Comment on Enhanced DNA and ElGamal cryptosystem for secure data storage and retrieval in cloud

Chenglian Liu[1][0000−0002−9086−9740] and Sonia Chien-I Chen[2][0000−0002−6296−4943]

[1] Software Engineering Institute of Guangzhou, Guangzhou 510990, China
liuzl@mail.seig.edu.cn
[2] Qingdao University, Qingdao 266061, China
drsoniachen@qdu.edu.cn

Abstract. Thangavel and Varalakshmi proposed an enhanced DNA and ElGamal cryptosystem for secure data storage and retrieval in cloud. They modified ElGamal algorithm which it calls enhanced ElGamal cryptosystem. We prove that their enhanced ElGamal scheme, which does not require two random numbers by data owner. Although the attacker is unable to find out what message the data owner gave to the data user. However, the attackers can still confuse the issue of sending messages to data users. On the other hand, this scheme can not against insider attack, therefore it is insecure.

Keywords: Enhanced ElGamal Cryptosystem · Forgery Attack · Jamming Attac · Redundancy.

1 Introduction

Indian scholars Thangavel, Varalakshmi, Murrali, and Nithya [6] proposed a paper titled "Secure file storage and retrieval in cloud" in 2015, which discusses the research on secure file storage in the cloud based on the RSA algorithm. Chen and Liu [2], Zhong et al. [12] analyzed security for its scheme. Subsequently, Thangavel and Varalakshmi [10] continued to propose an article "An Enhanced and Secured RSA Key Generation Scheme (ESRKGS)"; Luy et al. [5] proved a security analysis for their scheme. In 2016, Thangavel and Varalakshmi [7] presented a new idea article "Enhanced Schmidt-Samoa cryptosystem for data confidentiality in cloud computing"; Chen et al. [1] implemented the algorithm into mobile app., and also proved that scheme insecure. On December 2017, Thangavel and Varalakshmi [8] sustained to publish their article "Improved secure RSA cryptosystem for data confidentiality in cloud", Liu and Hsu [4] proved that the algorithm was flawed and insecure. On June 2018, Thangavel and Varalakshmi [9] modified their research topic from RSA to ElGamal and DNA cryptosystem, and then published the article "Enhanced DNA and ElGamal cryptosystem for secure data storage and retrieval in cloud". The RSA

cryptosystem is based on the difficulty of factorization, while the ElGamal cryptosystem is based on the difficulty of solving discrete logarithms. Although these two algorithms are different, we prove [9] cannot resist man in the middle attacks and forgery attacks, when we use mathematical form proof. Thus, that scheme insecure. Please see Figure 1.

Table 1. Thangavel and Varalakshmi's Related Literature

Item	Year	Original Research	Follow Research	Notes
1	2015	Thangavel et al. [6]	Chen and Liu [2]	RSA series
2			Zhong et al. [12]	RSA series
3	2015	Thangavel et al. [10]	Lüy et al. [5]	RSA series
4	2016	Thangavel & Varalakshm [7]	Chen et al. [1]	RSA series
5	2017	Thangavel & Varalakshmi [8]	Liu and Shu [4]	RSA series
6	2018	Thangavel & Varalakshmi [9]	This study	DNA and ElGamal

2 Review of Enhanced ElGamal Cryptosystem

In this section, the authors would like to introduce ElGamal algorithm of encryption and decryption without ElGamal digital signature scheme, and the other improvement algorithm, namely enhanced ElGamal algorithm.

2.1 The ElGamal Algorithm

We assume two parties Alice and Bob, if Alice wants to encrypt message m for Bob. She does follow steps.

Key Generation Phase:

Step 1. Bob chooses a large prime p, and a primitive root $g \in \mathbb{Z}_{\mathrm{I}}^*$.

