Secret Share Based Program Access Authorization Protocol for Smart Metering

Xiuxia Tian[1], Lisha Li[2], Jinguo Li[1], Hongjiao Li[1] and Chunhua Gu[1]
(Corresponding author: Xiuxia Tian)
School of Computer Science and Technology, Shanghai University of Electric Power[1]
No. 2588 Changyang Road, Shanghai 200090, China
School of Electronics and Information Engineering, Shanghai University of Electric Power[2]
No. 2588 Changyang Road, Shanghai 200090, China
(Email: xxtian@fudan.edu.cn)
(Received Aug. 16, 2015; revised and accepted Nov. 27, 2015 & Jan. 3, 2016)

Abstract

Varieties of data security protection technologies have been developed in smart metering. However, there are almost no researches focusing on the security of smart meter parameters. In fact, smart meter parameters, such as the total/sharp/peak/flat/valley period time, can be written locally or remotely by power companies or users through sending programming commands. Modifying smart meter parameters arbitrarily may lead to the non-authenticity of users' energy bills. This paper first considers the security of smart meter parameters and proposes a novel secret share based program access authorization protocol. The protocol introduces a program lock key which is used to lock the smart meter parameter modifying program. At least t participants (i.e. one user and $t-1$ power companies) jointly can have the authorization to obtain the program lock key. We describe the security and efficiency analysis through theory and experiments.

Keywords: Program access, smart meter parameters, security protection, secret share

1 Introduction

Smart meters have been widely deployed all over the world. Varieties of security protection technologies have been developed in smart metering. Most of them focus on the smart meter data security protection. However, almost no approaches focus on the security of smart meter parameters. Smart meter parameters, such as the total/sharp/peak/flat/valley period time, should be protected from being modified arbitrarily because they affect the authenticity of users' energy bills.

As shown in Figure 1, we take the sharp/peak period time for example. Assume the peak period time is $3pm-7pm$ and the sharp period time is $7pm-10pm$; the electricity price is 1.21567 $yuan/kwh$ in the sharp period time and 1.14460 $yuan/kwh$ in the peak period time. If a valid user's monthly electricity consumption is 600 kwh in the sharp period time and 100 kwh in the peak period time, the real bill is 843.86200 $yuan$, as shown in Figure 1. Assume the valid user's monthly electricity consumption/hour at one period time is equal. Thus the valid user's monthly electricity consumption/hour is $600 \div 3 = 200$ kwh at the sharp period time and $100 \div 4 = 25$ kwh at the peak period time. A curious power company may modify the peak period time from $3pm-7pm$ to $3pm-6pm$ and in turn the sharp period time from $7pm-10pm$ to $6pm-10pm$ through sending programming commands. The energy bill is changed to 845.63875 $yuan$, as shown in Figure 1. Thus the valid user needs to pay $845.63875-843.86200=1.77675$ $yuan$ more than what he need to pay while the curious power company will acquire additional profit 1.77675 $yuan$. A malicious user may modify the peak period time from $3pm-7pm$ to $3pm-8pm$ and in turn the sharp period time from $7pm-10pm$ to $8pm-10pm$, which is a behavior of stealing electricity. The energy bill becomes 829.64800 $yuan$, as shown in Figure 1. So the malicious user may pay $843.86200-829.64800=14.21400$ $yuan$ less than the real bill while the valid power company will loss 14.21400 $yuan$.

Therefore, we need to limit the modifying permissions on smart meter parameters. By introducing a program lock key which is used to lock the smart meter parameter modifying program, we present a novel secret share based program access authorization protocol to solve the problem. Any one user or power company can not arbitrarily modify smart meter parameters for their own purposes.

The contributions of this paper are in the following:

- First proposing a novel program access authorization protocol to guarantee the security of smart meter parameters.
- Introducing the secret share scheme to authorize the

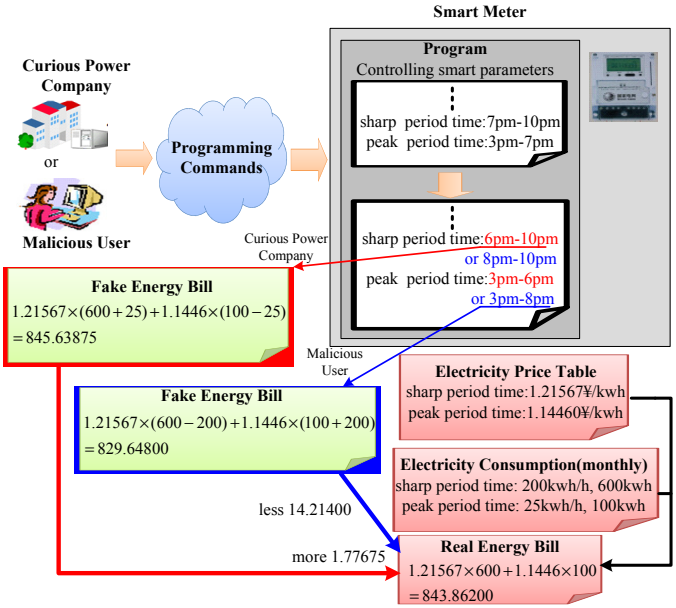

Figure 1: An example of unauthorized modifying smart meter parameters

modifying permissions of smart meter parameters.

