Research and Design of an Improved ElGamal Digital Signature Algorithm

IOP Conf. Series: Materials Science and Engineering **569** (2019) 052041 doi:10.1088/1757-899X/569/5/052041

Research and Design of an Improved ElGamal Digital Signature Algorithm

Yi Fang*, Linhu Cong and Jianqiu Deng

Naval Aeronautical University, Yantai, Shandong, 264001, China

*Corresponding author's e-mail: 395510796@qq.com

Abstract. Firstly, the digital signature technology and its basic principle are introduced. Then, an ElGamal digital signature algorithm based on discrete logarithm is described, and two aspects of security and signature efficiency are improved on the basis of this algorithm, and the correctness, security and complexity of the improved scheme are analyzed. The results show that the improved ElGamal algorithm achieves the purpose of improvement.

1. Introduction

Digital signature technology is one of the core technologies of information and data security, which can realize important requirements in data transmission such as identity authentication, data integrity protection, tamper-proof, impersonation proof and non-repudiation. Since the idea of public key cryptography was put forward [1], digital signature technology based on public key cryptography has emerged. Development has been produced based on the signature, based on the elliptic curve discrete logarithm of signature[2], based on the identification protocol's signature[3], blind signature[4], proxy signature[5], multi digital signature[6] and ring signature signature scheme[7]. With the advent of the era of big data, digital signature technology will play a more important role.

ElGamal algorithm is a non-deterministic digital signature algorithm based on discrete logarithm, which is widely used[8]. In this paper, the existing security and execution efficiency of the original ElGamal algorithm are improved, and its overall analysis is carried out.

2. Digital signature

In public key cryptosystem, the user's secret key is a pair of public key and private key, the private key is kept in secret and the public key is made public. Since the public key cannot deduce the private key, the public key will not compromise the security of the private key. Digital signature means that the signer encrypts a message with its own private key, and if the verifier can decrypt the message correctly with the public key of the signer, it is determined that the message is digitally signed by the signer. Generally speaking, the digital signature scheme is composed of a 5-tuple $(M, S, K, SIGN, VRFY)$ [9] and meets the following conditions:

- M is a finite set of possible messages.
- S is a finite set of possible signatures.
- the key space K is a finite set of possible secret keys.
- for each $k = (k_s, k_v) \in K$, there is a signature algorithm $Sign_{k_s} \in SIGN$ and a verification algorithm $Vrfy_{k_v} \in VRFY$. Each $Sign_{k_s}: M \to S$ and verification function $Vrfy_{k_v}: M \times S \to \{True, False\}$ is a function that satisfies the following equation for any message $m \in M$ and any signature $s \in S$:

$$Vrfy(m,s)=\begin{cases} True, s = Sign_{k_s}(m) \\ False, s \neq Sign_{k_s}(m) \end{cases} \quad (1)$$

For each $k \in K$, the sum of the functions $Sign_{k_s}$ is $Vrfy_{k_v}$ a polynomial time computable function. $Vrfy_{k_v}$ is a public function that k_v is the public key (verifies the secret key); $Sign_{k_s}$ is a cryptographic function that k_s is a private key (signed secret key) that needs to be kept secret. The general process of digital signature is shown in figure 1.

