RESA: A Robust and Efficient Secure Aggregation Scheme in Smart Grids

Zhiyuan Sui(✉), Michael Niedermeier, and Hermann de Meer

University of Passau, Innstr. 43, 94032 Passau, Germany
suizhiyu@fim.uni-passau.de, {michael.niedermeier,demeer}@uni-passau.de

Abstract. In this paper, we indicate the increasing interests in providing network security and privacy in Smart Grids, and propose a novel usage data aggregation scheme. The proposed scheme combines multiple cryptosystems to achieve anonymity and multidimensional data aggregation without a trusted third party. In our approach, smart meters transmit usage reports through hop-by-hop communication. If the communication is delayed or fails at one hop, it is possible to reroute the traffic through another hop. Therefore, the robustness of grid communication networks is improved. Additionally, an aggregation tree is constructed in order to optimize the aggregation time. Finally, smart meters utilize a highly efficient hash-based message authentication code to ensure data integrity and identity authentication. Although some existing approaches can achieve similar security features, our scheme has lower computational cost according to performance analysis and experiments.

Keywords: Smart grids · Multidimensional aggregation · Privacy preservation · Robustness · Security · Optimization

1 Introduction

Smart Grids have been gaining more popularity from both academia and industry because of the required grid reliability and potential tremendous benefits they offer to consumers. Smart Grids provide two-way communication between the power provider and consumers. The consumers are equipped with intelligent usage measurement devices, called smart meters, which report usage data to the power provider periodically. According to the collected usage data, power providers define the electricity prices to achieve better monitoring, control and stability of the Smart Grid. The concept of Smart Grids is obviously highly beneficial to the demand management and energy efficiency. However, the challenges concerning cyber attacks also arise with the development of Smart Grids. The communication between the power provider and consumers must be authenticated and secured against modification and forgery. Any of those attacks could be fatal to the electricity grid, and must be detected and treated accordingly. Transmitting trustworthy energy usage reports is an important task, considering the privacy and efficiency challenges in Smart Grids [1,2]. Firstly, privacy is a

E. Rome et al. (Eds.): CRITIS 2015, LNCS 9578, pp. 171–182, 2016.
DOI: 10.1007/978-3-319-33331-1_14

primary concern from the consumers' point of view. The fine-granular energy consumption readings of a smart meter can be used to spy on and expose a victim's behavior. This in turn can lead to personalized advertisements or discrimination against a user who is negatively classified according to his power usage profile. Therefore, the usage data must be protected in Smart Grids. Secondly, the computational resources at the consumers' side are very limited. The consumers need real-time usage instructions from their power providers to behave accordingly [3], otherwise the balance between energy generation and consumption cannot be ensured.

Fine-granular data collection in Smart Grids has raised a large amount of privacy preservation and security questions. The power provider requires aggregated data to forecast consumption of households in the future. According to this scenario, Rongxing et al. [4] propose a privacy preserving usage data aggregation scheme based on the Paillier homomorphic crpytosystem [5]. In this scheme, a trusted party, who has the knowledge of the connection between the ciphertexts and their sources, is responsible for the collection of the usage reports from the smart meters. Compared to traditional symmetric encryption schemes, this scheme improves privacy preservation of the consumers. However, finding a trusted party is a challenging task in itself. Therefore, some anonymous authentication schemes were introduced for Smart Grids. Chia-Mu et al. [6] employ ring signature schemes to protect the profiles. However, the computational cost is increasing with the size of the ring. Liu et al. [7] employ blind signatures to anonymize the consumers' real identities. This scheme however cannot protect consumers' usage data profiles. Zhang et al. [8] construct a self-certified signature scheme and Sui et al. [9] construct an incentive-based anonymous authentication scheme to realize third party free schemes. Those schemes are based on anonymity networks, in which the sources of usage reports are anonymous. This makes it difficult to identify smart meter or communication failures. Li et al. [10,11] roughed out a possible framework for secure information aggregation in Smart Grids. Smart meters transmit the usage data in a hop-by-hop way. The operation center decrypts the aggregated data using its secret key. Therefore, neither a trusted party nor an anonymity network is necessary. However, it does neither describe how to construct the aggregation tree, nor how to ensure aggregation in cases of failure. Moreover, the computation of public key signatures is costly.