- Quantitatively analyzing the security by introducing the binomial distribution metric.
- Experimental results show that the protocol is efficient.

The rest of this paper is organized as follows. Section 2 introduces the preliminary knowledge used in the protocol. Then we investigate the related work in Section 3. We describe the system model and security requirements in Section 4 and then present our proposed protocol in Section 5. Security analysis and experiment evaluation are shown in Sections 6 and 7 respectively. Finally, we conclude the paper.

2 Preliminary

2.1 Shamir's (t, n) Threshold Secret Sharing Scheme

(t, n) threshold key sharing scheme based on Lagrange interpolation formula was proposed in 1979 by A. Shamir [18], where a secret key can be divided into n shares, and each share is distributed to one participant, only the designated number of participants like t or more together can reconstruct the secret key [19].

In order to understand our protocol clearly, we give the definition of *share* used throughout the paper as follows:

Definition 1. *A share is the result value y by computing the following polynomial on inputting a known x.*

$$f(x) = (K + a_1 x + a_2 x^2 + ... + a_{t-1} x^{t-1}) \bmod Q$$

where $a_1, a_2, ..., K \in \mathcal{F}_Q$, Q is a large prime, $\mathcal{F}_Q$ is a finite domain on Q, K is the secret value.

From the definition above we know that n *share*s are $y_1, y_2, ..., y_n$ computed from known $x_1, x_2, ..., x_n$ respectively, and the polynomial $y = f(x)$ can be reconstructed from any t known pairs $(x_{i1}, y_{i1}), (x_{i2}, y_{i2}), ..., (x_{it}, y_{it})$ of n pairs $(x_1, y_1), ..., (x_n, y_n)$.

2.2 Binomial Distribution

Binomial distribution is a probability distribution with discrete random variables, symbolized by $X \sim B(n, p)$, where X is the result of the randomized trial, n is the number of independent repeated trials, p is the occurrence probability of an event in one trial. The following is the mathematical definition of binomial distribution.

Definition 2. *Assuming an event A. The occurrence probability of A is p $(0 < p < 1)$ in one trial, thus the nonoccurrence probability of A in one trial is $q = 1 - p$. The probability that A occurs k times in n independent repeated trials is:*

$$P = (X = k) = C_n^k p^k q^{n-k}, k = 0, 1, ..., n \quad (1)$$

The probability that A occurs no more than k times in n independent repeated trials is:

$$F = (X = k) = P(X \leq k) = \sum_{j=0}^{k} C_n^j p^j q^{n-j} \quad (2)$$

3 Related Works

In the studies of smart metering security protection, most of them focus on smart meter data security protection. We briefly review these concerned works from two aspects: Smart meter data privacy protection and smart meter data security defense.

3.1 Data Privacy Protection

Smart meter data privacy protection aimed at achieving the power company's billing purposes and preserving users' privacy in the meantime.

The scheme in paper [5] preserved users' daily electricity usage pattern from a power operator using anonymous credentials while vast credentials need to be generated beforehand. The scheme in paper [6] using pseudo random identity requested power, which hid the smart meter's true identity and thus preserved users' privacy. The scheme in paper [8] assumed that each smart meter had two separate IDs, one of which attached to private information (i.e. HFID) was anonymous to preserve the users' privacy. In paper [24], the key distribution center without the smart meter's true identity performed billing computation and sent the total power consumption to the power company, which protected the real-time

power consumption from power companies and thus preserved users' privacy. The scheme in paper [11] used data aggregation which was performed at all smart meters to construct a aggregation tree, where smart meters as aggregation nodes routed metering data from the source meter to the power company and thus the power company only saw the final results. The schemes in papers [21] and [10] used rechargeable batteries to protect users power load information. Rechargeable batteries and the power company could individually or collectively supply electricity to smart appliances with the reasonable battery policy. Thus smart meter data could not directly reflect the smart appliances electricity usage information, which fuzzed users' electricity usage pattern. The scheme in paper [17] analyzed the trade-off relationship between information leakage and practicability using a stationary Gaussian Markov model of the electricity load. The scheme in paper [9] split the amount of electricity which was requested by users into random shares, one share for each user. Users submitted the mixed shares to the power company. Thus the power company only saw the final power consumption but could not retrieve the meaningful information about the real-time power consumption.

