messages from users periodically and caches for a certain period of time. Meanwhile, AG sends the aggregated message to control center for corrected power generation. In addition, AG aggregates users' periodic request power to Control center for charging and tracing at the end of each billing period or when facing severe supply-demand imbalance.
- **Control Center (CC):** CC connects with all of AGs. It collects the total requested power from each AG and then decides the optimal amount of electricity to be generated. It will also receive each individual user's total requested power over a certain period of time to find out misbehaving users or just for billing purpose.

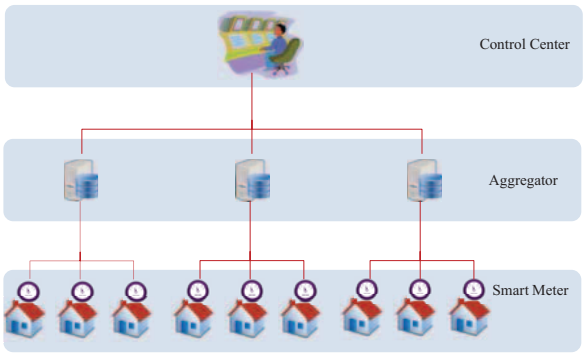


Fig. 1. Network Model for Smart Grid

B. Security requirements

In smart grid, the CC and AGs are placed in physically secure locations while SMs are installed around user's house, so it is reasonable to assume that CC and AGs are honest but curious, and will not be compromised by physical or network attacks. We regard the user as a dishonest component in smart grid, since the SM is vulnerable to be compromised by adversaries, and some users may deny the submitted requested power for paying less electricity bill.

We make four security requirements for the construction of secure and robust power request scheme in smart grid network, and they are summarized as follows:

1) **Message authentication:** Every message sent by SMs should be authenticated before it is aggregated by AG to resist against fake messages generated by adversaries.
2) **Privacy preservation:** No one except the user itself can know the user's power usage plan in detail. Even the CC and AGs cannot infer user's daily habits after they have collected lots of request messages.
3) **Non-repudiation:** A user cannot deny his/her requested power which has been submitted to AGs.
4) **Real-time traceability:** When the users' misbehaviors of power request have caused a severe supply-demand imbalance, tracing misbehaving users mechanism should be executed in real time to prevent further damage.

III. Preliminaries

Our scheme utilizes bilinear pairing [13] and Paillier cryptosystem [14], where, the former tool is used to construct authentication mechanism, and the latter has the property of additive homomorphism, that is implemented to operate data with privacy preservation.

A. Bilinear Pairing

Let $\mathbb{G}$ be a cyclic additive group of prime order q, P be a generator of $\mathbb{G}$, and $\mathbb{G}_T$ be a cyclic multiplicative group of the same order. Assume $\mathbb{G}$ and $\mathbb{G}_T$ are equipped with a pairing, i.e., a non-degenerated and efficiently computable bilinear map $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ such that $e(P, P) \neq 1_{\mathbb{G}_T}$, and $e(aP_1, bQ_1) = e(P_1, Q_1)^{ab} \in \mathbb{G}_T$ for all $a, b \in \mathbb{Z}_q^*$ and any $P_1, Q_1 \in \mathbb{G}$.

Definition I: A bilinear parameter generator $\mathcal{G}en$ is a probabilistic algorithm that takes a security parameter κ as input and outputs a 5-tuple$(q, P, \mathbb{G}, \mathbb{G}_T, e)$.

Definition II (Computational Diffie-Hellman (CDH) Assumption): Assume $a, b \in \mathbb{Z}_q^*$ are unknown, given elements $(P, aP, bP) \in \mathbb{G}$, it is difficult to compute $abP \in \mathbb{G}$.