Step 2. Bob randomly chooses his secret key y where $1 < y < p-1$ and $\gcd(y, p-1) = 1$, then find B,

$$B \equiv g^y \pmod{p}. \tag{1}$$

Step 3. Bob announces public parameters $\{p,\ g,\ B\}$, and keep secret key y.

Encryption Phase:

Step 1. Alice obtained public parameters $\{p,\ g,\ B\}$.

Step 2. Alice randomly selects her secret key x where $1 < x < p-1$ and $\gcd(x, p-1) = 1$, then computes

$$A \equiv g^x \pmod{p}. \tag{2}$$

Step 3. Alice digitizes the message $m \in [0,\ p-1]$ and computes cipher c where

$$c \equiv B^x \cdot m \pmod{p}. \tag{3}$$

Step 4. Alice sends $\{A,\ c\}$ to Bob.

Decryption Phase: When Bob received $\{A,\ c\}$, he then recovers message m by follow steps.

Step 1. Bob assumes w where

$$w = (p - 1 - y) \tag{4}$$

Step 2. To find m where

$$m \equiv A^w \cdot c \pmod{p}. \tag{5}$$

Proof.

$$\begin{aligned} m &\stackrel{?}{\equiv} A^w \cdot c \bmod p \\ &\equiv (g^x)^{p-1-y} \cdot c \bmod p \\ &\equiv (g^{p-1})^x \cdot g^{-xy} \cdot (g^y)^x \cdot m \bmod p \\ &\equiv m \bmod p. \end{aligned} \tag{6}$$

The ElGamal algorithm of encryption and decryption is shown in Figure 1.

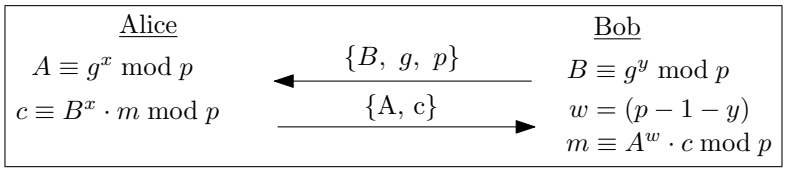

Figure 1. The ElGamal Encryption and Decryption Protocol.

2.2 The Enhanced ElGamal Algorithm

Thangavel and Varalakshmi modified the ElGamal algorithm, say enhanced ElGamal cryptosystem, the algorithm describes as following. We assume two parties Data User (DU) and Data Owner (DO).

Key Generation Phase:

Step 1. DU chooses a large prime p and two primitive roots g_1 and g_2 in $\mathbb{Z}_{\text{ı}}^*$.

Step 2. DU computes d where

$$d \equiv (g_1 \cdot g_2)^{-1} \pmod{p}. \tag{7}$$

Step 3. DU randomly chooses an integer x such that $1 < x < p-1$, and computes y where

$$y \equiv (g_1 \cdot g_2)^x \pmod{p}. \tag{8}$$

Step 4. DU publishes public parameters $\{g_1,\ y,\ p\}$, and keep secret $\{x,\ g_2,\ d\}$.

Encryption Phase:
Do digitizes message m such that $1 < m < p-1$.

Step 1. DO chooses two random numbers $k_1,\ k_2$ such that $1 < k_1,\ k_2 < p-1$.
Step 2. DO also chooses a shared secret key k_3 such that $1 < k_3 < p-1$.
Step 3. DO computes one-time key k where

$$k \equiv (k_1)^{k_2} \cdot y^{k_3} \pmod{p}. \tag{9}$$

Step 4. DO computes C_1 where

$$C_1 \equiv g_1^{k_3} \pmod{p}. \tag{10}$$

Step 5. DO computes C_2 where

$$C_2 \equiv k_1^{k_2} \pmod{p}. \tag{11}$$

Step 6. DO computes C_3 where

$$C_3 \equiv k \cdot m \cdot y \pmod{p}. \tag{12}$$

Step 7. DO secretly shares k_3 with DU, and sends $C = \{C_1,\ C_2,\ C_3\}$ to DU.