3.2 Data Security Defense

Smart meter data security defense aimed at protecting the smart meter data from physical attacks.

According to paper [15], smart meters could be equipped with seals, uncapping recordings, hardware programming switches or password authentication switches to protect smart meter data from being physically compromised. Both paper [25] and paper [13] suggested that the programming switch was used in conjunction with password authentication [22] to enhance defense capabilities. Paper [1] pointed out that using magnetic sensors or tilt sensors could check whether the authorized location of smart meters had been removed or physical tampered with.

Above all, there are almost no researches focusing on the security of smart meter parameters. The methods in smart meter data security defense also can be used to protect smart meter parameters from physical attacks. But smart meter parameters need to be protected from the aspect of software to against cyberattacks. This paper first proposes a secret share based program access authorization protocol to solve the problem. The protocol implements that users and power companies jointly control the modifying permissions on smart meter parameters.

4 System Model and Security Requirements

4.1 System Model

Figure 2 depicts the system architecture of the protocol where three types of participants exist: Smart meter, power companies and user.

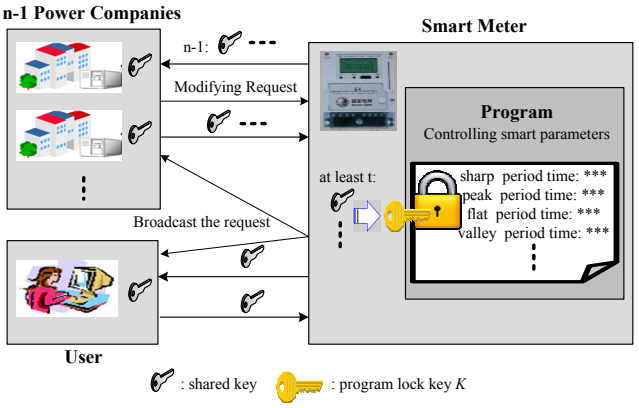

Figure 2: System architecture

- **Smart meter.** A smart meter locks the smart meter parameter modifying program by using the program lock key K and then divides the program lock key K into n shared keys. Each shared key is distributed to one participant. The smart meter asks for shared keys to recover the program lock key K when receiving a modifying request from one power company.

- **Power companies.** Power companies are semi-honest. They may modify smart meter parameters arbitrarily for their own purpose (e.g. increase users' energy bills though modifying smart meter parameters to earn the difference). There are $n-1$ power companies in the authorization protocol. Any one from these power companies can submit a modifying request to the smart meter. Other $n-2$ power companies submit shared keys to the smart meter according to the rationality of the reasons of modifying smart meter parameters.

- **User.** Users can modify smart meter parameters through modern devices, such as a new intelligent web-based electric meter concentrator according to paper [14], but they may arbitrarily modify for their own interests (e.g. decrease their own energy bills through modifying smart meter parameters to pay less than what they need to pay). In our protocol, the user is also a participant of the recovery of the program lock key K. The user submits his/her shared key to the smart meter according to the rationality of the reasons of modifying smart meter parameters.

It should be pointed out: A modifying request mainly includes programming commands and the reasons of modifying smart meter parameters; the total/sharp/peak/flat/valley period time varies with regions and seasons; the rationality of modifying reasons is determined by whether they confirm to the seasonal change or national policy.

4.2 Security Requirements

We aim at designing a program access authorization protocol to protect smart meter parameters from being modified arbitrarily. The security requirements are summarized as follows:

4.2.1 Request Message Authentication

Every modifying request message from one power company should be authenticated [3, 4] to confirm that it is from a valid entity. Thus an attacker cannot impersonate any valid power company to send out fake modifying request messages.

4.2.2 Shared Key Message Confidentiality

Shared keys distributed or recalled by the smart meter should be kept confidential to protect the security of the program lock key K [16]. Thus an attacker should not be able to get the shared keys to recover the program lock key K.

4.2.3 Anti-attacking Ability

The protocol should have anti-attacking abilities to protect the participants against outside attackers and thus protect the security of shared keys. An attacker should not be able to obtain the shared keys through attacking the participants.

5 The Protocol

5.1 The Notations in the Protocol

The notations used in the protocol are shown in Table 1.