B. Paillier Cryptosystem

Let $E(\cdot)$, $D(\cdot)$, m, and r be the notations of encryption function, decryption function, plaintext, and a random number in $\mathbb{Z}_n^*$, respectively. We choose two secure prime numbers p and q. Let $n = pq$ be RSA modulus. We then choose a random element $g \in \mathbb{Z}_{n^2}^*$. We calculate $\lambda = lcm(p-1, q-1)$, and $\mu = (L(g^\lambda \bmod n^2))^{-1} \bmod n$, where function $L(x) = (x-1)/n$. So the private key pr is $\{\lambda, \mu\}$, and the public key pu is $\{g, n\}$. The encryption is:

$$c = E(m) = g^m \cdot r^n \bmod n^2 \tag{1}$$

The decryption is:

$$m = D(c) = L(c^\lambda \bmod n^2) \cdot \mu \bmod n \tag{2}$$

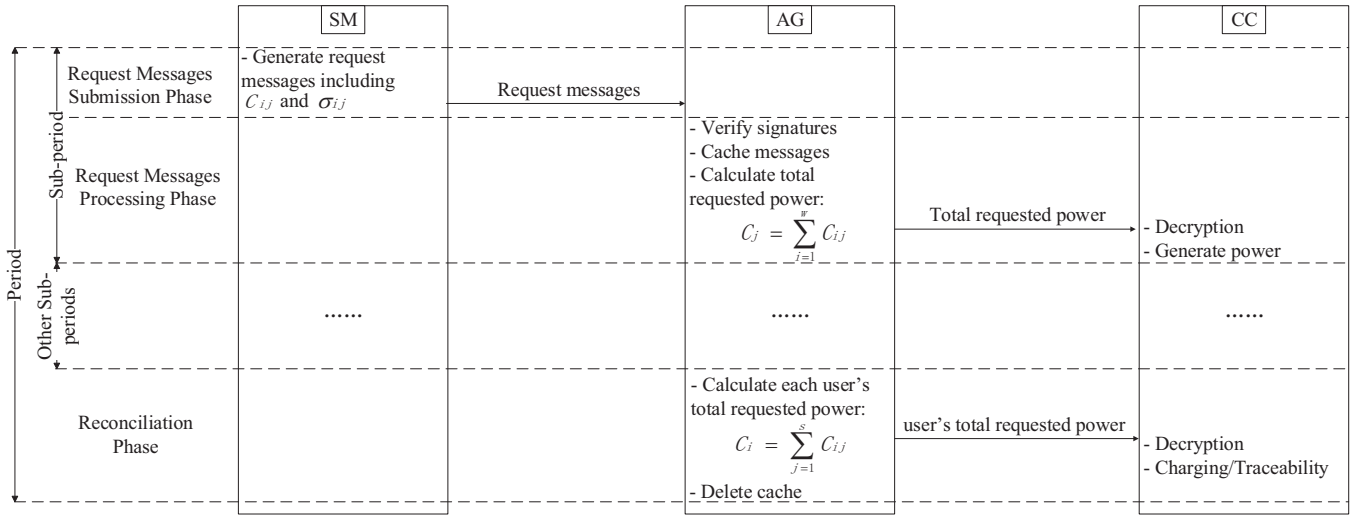


Fig. 2. Overview of Our Proposed Scheme

The additive homomorphism is shown as follows:

$$\begin{aligned} E(m_1) \cdot E(m_2) &= (g^{m_1} \cdot r_1^n)(g^{m_2} \cdot r_2^n) \bmod n^2 \\ &= g^{m_1+m_2} \cdot (r_1 r_2)^n \bmod n^2 \\ &= E(m_1 + m_2) \end{aligned} \tag{3}$$

IV. Proposed scheme

In this section, we first give an overview of our proposed scheme, and then present the details of our scheme including system initialization phase, request messages submission phase, request messages processing phase, and reconciliation phase.