Decryption Phase:
If DU wants to recover message m as following:

Step 1. Compute one-time key k where

$$k \equiv C_1^x \cdot C_2 \cdot g_2^{k_3 \cdot x} \pmod{p}. \tag{13}$$

Step 2. Compute K^{-1} such that

$$k^{-1} \cdot k^1 \equiv 1 \pmod{p}. \tag{14}$$

Step 3. Compute m where

$$m' \equiv k^{-1} \cdot C_3 \cdot d^x \pmod{p}. \tag{15}$$

Proof.

$$\begin{aligned} m' &\stackrel{?}{\equiv} k^{-1} \cdot C_3 \cdot d^x \pmod{p} \\ &\equiv C_1^x \cdot C_2^{-1} \cdot g_2^{-k_3 \cdot x} \cdot k \cdot m \cdot y \cdot d^x \pmod{p} \\ &\equiv m \cdot (g_1 g_2)^x \cdot ((g_1 g_2)^{-1})^x \pmod{p} \\ &\equiv m \pmod{p}. \end{aligned} \tag{16}$$

We get the proof. The detailed flow of Enhanced ElGamal protocol as show in Figure 2.

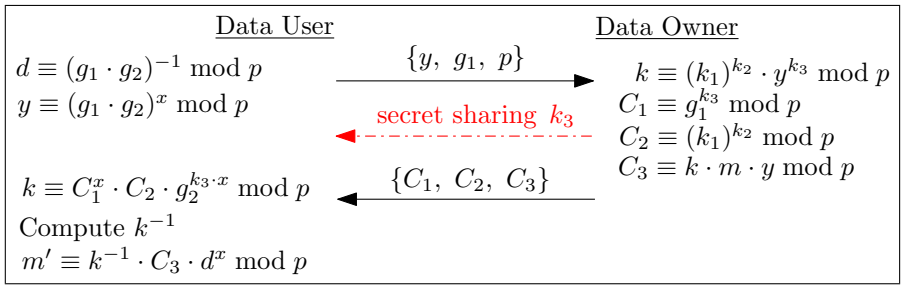

Figure 2. The Flow of Enhanced ElGamal Scheme.

3 Our Comment

3.1 Improved Algorithm

By Figure 2, the two random keys k_1 and k_2 generated by the DO which it did not play its role. If we remove the parameter C_2, it means that the entire algorithm has removed the three redundant parameters C_2, k_1, and k_2. Deleting C_2 not only speeds up calculations, also performs the transmission rate. The result is never changed. We rewrite Equation (9) as

$$k \equiv y^{k_3} \bmod p. \tag{17}$$

in encryption phase. And rewrite Equation (13) as

$$k \equiv C_1^x \cdot g_1^{k_3 \cdot x} \bmod p \tag{18}$$

in decryption phase.

Proof.

$$\begin{aligned} m' &\stackrel{?}{\equiv} k^{-1} \cdot C_3 \cdot d^x \bmod p \\ &\equiv k^{-1} \cdot k \cdot m \cdot y \cdot d^x \bmod p \\ &\equiv m \cdot y \cdot d^x \bmod p \\ &\equiv m \bmod p. \end{aligned} \tag{19}$$

The improvement protocol is shown in Figure 3.

3.2 Insider Attack

We assume the party's long-term secret shared key k_3 is compromised. The insider who is adversary Eve, if she obtained k_3 from DO's group, or she fake a valid k_3' instead of k_3 such as Equation (20). Eve secret shared k_3' before she returned $\{C_1,\ C_3\}$ to DU. This scheme can not against KCI (Key Compromise Impersonation) attack [3, 11], please see Figure 4.