In this paper, we improve the system model introduced in [11] and propose a robust and efficient secure aggregation scheme for privacy preservation, named RESA. Inspired by the fact that the locations of smart meters are fixed and stable, the usage data can be aggregated using hop-by-hop communication. For one thing, a highly efficient symmetric encryption algorithm can be employed to guarantee data integrity and identity authentication instead of an asymmetric one. For another, anonymity can be achieved without a trusted party. Apart from that, the contributions of RESA include:

1. Firstly, RESA optimizes the aggregation tree model to minimize the aggregation time and realize the almost real-time power usage report during demand and response time.

2. Secondly, the Elgamal encryption algorithm and Chinese Remainder Theorem are utilized to achieve multidimensional usage data aggregation. The Elgamal encryption algorithm is more efficient than the Paillier cryptosystem [5]. Using the Chinese Remainder Theorem, usage data can be compressed. This reduces the computational cost. Additionally, using the Elgamal encryption algorithm, the usage data can be protected from unauthorized access.
3. Thirdly, RESA employs an efficient symmetric encryption algorithm to authenticate smart meters' identities and protect usage reports against modification. Therefore, RESA can achieve more efficient authentication than previous anonymous authentication schemes [6 – 10].
4. Finally, RESA improves the system security compared with previous work [11], in which the power provider can be informed in time if there is a smart meter failure or communication delay. However, it does not consider the trustworthiness in Smart Grids. RESA employs cheap tamper-resistant black boxes and cryptosystems [12,13] to improve the security of the system.

The remainder of this paper is structured as follows: In Sect. 2, the preliminaries, which are later on required in this paper, are explained in detail. Section 3 illustrates the system model as well as the security requirements in our Smart Grid system, while Sect. 4 describes our proposed scheme that features both anonymity and security. Section 5 discusses the security requirements and compares the computational and communicational performance of our scheme with previous works. Section 6 concludes this paper.

2 Preliminaries

Several important cryptography technologies, which are necessary to understand our work, are listed in this section.

2.1 Chinese Remainder Theorem

In RESA, the Chinese Remainder Theorem is employed to compress the multidimensional data into a single dimensional one. Therefore, the communication overhead is reduced. The Chinese Remainder Theorem is a statement about simultaneous congruences. Randomly choose pairwise co-prime integers $a_1, a_2, ..., a_l$. For any given sequence of integers $d_1, d_2, ..., d_l$, the congruence equations

$$\begin{cases} N = d_1 \bmod a_1 \\ N = d_2 \bmod a_2 \\ \quad ... \\ N = d_l \bmod a_l \end{cases} \tag{1}$$

have the only solution $N = d_1 A_1 A_1^{-1} + ... + d_l A_l A_l^{-1} \bmod A$, where $A = a_1 a_2 ... a_l$, $A = a_j A_j (1 \leq j \leq l)$, $A_j A_j^{-1} = 1 \bmod a_j (1 \leq j \leq l)$.

2.2 Hash-Based Message Authentication Code

Hash-based Message Authentication Code (HMAC) is a short piece of information to authenticate a message. The sender and the receiver communicate with each other by running a key establishment protocol (e.g. Diffie-Hellman protocol [13]) and generate a shared session key K for a predefined cryptographic hash algorithm $\mathcal{H}$ (e.g. SHA-1, MD5 [14]).

1. **Sending:** The sender computes the hash value of message m: HMAC= $\mathcal{H}(K, m)$. After that, the sender sends (m, HMAC) to the receiver.
2. **Verification:** Upon the receipt of the message and the HMAC tag (m, HMAC), the receiver computes HMAC= $\mathcal{H}(K, m)$, HMAC$' = \mathcal{H}(K, m)$ and compares HMAC to HMAC$'$. If they are equal, the HMAC is valid.

2.3 EC-Elgamal Encryption Scheme

The EC-Elgamal encryption scheme [15], which is based on the elliptic curve discrete log assumption, is equivalent to the original Elgamal scheme and can achieve the additive homomorphic properties with less computation overhead than [5]. The EC-Elgalmal encryption scheme is comprised of three algorithms: key generation, encryption and decryption.