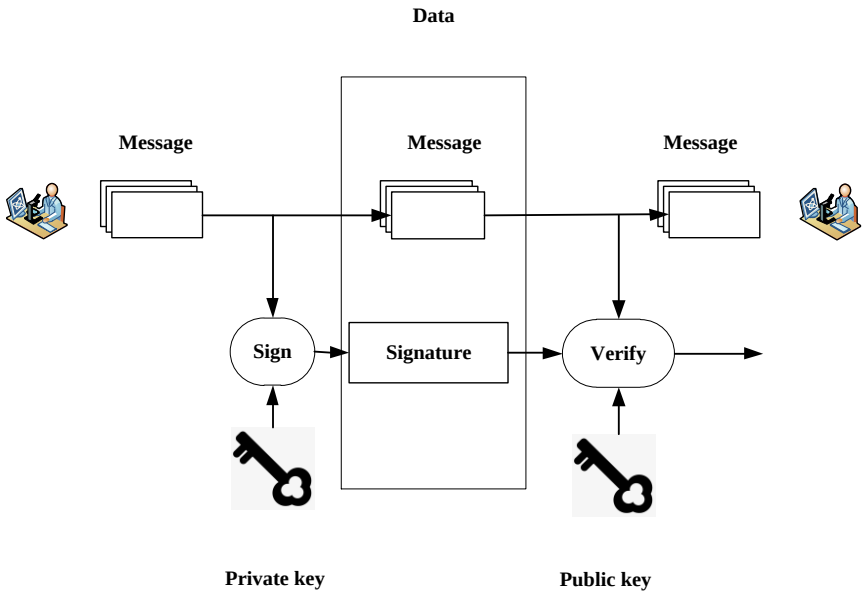

Figure 1. Digital signature process

3. ElGamal digital signature algorithm based on discrete logarithm

Based on the classification of mathematical problems, digital signature algorithms can be divided into digital signature algorithm based on discrete logarithm problem, signature algorithm based on large integer prime factorization, signature algorithm based on elliptic curve discrete logarithm problem and signature algorithm based on quadratic residual problem[10]. ElGamal algorithm is a signature algorithm based on the discrete logarithm problem, and it is a non-deterministic signature algorithm, that is, for the same message, there will be different legal digital signatures due to different random Numbers. ElGamal algorithm mainly includes three parts: parameter and secret key generation, signature algorithm and verification algorithm[11].

3.1. Parameter and secret key generation

Let P is a large prime number, and it is difficult to solve the discrete logarithm in Z_p. Then select a generator $g \in Z_p^*$ and a random number $x \in Z_{p-1}^*$, calculate

$$y = g^x \bmod P \quad (2)$$

The public key is (y, g, P), and the private key is x.

3.2. Signature algorithm

Suppose the message to be signed is M, and the signer selects a secret random number $k \in Z_P^*$ to calculate

$$r = g^k \bmod P \tag{3}$$

$$s = \left(h(m) - xr\right)k^{-1} \bmod (P-1) \tag{4}$$

(s,r) is the signature of M, where h is the Hash function.

3.3. Verification algorithm

The receiver of the signature has the public key (y,g,P), and after receiving the signature (s,r) of the message M, first calculate $h(m)$ and then verify

$$y^r r^s \equiv g^{h(m)} \bmod (P) \tag{5}$$

If formula (4) is true, the digital signature is valid; otherwise, the signature is invalid.

4. Improvement and analysis of ElGamal digital signature algorithm

4.1. ElGamal algorithm improvement

As can be seen from the description of ElGamal's digital signature algorithm above, the security of this algorithm is largely dependent on the selection of random Numbers. Different random Numbers will produce different digital signatures, which brings great difficulty to attack. At the same time, most of the attack targets of the ElGamal algorithm are random Numbers, because the attack on the secret key is more difficult, and the random number is less related to the secret key, so the attack is easy to realize. Therefore, random number selection is an important guarantee for the security of the algorithm. Formula (4) the Hash function also can to a large extent, ensure that the signature is not easy to be cracked, and the Hash function in the original ElGamal digital signature algorithm input plaintext only signing messages, if clear to reveal the Hash function will not exist safety protection ability, so the Hash function to increase the amount of data data can also be from a certain extent, improve the ability against the attack. In the original ElGamal algorithm, modular inverse operation exists, which greatly reduces the efficiency of the algorithm. Therefore, on the premise of ensuring the correctness of signature verification, there are two main ideas for improvement: improving the selection of random Numbers and increasing the input data of Hash function to make the original algorithm more secure; The efficiency of the original algorithm is improved by improving the modular inversion operation and changing the way of public key generation. Specific improvement schemes for improving security are as follows:

After selecting a random number k, select a random number n. In the process of signature, a random number n signature formula is added on the basis of the existing signature formula (3)

$$t = g^n \bmod P \tag{6}$$

On the basis of formula (6) added, the private key and the message plaintext together are taken as the input of Hash function, then formula (4) becomes

$$s = \left(h(m \,||\, y) - xr - xt\right)n^{-1} \bmod (P-1) \tag{7}$$

The verification equation is

$$y^r r^t t^s \equiv g^{h(m||y)} \bmod (P) \tag{8}$$

From the perspective of efficiency, the inverse operation in formula (8) is usually realized by the extended Euclidean algorithm in the computer, and the calculation process is complex. The specific improvement plan is as follows:

Use $h(m)$ as the private key and change the public key generation formula (2) to

$$e = g^{h(m)} \bmod P \tag{9}$$

At this point, the public key is e, and formula (2) is still retained as the intermediate process for the improvement of the next modular inverse operation.

The correlation between signature s and modular inverse operation in signature equation (8) is removed, and equation (8) is changed into

$$s = \left(t + nr + h\left(m\right)\right) \bmod \left(P - 1\right) \tag{10}$$

$$\lambda = \left(k - nr - xy\right) \bmod \left(P - 1\right) \tag{11}$$

The function of the newly added formula (11) is to replace the correlation of the inverse modular operation of the signature s in formula (8). The resulting digital signature is $\left(r, s, y, \lambda\right)$.

According to equations (10) and (18), the verification equation is changed to

$$g^{\beta} \bmod P \equiv re \bmod P \tag{12}$$

$$\beta = \left(s - t + xy + \lambda\right) \tag{13}$$

The public key generation equation of the improved ElGamal digital signature algorithm is (9), the signature equation is (2), (3), (6), (10) and (11), and the verification equation is (12).

4.2. The correctness analysis of the improved ElGamal algorithm

The correctness verification process of the improved ElGamal algorithm proposed in this paper is as follows:

According to equations (12) and (13)

$$g^{\beta} \bmod P \equiv g^{\left(s-t+xy+\lambda\right)} \bmod P \tag{14}$$

Assuming the signature is correct, the signature equation (10) is established. According to equations (10), (11) and (14)

$$g^{\beta} \bmod P \equiv g^{\left(t+nr+h(m)-t+xy-nr-xy\right)} \bmod P \equiv g^{\left(h(m)+k\right)} \bmod P \tag{15}$$

Finally, according to equations (3), (9) and (15)

$$g^{\beta} \bmod P \equiv g^{h(m)} g^{k} \bmod P \equiv re \bmod P \tag{16}$$

The obtained equation (16) is consistent with the verification equation (12), proving that the improved ElGamal algorithm can correctly verify the legitimate digital signature.

4.3. Security analysis of the improved ElGamal algorithm

The attacks on digital signature algorithms are mainly direct attacks on private keys and forged signatures. In this paper, security analysis is carried out by simulating various attack modes of attackers.

If the attacker wants to directly solve the private key from the public key, it can be seen from the formula (9) of public key generation that to solve the private key, it needs to solve discrete logarithm problem $h\left(m\right) = \log_{g} e$, which is very difficult to solve.

If the attacker intercepts the signature group $\left(x, y, r, t, s, \lambda\right)$ and wants to obtain the private key according to the signature group, then according to signature formulas (10) and (11), the attacker needs to solve three unknowns $\left(n, k, h\left(m\right)\right)$ from these two equations, so the private key cannot be obtained.

If the attacker had intercepted sends the signature of the message, want to replace the message in a way that signature forgery, on the premise of public and private key security, according to formula (10), joined the Hash function values of the real message signatures, and Hash function is an important feature: it is difficult to find two Hash value of the same message, is replaced so messages unless exactly the same as that of the original message, otherwise not in accordance with the original signature of the digital signature.

If the attacker had intercepted sends the signature of the message, want to replace the random manner signature forgery, assuming that the attacker to use to replace the random number, then according to the formula (3), can be obtained by formula (9) and (10), also need to be out of the two

equations of four unknown to the forgery of signature, so the signature of the attacker is invalid, is rejected in the validation process.