Table 1: The notations in the protocol

Notation	Description
M	Smart meter
A/A_i	Power company/ the i^{th} power company
U	User who uses the smart meter
Program	A program which controls modifying smart meter parameters
K	A program lock key which is used to lock the program
Pub_R/Pri_R	Public key/private key pair of an entity R
$E_k(program)$	Encrypting the program under the program lock key K
$E_{Pub_R}(J)$	Encrypting the information J under the public key of an entity R
Sig_{Pri_R}	Signature of an entity R

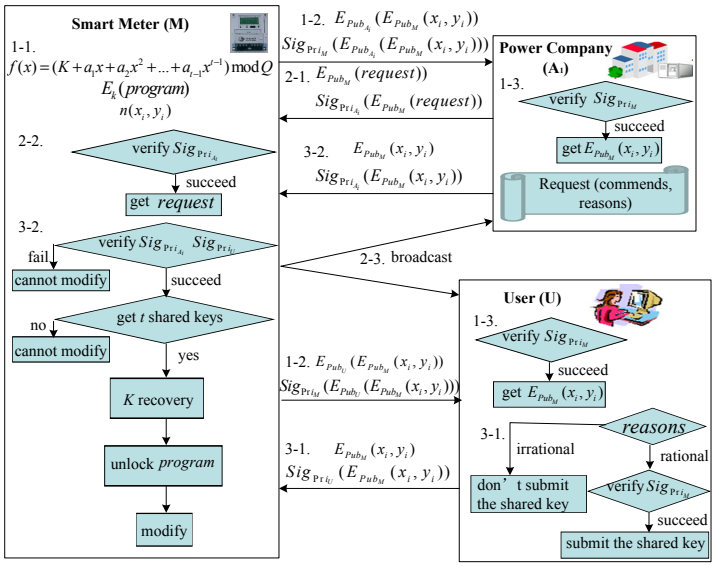

Figure 3: The protocol

5.2 The Protocol

Figure 3 shows the secret share based authorization protocol for smart metering. Actually there are $n-1$ power companies in our protocol. We only show one power company in Figure 3 for simplifying instructions. The following is the detailed processes of the protocol.

5.2.1 Generating And Distributing Shared Keys

Each protocol participant has its own private/public key and others' public keys (this process based on public key cryptography is not our focus, so we do not discuss it in detail). Below is the description of generating and distributing shared keys.

Step 1-1/2. A smart meter M randomly generates a $t-1$ power of polynomial, $f(x) = (K + a_1x + a_2x^2 + ... + a_{t-1}x^{t-1}) \bmod Q$ (Q is a prime number, $K < Q$). The smart meter uses the program lock key K to lock the smart meter parameter modifying program, $E_k(program)$. Then the smart meter selects n different non-zero elements (x_i, y_i) $(x_i \in Q, 1 \leq i \leq n, y_i = f(x_i))$ called secret keys. Anyone who has at least t $(t \leq n)$ secret keys can recover the program lock key K [23]. The smart meter encrypts these n secret keys with its public key, $E_Pub_M(x_i, y_i)$ called shared keys. Then the smart meter encrypts these n shared keys with the corresponding participants' public keys and sends them to the participants separately with its signature. At last, the smart meter destroys any information about the program lock key K, such as the shared keys $E_Pub_M(x_i, y_i)$, the polynomial $f(x)$, and the secret keys (x_i, y_i).

Step 1-3. Upon the power companies and user receive the massages, they first verify the signature of the smart meter. After succeeding, they get their own

shared keys $EPub_M(x_i, y_i)$ using their own private keys. But they can never know (x_i, y_i), the confidential information of the program lock key K, without the private key of the smart meter.

5.2.2 Modifying Request

If one power company (e.g. A_1) wants to modify smart meter parameters, he has to send a modifying request and his own shared key [7] to the smart meter. Below is the description of this phase (Figure 3).

Step 2-1. Power company A_1 sends a modifying request to the smart meter. The request is encrypted with the smart meter's public key and sent with the power company's signature.

Step 2-2. Upon the smart meter receives the request, it first verifies the power company's signature and then gets the request using its private key.

Step 2-3. Then the smart meter broadcasts the modifying request and it's signature to all participants asking for shared keys.

5.2.3 Recovering The Program Lock Key K

Step 3-1. Upon receiving the broadcast, other power companies and user first judge the rationality of the modifying reasons. If anyone considers the reasons are irrationality, he/she does not submit his/her own shared key to the smart meter. Otherwise, he/she verifies the smart meter's signature and sends his/her own shared key with his/her signature to the smart meter.

Step 3-2. Upon receiving the massages from the power companies and user, the smart meter first verifies their signatures respectively. After succeeding, the smart meter gets the shared keys. If getting less than t shared keys, the smart meter can not recover the program lock key K. This means the modifying request fails. Otherwise, the smart meter recovers the program lock key K and unlocks the program to modify the smart meter parameters according to the modifying request.

6 Security Analysis

In this section, we evaluate the protocol generally according to the security requirements summarized in Section 4.