$$k \equiv y^{k_3'} \bmod p. \tag{20}$$

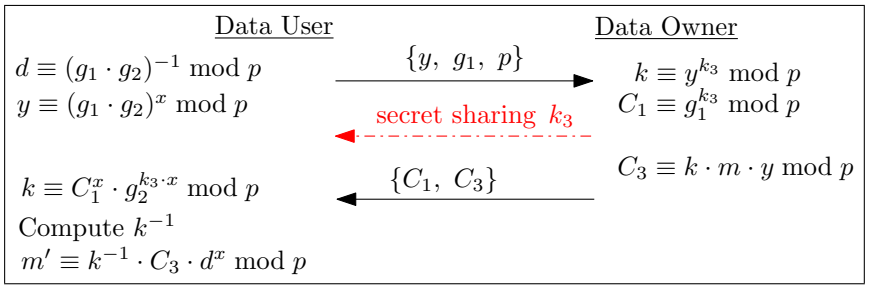

Figure 3. The improvement of Enhanced ElGamal Scheme.

Proof.

$$\begin{aligned} m' &\stackrel{?}{\equiv} k^{-1} \cdot C_3 \cdot d^x \bmod p \\ &\equiv \underline{k^{-1}} \cdot \underline{k} \cdot m \cdot y \cdot d^x \bmod p \\ &\equiv m \cdot \underline{(g_1 g_2)^x} \cdot \underline{(g_1 g_2)^{-x}} \bmod p \\ &\equiv m \bmod p. \end{aligned} \tag{21}$$

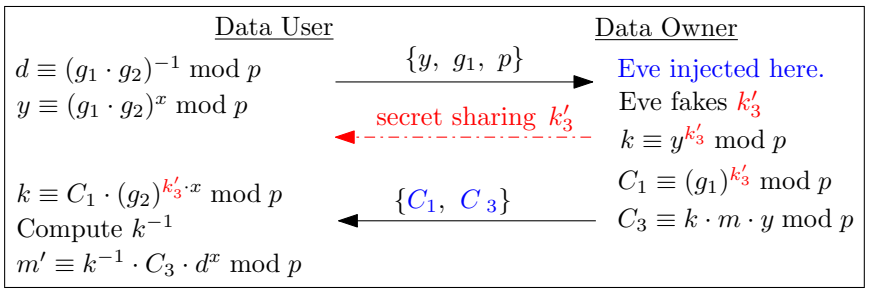

Figure 4. The insider attacks Enhanced ElGamal Scheme.

3.3 Outsider Attack

As known from algorithm, the parameters $\{C_1,\ C_3\}$ are announced by public network channel. As known from algorithm, the parameters $\{C_1,\ C_3\}$ are published. We suppose an outside attacker Eve who wants to forge these parameters in the channel. She simply fakes two equations by following steps. The detailed flow of conception is shown in Figure 5.

Encryption Phase:

Step 1. Eve randomly selects an integer $s \in 1 < s < p-1$, and find k' such as

$$k' \equiv k \cdot s \bmod p. \tag{22}$$

Step 2. Eve finds C_3' which it satisfied

$$C_3' \equiv k' \cdot m \cdot y \cdot s^{-1} \bmod p. \tag{23}$$

Step 3. Eve sends $\{C_1,\ C_3'\}$ to DU.

Decryption Phase:
When DU received parameters $\{C_1,\ C_3'\}$ by DO (Actually, it was Eve who impersonated the Data Owner). If DU wants to recover message m, he may does follow steps.

Step 1. DU computes k where

$$k \equiv C_1^x \cdot (g_2)^{k_3 \cdot x} \bmod p. \tag{24}$$

Step 2. DU computes K^{-1} where

$$k^1 \cdot k^{-1} \equiv 1 \bmod p. \tag{25}$$

Step 3. DU computes m' where

$$m' \equiv k^{-1} \cdot C_3' \cdot d^x \bmod p. \tag{26}$$

Proof.

$$\begin{aligned} m' &\stackrel{?}{\equiv} k^{-1} \cdot C_3' \cdot d^x \bmod p. \\ &\equiv k^{-1} \cdot k' \cdot m \cdot y \cdot s^{-1} \cdot d^x \bmod p \\ &\equiv \underline{k^{-1}} \cdot \underline{k} \cdot \underline{s} \cdot m \cdot y \cdot \underline{s^{-1}} \cdot d^x \bmod p \\ &\equiv m \cdot y \cdot d^x \bmod p \\ &\equiv m \cdot \underline{(g_1 g_2)^x} \cdot \underline{(g_1 g_2)^{-x}} \bmod p \\ &\equiv m \bmod p. \end{aligned} \tag{27}$$

We finished the proof, and please see Figure 5.