1. **Key Generation:** Input a random security parameter κ, output a prime p of size κ. Generate an additive group $\mathbb{G}$ over a finite field $\mathbb{Z}_p^*$, and select a generator P of $\mathbb{G}$. Then, randomly pick $x \in \mathbb{Z}_p^*$ as the secret key. After that, select a mapping pair $\mathcal{M} \leftarrow \mathbb{G}, \mathcal{M}' \leftarrow Z$ used to map values into points on curve. Finally, compute $P_{pub} = \gamma P$ and publish (q, P, P_{pub}) as the public parameters.
2. **Encryption:** Given data d, generate a group element $m = \mathcal{M}(d)$, randomly choose an integer r and output the ciphertext $(c_1 = rP,\ c_2 = rP_{pub} + m)$.
3. **Decryption:** Given the ciphertext (c_1, c_2), the group element can be decrypted as: $m = (c_2 - xc_1)$ and the corresponding data is $d = \mathcal{M}'(m)$.

3 System Model and Requirements

3.1 System Model

In RESA, we adopt the aggregation tree model [11], where the usage data is transmitted by hop-by-hop communication, as shown in Fig. 1. The system model mainly consists of two entities: the electricity utility (EU) and the smart meter (SM). We assume that each EU communicates with multiple SMs in a concrete area, and the number of SMs is large enough for each SM to cloak its usage data by data aggregation techniques.

1. **SM:** The SM is the energy consumption reporting device present at each consumer's site. But consumers cannot learn or modify the secret knowledge of SMs, which are assumed to be resistant against tampering. The SMs, which form a tree construction regularly, report consumers' encrypted energy usages

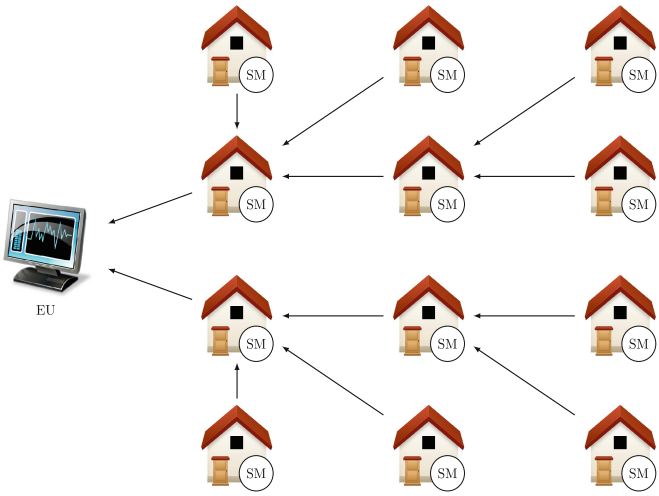

Fig. 1. Network model

to the EU using hop-by-hop communication. Therefore, no one can link the usage report to its source. Once the SM cannot receive the usage data from its neighbor SM in a predefined time, it can determine that the communication needs to be fixed.

2. **EU:** The EU is an infrastructure that is controlled by the power provider and is in charge of the SMs in a concrete area. It collects and analyzes the usage data from SMs periodically, and broadcasts consumption related instructions or electricity prices to customers according to the aggregated data. In our scheme, the EU is assumed as honest but curious. It acts according to the prescribed actions, but is curious to the consumers' energy usage.

3.2 Security Requirements

Because the EU makes decisions according to SMs' usage reports, adversaries might try to send fake usage reports to misinform the EU. Besides the security issues, the adversary might also be interested in consumers' usage profiles. They will try to eavesdrop the communication data or intrude the EU's database to steal data. In our aggregation scheme, the main aim is to ensure trustworthiness of the data from both EU and SMs while ensuring the privacy of legitimate users:

1. **Trustworthiness:** The adversary is not able to modify the consumption reports from SMs, nor the instruction from the EU without authorization (data integrity). The usage reports and the consumption instruction can be determined whether they derive from a legitimate source (authentication).
2. **Privacy Preservation:** The EU cannot learn individual SM's usage data (unconditional anonymity). Even if an adversary eavesdrops the communication among SMs, it cannot get the consumers' usage data without the help of the EU (confidentiality).