6.1 Request Message Authentication

Before a power company sends a modifying request message to the smart meter, the power company has to sign the message using his private key. The private key is only known by the power company. Hence an attacker does not know how to produce the signature of the power company without the power company's private key. Thus an attacker could not able to pretend the power company to transmit the request message.

6.2 Shared Key Message Confidentiality

Shared keys (i.e. $EPub_M(x_i, y_i)$) are encrypted using the public key of the smart meter, which is significant from the conventional secret share scheme, no one can get the secret keys (i.e. (x_i, y_i)) except the smart meter. Thus, an attacker can not acquire the secret keys, the confidential information of the program lock key K, through eavesdropping or intercepting the shared key messages.

6.3 Anti-attacking Abilities Analysis

The program lock key K is used to lock the smart meter parameter modifying program. If getting the program lock key K, attackers can open the program to do something with smart meter parameters and then achieve their malicious purposes. The security of the program lock key K is crucial to the protocol. According to the targets which attackers attack to, we analyze the anti-attacking abilities from the following two aspects.

6.3.1 Targets Are Smart Meters

A smart meter randomly generates a polynomial and then gets the program lock key K and shared keys from this polynomial. After using the program lock key K to lock the smart meter parameter modifying program and distributing the shared keys to all participants, the smart meter destroys the program lock key K, the polynomial (i.e. $f(x) = (K + a_1x + a_2x^2 + ... + a_{t-1}x^{t-1}) \bmod Q$), the secret keys (i.e. (x_i, y_i)) and the shared keys (i.e. $EPub_M(x_i, y_i)$). This means that the smart meter does not store any information about the program lock key K. Thus, attackers can not get any information about the program lock key K through attacking [2] smart meters.

6.3.2 Targets Are Power Companies/Users

A power company/user receives a shared key from the smart meter and keeps it for future use (i.e. sends the shared key back to the smart meter for modifying smart meter parameters). The shared key (i.e. $EPub_M(x_i, y_i)$) is encrypted using the public key of the smart meter, no one except the smart meter can decrypt it. Thus, even though obtaining the shared key through attacking the power company/user, attackers can not get the confidential information (i.e. (x_i, y_i)) of the program lock key K.

6.4 Less Than t Participants Colluding Together Can Never Modify Smart Meter Parameters

From Section 5, we know that if a smart meter gets less than t shared keys, it can not recover the program lock key K to open up the smart meter parameter modifying

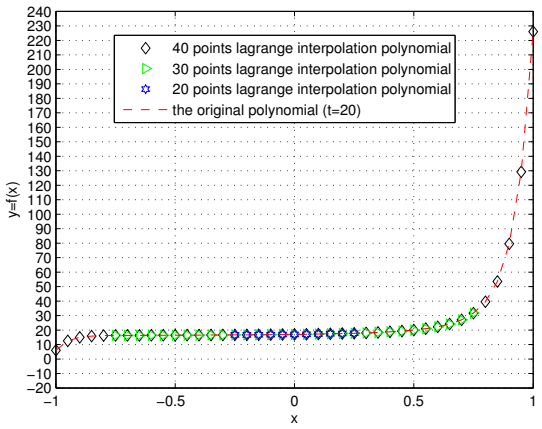

Figure 4: Success of recovering the program lock key K

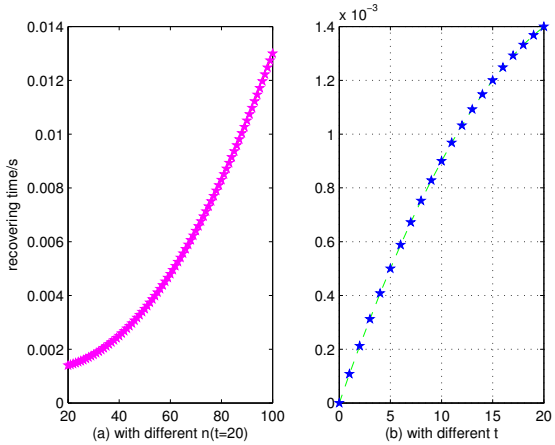

Figure 5: The time cost for the program lock key K recovering

program. So less than t malicious participants colluding together can not unlock the program to modify smart meter parameters. Any one power company/user cannot modify smart meter parameters arbitrarily through sending his/her only one shared key. Moreover, in real life, most power companies are trusted entities. So the probability of the smart meter parameters arbitrarily being modified is very small.

In addition, the program lock key K can not be derived out by participants. Since the shared keys are encrypted with the public key of the smart meter, the power companies and user can not get the confidential information of their own stored keys. So the program lock key K can not be deduced out by the participants colluding together through some algorithms (e.g. exhaustive algorithm), which is superior to the conventional secret share scheme.