3.4 Jamming Attack

Although Eve cannot find out what DO sent to DU, Eve can confuse the content when she sent to DU. Because the original m is generated in DO side (Eve), and it never sent to DU. Thus, DU cannot recognize that m. That is real reason why Eve can confuse DU by fake m'. This is one problem to ElGamal Encryption/Decryption algorithm, please see Figure 6.
Encryption Phase:

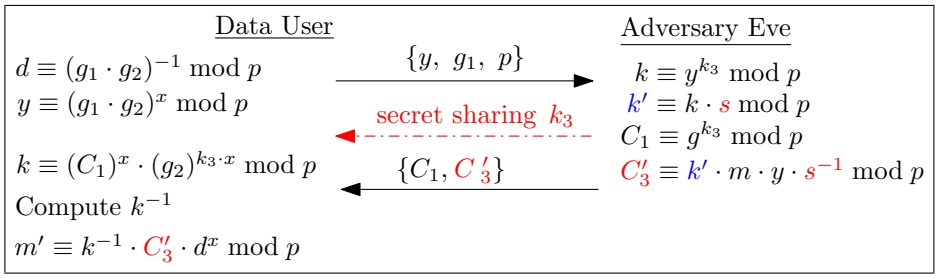

Figure 5. The outsider attacks Enhanced ElGamal Scheme.

Step 1. Eve randomly selects k' where $1 < k' < p-1$, and satisfies

$$C'_1 \equiv C_1 \cdot k' \bmod p. \tag{28}$$

Step 2. Eve computes C'_3 such as

$$C'_3 \equiv k' \cdot m' \cdot y \bmod p. \tag{29}$$

Step 3. Eve sends $\{C'_1, C'_3\}$ to DU.

Decryption Phase: When DU received $\{C'_1, C'_3\}$ from DO (Actually, it was Eve.), if he want to recover m, he then does follow steps.

Step 1. DU Computes k such as

$$k \equiv C'_1 \cdot g_2^{k_3 \cdot x} \bmod p. \tag{30}$$

Step 2. DU computes k^{-1} where

$$k^1 \cdot k^{-1} \equiv 1 \bmod p. \tag{31}$$

Step 3. DU computes m'' such as

$$m'' \equiv k^{-1} \cdot C'_3 \cdot d^x \bmod p. \tag{32}$$

Proof.

$$\begin{aligned} m'' &\stackrel{?}{\equiv} k^{-1} \cdot C'_3 \cdot d^x \bmod p. \\ &\equiv k^{-1} \cdot k' \cdot m' \cdot y \cdot d^x \bmod p \\ &\equiv k^{-1} \cdot k' \cdot m' \cdot \underline{y} \cdot \underline{d^x} \bmod p \\ &\equiv k^{-1} \cdot k' \cdot m' \bmod p \\ &\not\equiv m' \bmod p \end{aligned} \tag{33}$$

Because DU cannot recognize original m, he therefore does not know $m'' \not\equiv m' \bmod p$ by Equation (33). Thus, it does not matter DU received real or fake parameters $\{C_1, C_3\}$ upon on insider attack, jamming attack or outsider attack.