3.3 Design Goal

Under the aforementioned system model and security requirements, we design a robust and efficient aggregation scheme, which satisfies the following goals simultaneously:

1. **Optimal Aggregation:** The EU and SMs need almost real-time communication to balance the electricity consumption and generation in the grid. Due to the limited computational resources at the consumers' side, it is necessary to optimize the aggregation time.
2. **Security Requirements:** As stated above, the security requirements should be achieved to resist malicious behaviors, which would lead to fake information injection or consumers' privacy leakage.
3. **Robust Communication:** Missing usage data also leads to misinformation. Security mechanisms can ensure that accepted usage reports are from legitimate sources, but they cannot guarantee that all the legitimate reports can be successfully received. According to the usage report, failing communication and the broken SMs must be identified.

4 Proposed Scheme

The RESA scheme consists of the following procedures: initialization, hop-by-hop aggregation, failure report and secure response protocols. To simplify the description, we assume that each SM M_i only has one child M_{i-1}. It can be easily extended to other tree construction.

In the initialization protocol, the interrelated SMs generate their session keys. After that, the SM reports its energy consumption data regularly (normally every 15 min) using the hop-by-hop aggregation protocol. The SM encrypts the usage data using Elgamal encryption algorithm, compresses it to the usage report from the child and sends it to its parent. During the failure report protocol, if a SM M_i does not receive the usage report from its child M_{i-1}, it will report the communication delay and inform M_{i-1}'s child M_{i-2} and reconstruct the aggregation tree. In the secure response protocol, The EU broadcasts the instructions with the signature to the SMs once it discovers the generated amount of electricity cannot satisfy the requested quantity. Consumers check the timestamp, confirm that the instruction is valid and make their decisions.

4.1 Initialization

Based on the system model in Sect. 3, the EU and SMs execute the following steps to initialize the system.

1. Firstly, each SM is equipped with a tamper-resistant black box [12]. The black box contains a key pair **(SK, PK)**. Any other party has access to the public key **PK**. The secret key **SK** is stored securely within the black box

and is never disclosed or changed, as the black box is assumed to be tamper-resistant. A secure public key signature scheme, including a signing algorithm **sig** and a verification algorithm **ver**, has been selected for a SM with the key pair **(SK, PK)**. Additionally, a hash function $\mathcal{H} \leftarrow \mathbb{Z}_p^*$ will be selected to generate the hash-based message authentication code.

2. Then, the EU selects l random integers $a_1, ..., a_l$, computes $A = \prod_{j=1}^{l} a_j$, $A_1 = A/a_1$, ..., $A_l = A/a_l$ and $A_1^{-1} = 1/A_1 \bmod a_1$, ..., $A_l^{-1} = 1/A_l \bmod a_l$. After that, the EU selects a mapping function pair $(\mathcal{M} \leftarrow \mathbb{G}, \mathcal{M}' \leftarrow Z)$ and generates $(p, P, \mathbb{G}, \mathbb{Z}_p^*)$ as defined in Subsect. 2.3. The EU randomly selects a number $x \in \mathbb{Z}_p^*$ and computes $P_{pub} = xP$. The EU keeps x as the secret key and publishes the public parameters and functions $(p, P, P_{pub}, \mathbb{G}, \mathbb{Z}_p^*, A_1A_1^{-1}$, ..., $A_lA_l^{-1}$, A, $\mathcal{H}, \mathcal{M}, \mathcal{M}')$.
3. Finally, the EU and all SMs construct an aggregation map. The optimal approach to construct the aggregation tree is proved in Subsect. 5.2. Each SM M_i securely communicates with its neighboring SMs by running the key establishment protocol [13]. Then, two session keys $K_{i-1,i}$ and $K_{i,i+1}$are shared between M_i and M_{i-1} and M_i and M_{i+1} respectively.

4.2 Hop-by-hop Aggregation

In order to achieve the almost real-time usage report, SMs run the hop-by-hop aggregation protocol to aggregate the multidimensional usage data.