7 Experiment Evaluation

7.1 The Determination of the Optimal t

We balance the time cost of the program lock key K recovering and the security of the program lock key K to determinate the optimal threshold t.

Figure 4 shows an 19th (i.e. $t = 20$) polynomial $y = f(x)$ is reconstructed by 20, 30 and 40 points from this polynomial respectively. From Figure 4, we can see that the curves of 20, 30 and 40 points lagrange interpolation polynomials overlap at zero, which proves that 20, 30 and 40 points can recover the program lock key K successfully. So we use the lagrange interpolation polynomial to test the time cost for the program lock key K recovering in MATLAB R2013a. The test results are shown in Figure 5.

From Figure 5(a), we can see that: When t ($t = 20$) is certain, the time cost increases as n increases, where n is the number of all participants in the protocol (i.e. one user and $n-1$ power companies). For example: When $n = 40$, the time cost for recovering is about 2.5 milliseconds; when $n = 100$, the time cost for recovering is about 13 milliseconds. So we assume that the smart meter choose t but not n shared keys to recover the program lock key K to reduce the recovering time cost.

From Figure 5(b), we can see that: When t is not certain, the time cost increases as t increases. Just take the time cost of the program lock key K recovering into consideration, the threshold t is as small as possible. But if consider the security of the program lock key K in the meantime, the threshold t is really not as small as possible.

Paper [12], which uses a precise number complementary judgment matrix sorting algorithm, established a decision analysis method for the selection of the $t \setminus n$ value in the threshold key sharing scheme. But this decision analysis method does not quantitatively analyze the security of Shamir's (t, n) threshold secret sharing scheme. We propose a method that uses binomial distribution, a simple mathematical algorithm, to quantitatively analyze the security of the program lock key K. The experiments we do are as follows.

Binomial distribution is symbolized by $B(n, p)$ mentioned in Section 2. Here we convert the symbol $B(n, p)$ into the analysis of the security of the program lock key K. We assume that n is the number of all participants in the protocol as before, and p is the probability of one secret key (i.e. (x_i, y_i)) leakage. Less than t secret keys leakage can not threat the security of the program lock key K, so the formula of the security degree (sd) of the program lock key K is as follows and the explication is shown in the following box. The experimental results are shown in Figure 6.

$$sd = P(X \leq t) - P(X = t)$$
$$= \sum_{j=0}^{t} C_n^j p^j (1-p)^{n-j} - C_n^t p^t (1-p)^{n-t} (t = 1, ..., n) \quad (3)$$

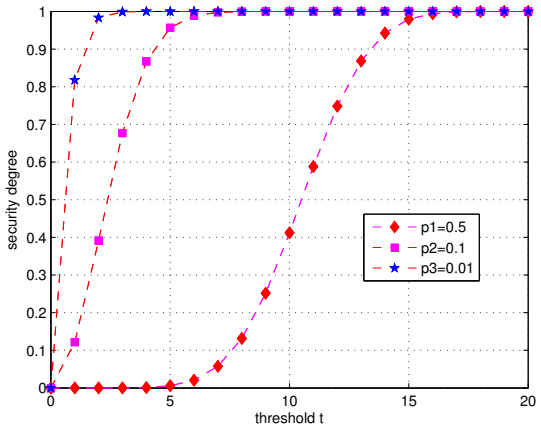

Figure 6: The security of the program lock key K

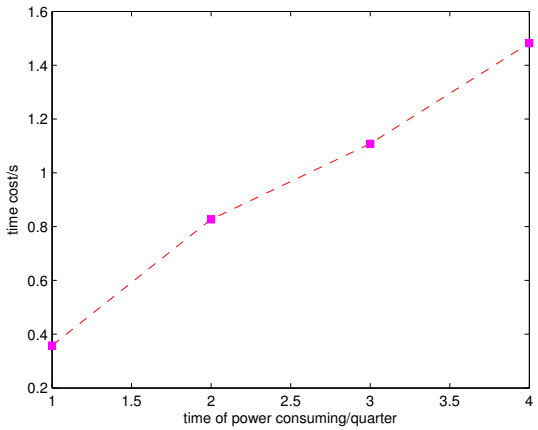

Figure 7: The time cost for encryption and decryption

The probability (degree) of security
= The probability of $(t-1)$ secret keys leakage
+ the probability of $(t-2)$ secret keys leakage
+ ... + the probability of one secret key leakage
= The probability of no more than t secret keys leakage - The probability of t secret keys leakage

Figure 6 shows the security of the program lock key K when $n = 20$. From Figure 6, we can see that: When n and p are certain, as t increases, the security degree is from zero to the maximum and then remains unchanged; when n is certain and p is very small, choosing small t can reach a high security degree. As shown in Figure 6, when $p = 0.5$, the security degree reaches the maximum (about 100%) when $t = 17$. So $t = 17$ is the optimal threshold when $n = 20$ and $p = 0.5$. From Figure 5(b), we can also get the time cost for recovering is about 1.3 milliseconds when $t = 17$. In addition, the secret key (i.e. (x_i, y_i)) is encrypted under the public key of the smart meter and no one except the smart meter can decrypt it, thus the probability of one secret key (i.e. p) leakage is quite small. So the protocol can have very high security degree with a small threshold (i.e. t).