A. Overview

A period in our scheme is split into s sub-periods, so that a user's accurate power usage in each sub-period is preserved, while the usage in one period can be calculated by CC for further traceability. As shown in Fig. 2, users submit their power demands in request messages submission phase. Request messages processing phase is executed to calculate total requested power by utilizing the property of homomorphic encryption, while the reconciliation phase is carried out at the end of each period, so that the CC can check each user's amount of requested power and actual amount of power usage. Specially, reconciliation phase will also be executed to trace misbehaving users when the severe supply-demand imbalance happens.

AG collects users' power requests in encrypted form C_{ij}, in which i represents the identity of the user, and j is the corresponding sub-period. For clarity, the requests are organized in a matrix in Fig. 3. Requests in a same column j belong to the same sub-period: AG aggregates them into C_j and sends to CC to help determine the optimal amount of power generation. Requests in a same row i belong to the same user: AG aggregates them into C_i and send to CC for charging and tracing.

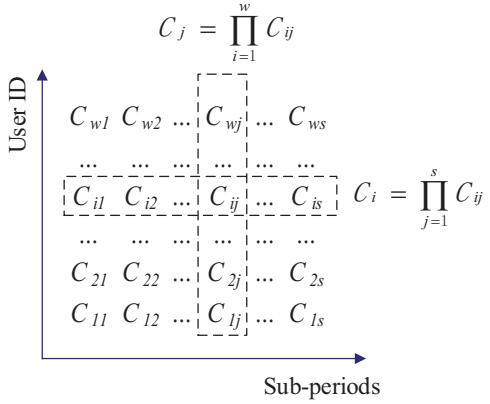


Fig. 3. Aggregator's Cached Data and Two Aggregation Operations

B. System Initialization Phase

The control center (CC) runs $\mathcal{G}en(\kappa)$ to generate the bilinear parameters $(q, P, \mathbb{G}, \mathbb{G}_T, e)$, so as the Paillier Cryptosystem's public key (n, g) and its corresponding private key (λ, μ). A secure hash function $H : \{0,1\}^* \to \mathbb{G}$ is selected. The system parameters are published as

$$pubs = \{q, P, \mathbb{G}, \mathbb{G}_T, e, H, g\} \tag{4}$$

The master key $\{\lambda, \mu\}$ is kept secretly by CC.

When a user U_i wants to register to the system, U_i first chooses a random number $x_i \in \mathbb{Z}_q^*$ as the private key, and then computes $Y_i = x_i P$ as the corresponding public key.

C. Request Messages Submission Phase

In order to submit user's power demand to CC, each user makes the power usage plan and submits it to CC ahead. Let m_{ij} be U_i's amount of requested power in the j-th sub-period. The following steps are proceeded in this phase by user's smart meter:

- *Encryption:* With a random $r_{ij} \in \mathbb{Z}_q^*$, the user's power demand for j-th sub-period m_{ij} is encrypted as:

$$C_{ij} = E(m_{ij}) = g^{m_{ij}} \cdot r_{ij}^n \tag{5}$$

- *Signature:* The user signs C_{ij}, with its identity U_i and timestamp TS:

$$\sigma_{ij} = x_i H(C_{ij} \| U_i \| TS) \tag{6}$$

- *Submission:* The request message $(C_{ij} \| U_i \| TS \| \sigma_{ij})$ is sent to the AG to complete this phase.

D. Request Messages Processing Phase

In this phase, once an AG receives a user's power request message. It first verifies the signature, then aggregates all its responsible users' requested power and sends it to CC. Detailed description is divided into four steps as follows:

- *Verification:* After receiving $(C_{ij} \| U_{ij} \| TS \| \sigma_{ij})$, the AG checks $e(P, \sigma_{ij}) \stackrel{?}{=} e(Y_i, H(C_{ij} \| U_i \| TS))$ with user's public key. If it holds, the user's message is accepted. Otherwise, AG discards it and requires the user to repeat the request messages submission phase. For the purpose of efficiency, this step can also be executed in a batch method as the formula (assume the total number of this AG's responsible users is w):

$$\begin{aligned} e(P, \sum_{i=1}^{w} \sigma_{ij}) &= e(P, \sum_{i=1}^{w} x_i H(C_{ij} \| U_i \| TS)) \\ &= \prod_{i=1}^{w} e(P, x_i H(C_{ij} \| U_i \| TS)) \\ &= \prod_{i=1}^{w} e(Y_i, H(C_{ij} \| U_i \| TS)) \end{aligned} \tag{7}$$