1. Firstly, the SM M_i compresses its multidimensional usage data $d_{1,i}$, ..., $d_{l,i}$ into a single dimensional one according to the Chinese remainder theorem: $D_i = d_{1,i}A_1A_1^{-1} + ... + d_{l,i}A_lA_l^{-1} \bmod A$, maps D_i into group $m_i = \mathcal{M}(D_i)$ and encrypts the element m_i. It randomly picks an integer $r_i \in \mathbb{Z}_p^*$ and computes $c_{i,1} = r_iP$, $c_{i,2} = r_iP_{pub} + m_i$.
2. Then, upon the receipt of the encrypted usage report $(\text{HMAC}_{i-1}, C_{i-1})$ from M_{i-1}, M_i checks its validity by using the session key $K_{i-1,i}$. M_i computes $\text{HMAC}_{i-1} = \mathcal{H}(K_{i-1}, C_{i-1}, t)$, and compares HMAC_{i-1} and HMAC'_{i-1}. If they are equal, M_i can confirm the usage report is from M_{i-1}.
3. After that, M_i compresses its encrypted data D_i into the usage report $C_{i,1} = c_{i,1} + C_{i-1,1}$, $C_{i,2} = c_{i,2} + C_{i-1,2}$, and reports it to its parent. It computes $\text{HMAC}_i = \mathcal{H}(C_{i,1}, C_{i,2}, t)$ and sends $(\text{HMAC}_i, C_{i,1}, C_{i,2})$ to M_{i+1}.
4. Finally, upon the receipt of a usage report $(\text{HMAC}_n, C_{n,1}, C_{n,2})$ from M_n, the EU checks its validity. If it is valid, the EU decrypts the report $(C_{n,1}, C_{n,2})$ and gets the aggregated usage data. It computes $m = C_{n,2} - xC_{n,1}$, $D = \mathcal{M}'(m)$, and extracts $d_1 = D \bmod a_1$,..., $d_l = D \bmod a_l$. The multidimensional usage data D_i is also additive homomorphic, which can be implicitly expressed as $D = \sum_{i=1}^{n} D_i = \sum_{i=1}^{n} d_{1,i}A_1A_1^{-1} + ... + d_{l,i}A_lA_l^{-1}$. Therefore, $d_1 = \sum_{i=1}^{n} d_{1,i}$, ..., $d_l = \sum_{i=1}^{n} d_{l,i}$.

4.3 Failure Report

If the communication is delayed or the SM fails, its neighbors will execute the failure report protocol to inform the EU where the failure has occurred.

1. If M_{i+1} does not receive the usage report from its child M_i, it sends the report request $(R, t,$ **sin** $($**SK**, $\mathcal{H}(R, t)))$ to M_{i-1}. Next, M_{i-1} requests the usage report of M_i.
2. If M_i can receive the request, it checks whether the request **ver**$($**PK**, $\mathcal{H}(R, t))$ is valid. If it's valid, it sends the report to M_{i-1}. If M_{i-1} cannot receive the usage report from M_i either, it reports the failure of M_i to EU.

4.4 Secure Response

Once the EU finds that the energy consumption is not equal to the production, it executes the secure response protocol to instruct consumers to adapt their usage behavior.

The EU first defines the consumption curtailment λ as the instruction. The EU then generates a valid signature to prove its identity using the selected public key signature scheme. It computes $f =$**sin**$(x, \mathcal{H}(\lambda, t))$. At last, the EU broadcasts the instructions and the signature (λ, f, t) to all SMs. Upon receiving the usage instructions, the SM checks whether the timestamp and the instruction are valid. It checks whether **ver**$(P_{pub}, \mathcal{H}(\lambda, t))$ holds. If it holds, the SM informs its consumer; otherwise, it just rejects the instruction and signature.

5 Performance Analysis

In this section, we first show that RESA satisfies the security requirements proposed in Subsect. 3.2. Then, we simulate the aggregation protocol of RESA, and compare it with existing approaches.