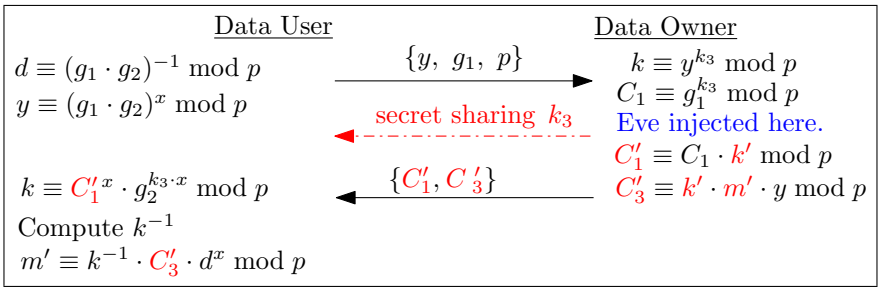

Figure 6. The Jamming Attack to Enhanced ElGamal Scheme.

4 Conclusions

In the paper the authors extensively analyzed the flaws in the enhanced ElGamal algorithm, and also point out how cancel redundant useless parameters such as C_2, k_1 and k_2 to improve performance and computation. On the other hand, the author also pointed out three types of attacks such as insider attack, outsider attack and jamming attack. That is to say, the enhanced ElGamal algorithm proposed by Thangavel and Varalakshmi's version, cannot against forgery attack. We have proven our claims by formal mathematical proof.

References

1. Chen, H., Liu, C., Huang, J., Chen, S.C.I.: Security analysis of enhanced schmidt-samoa scheme. In: AIP Conference Proceedings. vol. 2186. AIP Publishing (2019)
2. Chen, S.C.I., Liu, C.: Comment on secure file storage and retrieval in cloud. In: AIP Conference Proceedings. vol. 2293, p. 410002
3. Gorantla, M.C., Boyd, C., González Nieto, J.M.: Modeling key compromise impersonation attacks on group key exchange protocols. In: Jarecki, S., Tsudik, G. (eds.) Public Key Cryptography–PKC 2009. pp. 105–123. Springer Berlin Heidelberg, Berlin, Heidelberg (2009)
4. Liu, C., Hsu, C.W.: Comment on" improved secure rsa cryptosystem (isrsac) for data confidentiality in cloud". International Journal of Network Security **21**(4), 709–712 (2019)
5. Lüy, E., Karatas, Z.Y., Ergin, H.: Comment on "an enhanced and secured rsa key generation scheme (esrkgs)". Journal of Information Security and Applications **30**, 1–2 (2016)
6. Murugan, T., Varalakshmi, P., Murrali, M., Nithya, K.: Secure file storage and retrieval in cloud. International Journal of Information and Computer Security **7**, 177 (01 2015). https://doi.org/10.1504/IJICS.2015.073025
7. Thangavel, M., Varalakshmi, P.: Enhanced schmidt-samoa cryptosystem for data confidentiality in cloud computing. International Journal of Information Systems and Change Management **8**(2), 160–188 (October 2016)
8. Thangavel, M., Varalakshmi, P.: Improved secure RSA cryptosystem for data confidentiality in cloud. International Journal of Information Systems and Change Management **9**(4), 261–277 (January 2017)

9. Thangavel, M., Varalakshmi, P.: Enhanced dna and elgamal cryptosystem for secure data storage and retrieval in cloud. Cluster Computing **21**(2), 1411–1437 (June 2018)
10. Thangavel, M., Varalakshmi, P., Murrali, M., Nithya, K.: An enhanced and secured rsa key generation scheme (esrkgs). Journal of Information Security and Applications **20**, 3–10 (2015)
11. Tian, H., Chen, X., Ding, Y.: Analysis of two types deniable authentication protocols. International Journal of Network Security **9**(3), 242–246 (November 2009)
12. Zhong, Z., Liu, H., Chen, S.C., Liu, C., Gardner, D.: Comment of secure file storage and retrieval in cloud based on mrsa cryptographic algorithm. In: 2020 9th International Conference on Industrial Technology and Management (ICITM). pp. 261–264. IEEE (2020)