- *Cache:* AG caches valid power request messages for each individual user until the period finishes. Section VI will analyze the storage overhead and prove the rationality to deploy this mechanism in smart grid.
- *Aggregation:* After receiving and verifying all users' request messages, AG calculates the aggregated request message, e.g., the aggregated request message C_j for the j-th sub-period is as:

$$C_j = \prod_{i=1}^{w} C_{ij} \bmod n^2 \tag{8}$$

where C_j is the ciphertext of the total amount of requested power for the j-th sub-period. Fig. 3 provides a clear description for this aggregation. After that, AG signs the aggregated message in the same way as the user does and sends it to CC.

- *Generation:* After CC receives the encryption message C_j, CC performs the following calculation to decrypt it.

$$m_j = D(C_j) = L(C_j^{\lambda} \bmod n^2) \cdot \mu \bmod n \tag{9}$$

where m_j is the total power that CC needs to generate for the j-th sub-period, and the value equals to $\sum_{i=1}^{w} m_{ij}$ according to the addictive homomorphic property.

E. Reconciliation Phase

At the end of each whole period, or when the noticeable supply-demand imbalance happens, the CC implements the following steps for reconciliation:

1) At this moment, AG has collected all users' request messages under its management over the whole period. It sums up each user's amount of requested power separately as:

$$C_i = \prod_{j=1}^{s} C_{ij} \bmod n^2 \tag{10}$$

Fig. 3 provides a clear description to aggregate user U_i's power amount into C_i. Each C_i is sent to CC after aggregation.

2) The CC decrypts the encrypted message with its private key (λ, μ) and compares the requested power with the corresponding actual power usage, which has two purposes:
   a) *Charging:* If the requested power roughly matches the actual power usage, CC can provide a discount/reward to the user. Otherwise, an additional fee may apply.
   b) *Tracing misbehaving users:* If user's actual power consumption is completely different with his/her amount of requested power, which means the user has misbehaved in requesting power. Penalty thus is imposed on him/her, such as insulating the user from the system for security consideration.
3) Once a user doubts his/her amount of requested power, AG first sends this user's cached power request messages to a law authority who is fully trusted (e.g., local police office). Then the law authority verifies these messages, and acts as AG to aggregates the messages, and then uses CC's private key to decrypt the message to see whether the electricity bill is correct.
4) AG deletes all cached messages to complete this phase.

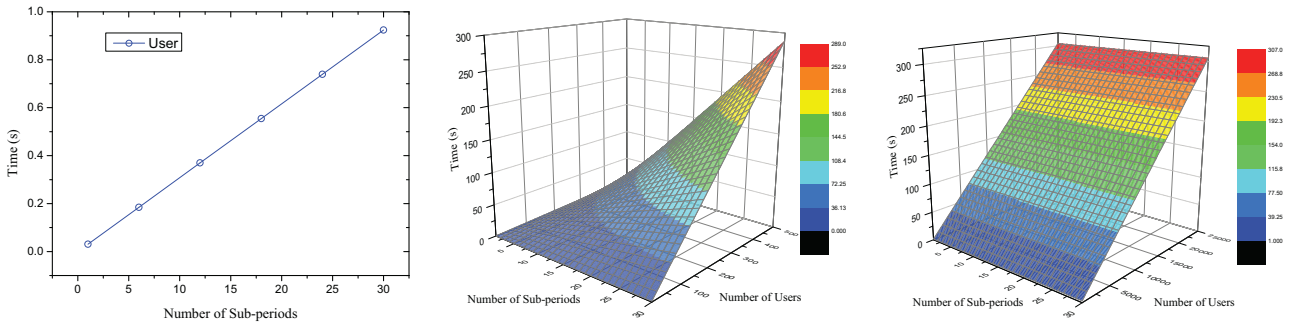
V. Security Analysis

In this section, we analyse the security properties of our proposed scheme. Especially, we focus on demonstrating how our proposed scheme can achieve the security requirements of message authentication, privacy preservation, non-repudiation, and real-time traceability.