5.1 Security and Privacy

The security requirements can mainly be divided into trustworthiness and privacy preservation part. Trustworthiness includes the authentication and integrity of usage reports, requests and instructions in the RESA scheme. In the Initialization protocol, the public keys of SMs are predefined and auditable. Each SM shares the session key **K** with its neighbors using Diffie-Hellman key exchange protocol [13]. The key exchange protocol is secure under Diffie-Hellman assumption. Therefore, the adversary cannot get the session key between the SM and its neighbors. When the SM sends the power usage report to its parent, it proves the usage report with the hash-based message authentication code. According to the security requirements of key exchange protocol and hash-based message authentication code [14], the adversary cannot forge a valid usage report or modify the legitimate usage data without authorization. If a SM finds it needs to execute the failure report protocol, it signs the request using the selected signature scheme **(sig, ver)**. Therefore, the adversary cannot disturb the aggregation protocol without a SM's secret key **SK**, which is assumed to be protected by a tamper resistant black box [12]. During the demand response part, the EU's consumption instruction is signed by the selected public key signature (**sin, ver**). As the result, RESA can ensure integrity and authentication.

The confidentiality of RESA is based on the Engamal encryption algorithm. In RESA, M_i's usage data $d_{i1}, ..., d_{il}$ is compressed into D_i, then the ciphertext $(c_{i,1}, c_{i,2})$ is still a valid ciphertext of M_i's encryption. Elgamal encryption algorithm is semantic secure against chosen plaintext attacks. Therefore, even if an adversary eavesdrops the communication between two SMs and gets the ciphertext (C_1, C_2), he cannot get the aggregated usage data D_i. According to the analysis of the confidentiality, other parties, except the EU, cannot get the plaintext of the usage report. If an adversary intrudes into the database of the EU, he can only get the aggregated data $d_1, d_2, ..., d_l$. Therefore, the adversary cannot get the individual usage data, if the number of consumers is large enough.

5.2 Communication Analysis

During the hop-by-hop aggregation protocol, each SM compresses the multi-dimensional data, encrypts it and adds it to the aggregated usage data. Although the multi-dimension data leads to a larger message, which is inefficient in public key encryption, it is more efficient than the sum of encryption operations for all dimensions [4]. To our best knowledge, only [4,7,8] focus on the secure multi-dimensional data aggregation. [7] does not adopt additive aggregation approaches. [4,8] adopt the Paillier encryption system. The SM needs an exponentiation operation for each dimension. Moreover, the size of the base affects the security of the ciphertext. In RESA, the SM performs a multiplication operation. The whole compressed dimensional data is encrypted. The size of the parameter is not related to the security. Therefore, the computational cost is much smaller than that of [4,8] for each data dimension.

Compared with Elgamal encryption, the overhead of hash evaluations and additive operations is very small [17]. However, all SMs can perform the encryption in parallel. For one thing, the expensive encryption calculations are distributed to all SMs. For another, the highly efficient HMAC can be employed instead of public key signatures. Therefore, the computation time is $T_h + kT_a$, where k stands for the number of children, T_h stands for he homomorphic computation time, and T_a stands for authentication and addition operation time. Therefore, the more children the SM has, the more computational time it costs. At the same time, we assume that the communication time between any two neighbor SMs T_c is constant in the Smart Grid system. The height of the aggregation tree is no less than $\log_k n$, as the number of SMs n is fixed in the Smart Grid system. The lower the aggregation tree is, the less computational overhead is required. The hop-by-hop aggregation time is the sum of the computational cost and the communication overhead. Therefore, it is necessary to determine the children number k for a SM to optimize the aggregation time. We assume that the number of SMs n is fixed. The data encryption computation of all SMs can be calculated in parallel.

According to the performance analysis above, the aggregation time is $T(k) = T_h + (kT_a + T_c)\log_k n$. The differentiation of $T(k)$ is $T'(k) = -\frac{\ln n}{k(\ln k)^2 k}(kT_a + T_c) + \frac{T_a \ln n}{\ln k}$. Therefore, the aggregation time is minimal if k satisfies $k \ln k - k = \frac{T_c}{T_a}$.