7.2 The Time Cost for Encryption and Decryption

We deploy the environment with a ThinkPad Core 2 CPU E425 @1.90GHz PC, and choose RSA (1024 bits) as an the asymmetric encryption algorithm, coding in C.

The time cost for encryption and decryption in the fist phase of the protocol, namely the generating and distributing shared keys phase, is only about 94 milliseconds. Moreover, the generating and distributing shared keys phase is only carried out once in a smart meter unless the program lock key K leaks. Therefore, The time cost for encryption and decryption in the generating and distributing shared keys phase has little influence on the protocol efficiency.

In general, the second phase and the third phase of the protocol, namely the modifying request phase and the recovering the program lock key K phase, carry out once every three months (i.e. a quarter of a year) in China. We take encrypting and decrypting the longest information for example to test the time cost for encryption and decryption in these two phases during one year. The test results are shown in Figure 7. From Figure 7, we can see that the cost time for encryption and decryption is roughly linear with the quarter. The ratio of the cost time and the effective time is the average gradient (ag) of the curve in Figure 7, $ag = (ag_1 + ag_2)/2$, $ag_1 = (y_3 - y_1)/(x_3 - x_1)$, $ag_2 = (y_4 - y_2)/(x_4 - x_2)$, where x_i and y_i are the power consuming time (effective time) and the time cost for encryption and decryption respectively. This ratio is almost a constant value and so small, only about $5.26 \times 10^{-6}\%$. So the protocol has good efficiency.

8 Conclusions

Smart meter parameters, such as the total/sharp/peak/flat/valley period time, affect the authenticity of energy bills of users. Users or power companies can modify smart meter parameters through sending programming commands. However, they should not arbitrarily modify smart meter parameters for their own purposes. This paper proposes a secret share based program access authorization protocol for smart metering. The protocol realizes that one user and power companies jointly control the modifying permissions on smart meter parameters. Our future work is to construct the hierarchy based administrator domain [20] to enforce the program access authorization mechanism.

Acknowledgments

This work was supported by NSFC Grants (No. 61202020), Project of Shanghai Science and Technology Committee Grant (No. 15110500700) and CCF-Tencent Open Fund Grant (No. IAGR20150109, RAGR20150114).

References

[1] M. Balakrishnan, "Security strategy of smart meters," *China Electronic Market*, pp. 56–57, 2012.

[2] M. Bayat and M. R. Aref, "An attribute based key agreement protocol resilient to kci attack," *International Journal of Electronics and Information Engineering*, vol. 2, no. 1, pp. 10–20, 2015.

[3] C. C. Chang, J. H. Yang and Y. C. Wu, "An efficient and practical authenticated communication scheme for vehicular ad hoc networks," *International Journal of Network Security*, vol. 17, no. 6, pp. 702–707, 2015.

[4] Q. F. Cheng and C. M. Tang, "Cryptanalysis of an id-based authenticated dynamic group key agreement with optimal round," *International Journal of Network Security*, vol. 17, no. 6, pp. 678–682, 2015.

[5] J. C. L. Cheung, T. W. Chim, S. M. Yiu, C. K. Hui and V. O. K. Li, "Credential-based privacy-preserving power request scheme for smart grid network," in *Proceedings of the IEEE Global Telecommunications Conference*, pp. 1–5, Houston, 2011.

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

[7] X. D. Dong, "A multi-secret sharing scheme based on the CRT and RSA," *International Journal of Electronics and Information Engineering*, vol. 2, no. 1, pp. 47–51, 2015.

[8] C. Efthymiou and G. Kalogridis, "Smart grid privacy via anonymization of smart metering data," in *Proceedings of the 1st IEEE International Conference on Smart Grid Communications*, pp. 238–243, Gaithersburg, MD, 2010.

[9] F. D. Garcia and B. Jacobs, "Privacy-friendly energy-metering via homomorphic encryption," in *Proceedings of the 6th Workshop on Security and Trust Management (STM'10)*, pp. 226–238, Athens, 2010.

[10] G. Kalogridis, C. Efthymiou, S. Denic, T. Lewis and R. Capeda, "Privacy for smart meters: Towards undetectable appliance load signatures," in *Proceedings of the IEEE International Conference on Smart Grid Communication*, pp. 232–237, Gaithersburg, MD, 2010.