- **Message authentication:** In the request messages submission phase (Section IV-C), the signature σ_{ij} is signed with U_i's private key x_i as Eq. (6), which is only mastered by the user. Any entity can verify σ_{ij} with user's public key Y_i. The correctness of verification is as follows:

$$\begin{aligned} e(P, \sigma_{ij}) &= e(P, x_i H(C_{ij} \| U_i \| TS)) \\ &= e(x_i P, H(C_{ij} \| U_i \| TS)) \\ &= e(Y_i, H(C_{ij} \| U_i \| TS)) \end{aligned} \tag{11}$$

Following the CDH assumption, any entity without U_i's private key x_i cannot forge a feasible σ_{ij} with non-negligible probability. The hash function (see (6)) with



(a) Computation Cost of User (b) Computation Cost of AG (c) Computation Cost of CC

Fig. 4. Computation Cost of Each Component in Our Scheme

timestamp TS inside can effectively resist the potential reply attack. In addition, batch verification in (7) helps to efficiently verify a number of signatures if none of them is illegal.

- **Privacy preservation:** The privacy issue is analyzed from two perspectives, the AG and CC. As AG and malicious users have no knowledge of the private key (λ, μ), they cannot access the plaintext of power request m_{ij}, although the AG has the capability to conduct aggregation operations on them. Thus the security property of Paillier Cryptosystem [14] prevents AG from gaining the privacy of power request amount. On the other side, although CC can unfold the ciphertexts, the received messages are aggregated according to (8) and (10) with the method shown in Fig. 3. This mechanism helps CC to get accurate total amount of generated power and power amount of each user's long-term power request C_i, without leaking an individual user's detailed requested power.
- **Non-repudiation:** During each period, AG not only caches C_{ij}, which contains the amount of requested power, but also caches the signature σ_{ij} generated by the user in every sub-period. When a user wants to deny his/her submitted power plans, the mechanism in Section IV-E can provide a powerful evidence to the law authority to judge which entity has lied.
- **Real-time traceability:** Let M_{ij} and E_{ij} be U_i's requested power and actual power consumption respectively in j-th sub-period ($i = 1, 2, ..., w$. $j = 1, 2, ..., s$), T_1 and T_2 be threshold values respectively for the maximum of acceptable supply-demand deviation for a sub-period and a user. This restriction can be represented by the inequality (12) and (13), while the threshold values T_1 and T_2 are decided by the control center and written in the agreement.

$$|\sum_{i=1}^{w}(M_{ij} - E_{ij})| \leq T_1 \text{ (for a sub-period)} \qquad (12)$$

$$|\sum_{j=1}^{s}(M_{ij} - E_{ij})| \leq T_2 \text{ (for a user)} \qquad (13)$$

Ideally, normal users' request power is equal to their actual power usage, which means $|M_{ij} - E_{ij}| = 0$ for all i, j. Thus, both T_1 and T_2 should be set to 0 for tracing misbehaving users (who don't satisfy (13)). Assume a supply-demand imbalance occurs in s-th sub-period, that is, $|\sum_{i=1}^{w}(M_{ij} - E_{ij})| > 0$ when $j = s$. According to the "Pigeonhole principle" [15], we can infer there must be at least one misbehaving user can be found by verifying inequality (13) for every user. With the reward/penalty mechanism, misbehaviors can be reduced in real life. Although, in real life, random deviations exist between request power and actual usage, proper value for T_1 and T_2 can also be selected to achieve the traceability of misbehaving power request users according to the actual requirements.