If the attacker had intercepted sends the signature of the message, want to replace the random manner signature forgery, assuming that the attacker to use w to replace the random number k, then according to the formula (3), r_w can be found by the formula (10) and (11), also need to be out of the two equations of four unknown to the forgery of signature, so the signature of the attacker is invalid, is rejected in the validation process.

According to the security analysis, it can also be seen that improving the random numbers k and n in the ElGamal algorithm is an important means to ensure the security of the algorithm. Therefore, $\gcd(k,n)=1$ must be guaranteed when selecting and must be stored in secret so as not to be disclosed.

4.4. The complexity analysis of the improved ElGamal algorithm

The Times of various operations involved in the original ElGamal algorithm are shown in table 1, and The Times of various operations involved in the improved ElGamal algorithm proposed in this paper are shown in table 2.

Table 1. Original ElGamal algorithm operation types and times.

	Public key generation	Signature	Validation	Total
Exponent times	1	1	3	5
Modular inversion times	0	1	0	1
The dot product times	0	1	1	2
Hash times	0	1	1	2

Table 2. Improved ElGamal algorithm operation types and times.

	Public key generation	Signature	Validation	Total
Exponent times	1	3	1	5
Modular inversion times	0	0	0	0
The dot product times	0	2	2	4
Hash times	1	0	0	1

According to the comparison between table 1 and table 2, both schemes used 5 exponential operations. In the improved ElGamal algorithm, the number of modular inversion is 0, and modular inversion is the most complex of the four operations. Therefore, reducing the number of modular inversion can effectively improve the efficiency of the algorithm. Compared with the original algorithm, the improved ElGamal algorithm takes more than two dot product operations. The dot product operation is a very fast operation, which takes much less time than the modular inverse operation. At the same time, it also reduces one Hash operation.

5. Conclusion

This paper mainly improves the ElGamal digital signature algorithm from the aspects of security and algorithm execution efficiency. In terms of security, the number of random Numbers is increased, and the Hash function is introduced when the public key is generated. In terms of execution efficiency, the inverse modular operation is reduced by adding auxiliary equations. Finally, the analysis results show that the improved ElGamal algorithm proposed in this paper has higher security and more efficient execution efficiency, achieving the purpose of improvement.

References

[1] Diffie W, Hellman M. New direction in cryptography[J]. IEEE Transaction on Information Theory, 1976, 6(22):644-654.

[2] Xu Z.F., Zeng K, Zhou F.C. Anonymous Electronic Voting Scheme Based on Time-released Encryption and Digital Signature[J]. Computer Applications and Software, 2016, 33(12):325-328.
[3] Cao Y. ElGamal Multiple Digital Signature Scheme Based on Identity[J]. Bulletin of Science and Technology, 2015, 31(05):197-199.
[4] Zhao Z.G. Certificateless Partially-blind Signature Scheme with Provable Security[J]. Journal of University of Electronic Science and Technology of China, 2016, 45(05):812-818.
[5] Ge L.X., Li X, He M.X., et al. Improved Certificateless Proxy Signature Scheme[J]. Computer Engineering and Applications, 2017, 53(08):92-94.
[6] Zhang J.H., Xiao H, Wang J.L. Efficient Identity-based RSA Multi-signature Scheme[J]. Journal of Chinese Computer Systems. 2018, 39(09):1978-1981.
[7] Jia X.Y., He D.B., Xu Z.Y., et al. An Efficient Identity-based Ring Signature Scheme over a Lattice[J]. Journal of Cryptologic Research, 2017, 4(04):392-404.
[8] Li L.J., Guo Y.J. An Improved ElGamal Digital Signature Scheme[J]. Computer Engineering & Science, 2016, 38(06):1097-1102.
[9] Zhao X. Overview of Digital Signature[J]. Computer Engineering and Design, 2006, 27(02):195-197.
[10] Zheng D, Zhao Q.L., Zhang Y.H. A Brief Overview on Cryptography[J]. Journal of Xi'an University of Posts and Telecommunications, 2013, 18(06):1-10.
[11] Elgamal T. A Public Key Cryptosystem And a Signature Sheme Based on Discrete Logarithms[J]. IEEE Trans information Theory, 1985, 31(04):469-472.