Here, we set a scenario to compare the aggregation time, which includes the encryption, transmission and authentication time, of RESA and other third party free schemes [8 – 10]. We emulate the schemes on an Ubuntu 12.04 virtual machine with an Intel Core i5-4300 dual-core 2.60 GHz CPU. We only use one core 798 MHz CPU and 256 MB of RAM to simulate a SM, which is not far away from a real SM. We assume that the computational ability of EU is 100 times more than a SM. To achieve 80 bits security level, we set the length of $\mathbb{G}$ to 161 bits and p to 160 bits. The form of usage report is $\text{HMAC}_i \| C_{i,1} \| C_{i,2}$ whose size is 578 bits. The size of $\mathbf{K}$ is 128 bits. All results are obtained running 20 test repetitions for each algorithm. According to the simulation, the mean result of T_h and T_a are 21,89 ms and 0.01778 ms, which have 0.45 ms and 0.00017 ms size of 95 % confidence intervals. [19] summarizes the data rate of WIMAX in Smart Grids. Here the communication rate is set to 2 Mbps. The variation of aggregation time in terms of the number of SMs among RESA and some previous third party free schemes are compared. According to the Newton's method, the maximum number of children is 3 for each SM when communication rate is 2 Mbps. The computational costs and simulation results are depicted in Fig. 2.

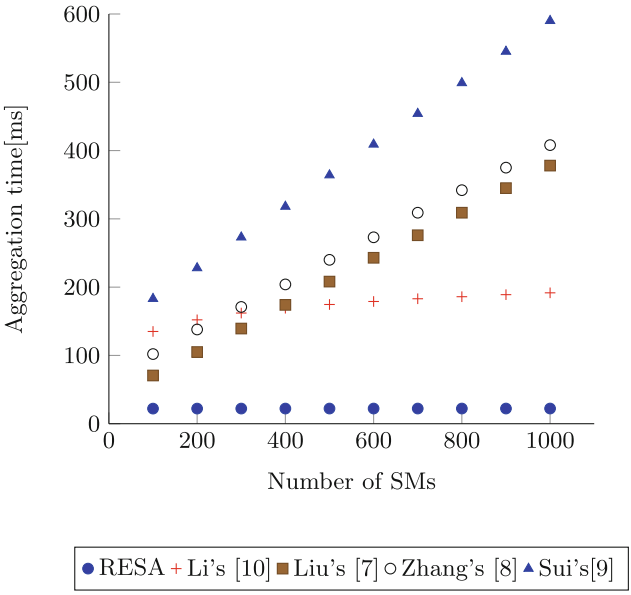

Fig. 2. Computational cost

It can be seen from Fig. 2 that the aggregation time is significantly reduced in RESA compared to other third party free aggregation schemes. According to the requirement of the IETF RFC 4594 standard [3], the signaling delay value should be no more than 240 ms. However, the aggregation time exceeds the threshold

when the number of SMs reaches 1000 in [7 – 9]. One of the most important reasons is the limitation of EU's computation ability. There are usually hundreds of even thousands of SMs in a Smart Grid system. To ensure the trustworthiness, the EU should check all SMs' signatures in previous schemes. Although batch verification is popular in some systems, it cannot reduce the computation complexity to be lower than $\mathcal{O}(n)$ currently. This also can be solved by improving the computation ability at the EU's side. However, the improvement requires costly devices. In RESA and [10], the costly computation is distributed to all SMs. Each SM is responsible for checking the validity of its children's usage report. The aggregation time is $\mathcal{O}(\ln n)$. Therefore, the communication delay satisfies the requirement of IETF RFC 4594. Although [10] also employs aggregation tree models, RESA utilizes high efficient hash-based message authentication codes to reduce the computation costs, instead of public key signatures.

6 Conclusion

In this paper, a novel aggregation tree model named RESA is developed, a robust and efficient secure aggregation scheme in Smart Grids. The usage reports are aggregated according to the aggregation tree. RESA can identify and report communication failure during the aggregation time. Therefore, the robustness of the Smart Grid system is improved. Although the multi-hop communication would lead to more communication delay, we optimize the aggregation tree and build a novel model to determine the number of children in the aggregation tree. The security analysis shows that RESA satisfies the security and privacy requirements. Communication analysis proves that RESA has better communication performance than some previous works under the assumption that the computation ability is limited. Therefore, we conclude that RESA is more suitable for real time requirement than other similar approaches.

Acknowledgments. The research leading to these results was supported by the European Commission's Project No. 608090, HyRiM (Hybrid Risk Management for Utility Networks) under the 7th Framework Programme (FP7-SEC-2013-1). The first author of this work is supported by the Chinese Scholarship Council.