VI. Performance Evaluation

In this section, we evaluate the computation and storage overhead in the proposed scheme.

A. Computation Overhead

In order to evaluate the computation overhead of our proposed scheme, we conduct experiments to show the time costs of a period for the CC, AG and user, respectively. Our experiments are based on the PBC [16] and MIRACL [17] libraries running on a 3.0 GHz-processor computer, with 1,024-bits RSA modulus ($|n| = 1,024$) and 160-bits $\mathbb{G}$. For clarity, we assume there is only one AG with w users under its management. And we denote the number of sub-periods for each period as s. The experiment results are described as the following three parts:

- *Cost of user*: We depict the variation of time cost of a user in terms of the number of sub-periods for a period s in Fig. 4(a). It should be noted that smart meter is low-capacity-computing device [12], and the evaluation shown in Fig. 4(a) illustrates that our scheme brings in quite little burden upon smart meters.

- *Cost of AG*: We plot the cost of an AG in terms of the number of users w and the number of sub-periods s, as shown in Fig. 4(b). If we set a period to be a month, and users submit request messages daily, even an AG is responsible for 500 users, it only costs about 5 minutes to complete all operations for every month.
- *Cost of CC*: The computation cost of CC is depicted in Fig. 4(c). Although, the time cost of CC increases with the increase of the number of users w and the number of sub-periods s. The cost cannot cause a obvious computation burden on CC, even when $w = 20,000$ and $s = 30$, CC only needs to cost approximately 4 minutes per month.

B. Storage Overhead

Our proposed scheme requires AG to cache its responsible users' request messages until the reconciliation phase is done. The form of request message is $C_{ij}\|U_i\|TS\|\sigma_{ij}$, so its size is $S_{ij} = |C_{ij}| + |U_i| + |TS| + |\sigma_{ij}|$. Let n be 1,024 bits, $\mathbb{G}$ be 160 bits, both $|U_i|$ and $|TS|$ be 50 bits. Thus $S_{ij} = 2,048 + 50 + 50 + 160 = 2,308$ bits. According to these data, we depict the variation of an AG's total storage cost over a period of time in terms of the amount of users and the number of sub-periods for a period in Fig. 5. As shown in Figure, we

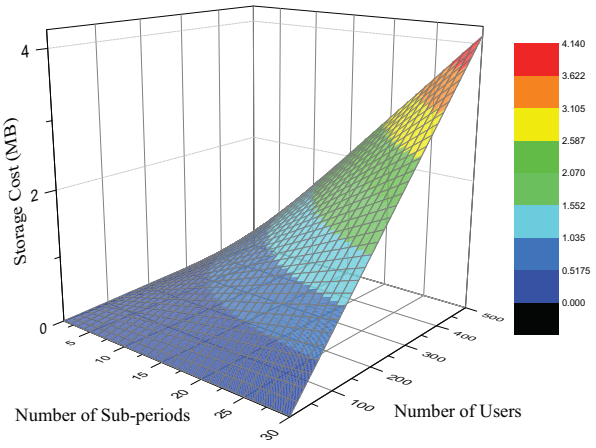


Fig. 5. Storage Cost of Our Scheme

can see even an AG is responsible for 500 users, it just costs about 4MB memory to cache all the request messages when the number of sub-periods in one period is 30. The cost of storage can be negligible.

The above analysis show that our scheme which has a reasonable computation overhead and a small storage overhead, is efficient and feasible for power request in smart grid.

VII. Conclusion

In this paper, we propose a privacy-preserving power request scheme with real-time traceability for smart grid. This scheme preserves users' privacy of living habits without losing the system's traceability and the accuracy of power generation: Any entities in the system, whether control center or aggregators cannot obtain the detailed power usage pattern of each user. With the misbehavior tracing mechanism and reward/penalty method, malicious power request is prevented in real life, which ensures the robustness and sustainability of smart grid system. The performance evaluation shows that our scheme is efficient and practical in terms of the overhead of computation and storage.