[11] F. Li, B. Luo and P. Liu, "Secure information aggregation for smart grids using homomorphic encryption," in *Proceedings of the 1st IEEE International Conference on Smart Grid Communications*, pp. 327–332, Gaithersburg, MD, 2010.

[12] Q. D. Li and Y. H. Zhou, "Research and application based on a. shamir's (t, n) threshold secret sharing scheme," in *Proceedings of the IEEE International Conference on Computer Science & Education (ICCSE'12)*, pp. 14–17, Melbourne, 2012.

[13] X. F. Li, "Smart energy meter data security protection technology," *Urban Construction Theory Research*, 2013.

[14] J. H. Lin, "Icpdas issued a new intelligent web-based electric meter concentrator," *DIGITIMES*, 2014.

[15] W. Liu, "Smart energy meter data security protection technology," *Power Supply Technologies and Applications*, 2013.

[16] D. Manivannan and P. Neelamegam, "An efficient key management scheme in multi-tier and multi-cluster wireless sensor networks," *International Journal of Network Security*, vol. 17, no. 6, pp. 651–660, 2015.

[17] S. R. Rajagopalan, L. Sankar, S. Mohajer and H. V. Poo, "Smart meter privacy: A uility-privacy framework," in *Proceedings of the IEEE International Conference on Smart Grid Communication*, pp. 190–195, Brussels, 2011.

[18] A. Shamir, "How to share a secret," *Communications of the ACM*, vol. 22, no. 11, pp. 612–613, 1979.

[19] X. X. Tian, C. F. Sha, X. L. Wang and A. Y. Zhou, "Privacy preserving query processing on secret share based data storage," in *Proceedings of International Conference on 16th Database Systems for Advanced Applications*, pp. 108–122, Hong Kong, China, 2011.

[20] X. X. Tian, X. L. Wang and A. Y. Zhou, "DSP re-encryption based access control enforcement management mechanism in DaaS," *International Journal of Network Security*, vol. 15, no. 1, pp. 28–41, 2013.

[21] D. Varodayan and A. Khisti, "Smart meter privacy using a rechargeable battery: Minimizing the rate of information leakage," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 1932–1935, Prague, 2011.

[22] Y. Wang and X. G. Peng, "Cryptanalysis of two efficient password-based authentication schemes using smart cards," *International Journal of Network Security*, vol. 17, no. 6, pp. 728–735, 2015.

[23] J. Q. Xie, "A practical key-sharing method," *Microcomputer Applications*, vol. 21, no. 6, 2005.

[24] 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.

[25] S. Zhang and Y. F. Li, "Smart energy meter data security protection technology," *Power Demand Side Management*, vol. 12, no. 2, 2010.

Xiuxia Tian, Professor, School of Computer Science and Technology in Shanghai University of Electric Power. She received her master degree from Shanghai Jiaotong University in 2005 and doctoral degree from Fudan University in 2011 respectively. Now she is a visiting scholar of two years working with group of Secure Machine

Learning (SecML) at Computer Science Division, University of California, Berkeley, email: xxtian@fudan.edu.cn. She has published more than 30 papers and some papers publish in international conferences and journals such as DASFAA, ICWS and SCN. Her main research interests: Database security, privacy preserving, applied cryptography, machine learning.

Lisha Li Graduate, School of Electronics and Information Engineering in Shanghai University of Electric Power. Her research interests mainly focus on the security and privacy protection for the smart meter.

Jinguo Li, Assistant Professor, School of Computer Science and Technology in Shanghai University of Electric Power. He received his bachelor degree and doctoral degree from Hunan University in 2007 and 2014 respectively. Email: lijg@shiep.edu.cn. He has published more than 10 papers and some papers are published in international conferences and journals such as WASA and SCN. His main research interests: Applied cryptography, cloud computing, network security.

Hongjiao Li, Associate Professor, School of Computer Science and Technology in Shanghai University of Electric Power. She received her master degree from Huazhong University of Science and Technology in 2002 and doctoral degree from Shanghai Jiaotong University in 2008. Email: hjli@shiep.edu.cn. She has published more than 30 papers in international conferences and journals. Her main research interests: Trust Computing, cloud computing and bigdata security, operating system security.

Chunhua Gu, Professor, School of Computer Science and Technology in Shanghai University of Electric Power. He received his master degree and doctoral degree from East China University of Science and Technology (ECUST) during 1988-2007. He was a visiting scholar at Florida International University in 2002 and University of Wisconsin at Madison in 2006. His main research interests: Smart grid security, cloud based data management, security calculation for smart grid user.