References

1. Iyer, S.: Cyber security for smart grid, cryptography, privacy. Int. J. Digit. Multimedia Broadcast. **2011**, 1–8 (2011)
2. Fan, Z., Kalogridis, G., Efthymiou, C., Sooriyabandara, M., Serizawa, M., McGeehan, J.: The new frontier of communications research: smart grid and smart metering. In: First International Conference on Energy-Efficient Computing and Networking, Passau, Germany, 13–15 April, pp. 115–118. ACM (2010)
3. Babizrz, J., Chan, K., Baker, F.: Configuration Guidelines for DiffServ Service Classes. IETF RFC 4594, August 2006. http://www.ietf.org/rfc/rfc4594.txt

4. Rongxing, L., Liang, X., Li, X.L., Shen, X.: Eppa: an efficient and privacy-preserving aggregation scheme for secure smart grid communications. IEEE Trans. Parallel Distrib. Syst. **23**(9), 1621–1631 (2012). IEEE
5. Paillier, P.: Public-key cryptosystems based on composite degree residuosity classes. In: Stern, J. (ed.) EUROCRYPT 1999. LNCS, vol. 1592, pp. 223–228. Springer, Heidelberg (1999)
6. Chia-Mu, Y., Chen, C.-Y., Kuo, S.-Y., Chao, H.-C.: Privacy-preserving power request in smart grid networks. IEEE Syst. J. **8**(2), 441–449 (2014). IEEE
7. Liu, X., Zhang, Y., Wang, B., Wang, H.: An anonymous data aggregation scheme for smart grid systems. Secur. Commun. Netw. **7**(3), 602–610 (2014). John Wiley & Sons
8. Zhang, J., Liu, L., Cui, Y., Chen, Z.: SP 2 DAS: self-certified PKC-based privacy-preserving data aggregation scheme in smart grid. Int. J. Distrib. Sens. Netw. **2013**, 1–11 (2013). Hindawi
9. Sui, Z., Alyousef, A., de Meer, H.: IAA: incentive-based anonymous authentication scheme in smart grids. In: Tiropanis, T., Vakali, A., Sartori, L., Burnap, P. (eds.) Internet Science. LNCS, vol. 9089, pp. 133–144. Springer, Heidelberg (2015)
10. Li, F., Luo, B.: Preserving data integrity for smart grid data aggregation. In: Third International Conference on Smart Grid Communications (SmartGridComm), Tainan, China, 5–8 November, pp. 366–371. IEEE (2012)
11. Li, F., Luo, B., Liu, P.: Secure information aggregation for smart grids using homomorphic encryption, in First IEEE International Conference on Smart Grid Communications (SmartGridComm), Gaithersburg, USA, 4–6 October, pp. 327–332. IEEE (2010)
12. Stamm, S., Sheppard, N.P., Safavi-Naini, R.: Implementing trusted terminals with a, SITDRM. Electr. Not. Theoret. Comput. Sci. **197**(1), 73–85 (2008). Elsevier
13. Diffie, W., Hellman, M.E.: New directions in cryptography. IEEE Trans. Inf. Theory **22**(6), 644–654 (1976). IEEE
14. Krawczyk, H., Canetti, R., Bellare, M.: HMAC: keyed-hashing for message authentication. Network Working Group (1997)
15. Mykletun, E., Girao, J., Westhoff, D.: Public key based cryptoschemes for data concealment in wireless sensor networks. In: IEEE International Conference on Communications, Istanbul, 11–15 June, vol. 5, pp. 2288–2295. IEEE (2006)
16. Schnorr, C.-P.: Efficient signature generation by smart cards. J. Cryptol. **4**(3), 161–174 (1991)
17. Lynn, B.: PBC library. http://crypto.stanford.edu/pbc/. last accessed January 2015
18. Multiprecision integer and rational arithmetic c/c++ library. http://www.shamus.ie/. last accessed January 2015
19. Gungor, V.C., Sahin, D., Kocak, T., Ergut, S., Buccella, C., Cecati, C., Hancke, G.P.: Smart grid technologies: communication technologies and standards. IEEE Trans. Industr. Inf. **7**(4), 529–539 (2011)