VIII. Acknowledgment

This work is supported by the National Natural Science Foundation of China under Grant No. 61379129 and No. 61671420, Youth Innovation Promotion Association CAS, and the Fundamental Research Funds for the Central Universities.

References

[1] H. Farhangi, "The path of the smart grid," *IEEE Power and Energy Magazine*, vol. 8, no. 1, pp. 18–28, 2010.

[2] D. Seo, H. Lee, and A. Perrig, "Secure and Efficient Capability-Based Power Management in the Smart Grid," in *Proceedings of the 9th International Symposium on Parallel and Distributed Processing with Applications Workshops*, 2011, pp. 119–126.

[3] H. Li, X. Lin, H. Yang, X. Liang, R. Lu, and X. Shen, "EPPDR: An efficient privacy-preserving demand response scheme with adaptive key evolution in smart grid," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 8, pp. 2053–2064, 2014.

[4] T. W. Chim, S. M. Yiu, L. C. K. Hui, and V. O. K. Li, "PASS: Privacy-preserving authentication scheme for smart grid network," in *Proceedings of 2011 IEEE International Conference on Smart Grid Communications*, 2011, pp. 196–201.

[5] J. C. L. Cheung, T. W. Chim, S. M. Yiu, L. C. K. Hui, and V. O. K. Li, "Credential-Based Privacy-Preserving Power Request Scheme for Smart Grid Network," *proceedings of 2011 IEEE Global Telecommunications Conference - GLOBECOM 2011*, pp. 1–5, 2011.

[6] T. Chim, S. Yiu, L. Hui, and V. Li, "Privacy-preserving advance power reservation," *IEEE Communications Magazine*, vol. 50, no. 8, pp. 18–23, 2012.

[7] C.-M. Yu, C.-Y. Chen, S.-Y. Kuo, and H.-C. Chao, "Privacy-Preserving Power Request in Smart Grid Networks," *IEEE Systems Journal*, vol. 8, no. 2, pp. 441–449, 2014.

[8] T. W. Chim, S.-M. Yiu, V. O. K. Li, L. C. K. Hui, and J. Zhong, "PRGA: Privacy-Preserving Recording & Gateway-Assisted Authentication of Power Usage Information for Smart Grid," *IEEE Transactions on Dependable and Secure Computing*, vol. 12, no. 1, pp. 85–97, 2015.

[9] R. E. Edwards, J. New, and L. E. Parker, "Predicting future hourly residential electrical consumption: A machine learning case study," *Energy and Buildings*, vol. 49, pp. 591–603, 2012.

[10] D. Lachut, N. Banerjee, and S. Rollins, "Predictability of energy use in homes," in *proceedings of Green Computing Conference (IGCC), 2014 International*. IEEE, 2014, pp. 1–10.

[11] H. Khurana, M. Hadley, Ning Lu, and D. Frincke, "Smart-grid security issues," *IEEE Security & Privacy Magazine*, vol. 8, no. 1, pp. 81–85, 2010.

[12] F. M. Tabrizi, "A Model for Security Analysis of Smart Meters," in *Proceedings of the 42nd International Conference on Dependable Systems and Networks Workshops (DSN-W)*, 2012, pp. 1–6.

[13] D. Boneh and M. Franklin, "Identity-Based Encryption from the Weil Pairing," *SIAM Journal on Computing*, vol. 32, no. 3, pp. 586–615, 2003.

[14] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," in *Advances in Cryptology - EUROCRYPT 99*, vol. 1592, 1999, pp. 223–238.

[15] W. A. Trybulec, "Pigeon hole principle," *Journal of Formalized Mathematics*, vol. 2, no. 199, p. 0, 1990.

[16] B. Lynn, "PBC Library," http://crypto.stanford.edu/pbc/, 2012.

[17] "Multiprecision Integer and Rational Arithmetic c/c++ Library," http://www.shamus.ie/, 2012.