Survey of Various Homomorphic Encryption algorithms and Schemes

Payal V. Parmar
Dept. Computer Sci & Eng.
Shri S'ad Vidya Mandal
Institute of Technology
Bharuch,India.

Shraddha B. Padhar
Dept. Computer Sci & Eng.
Shri S'ad Vidya Mandal
Institute of Technology
Bharuch,India.

Shafika N. Patel
Dept. Computer Sci & Eng.
Shri S'ad Vidya Mandal
Institute of Technology
Bharuch,India.

Niyatee I. Bhatt
Dept. Computer Sci & Eng.
Shri S'ad Vidya Mandal
Institute of Technology
Bharuch,India.

Rutvij H. Jhaveri
Dept. Computer Sci & Eng.
Shri S'ad Vidya Mandal
Institute of Technology
Bharuch,India.

ABSTRACT

Homomorphic encryption is the encryption scheme which means the operations on the encrypted data. Homomorphic encryption can be applied in any system by using various public key algorithms. When the data is transferred to the public area, there are many encryption algorithms to secure the operations and the storage of the data. But to process data located on remote server and to preserve privacy, homomorphic encryption is useful that allows the operations on the cipher text, which can provide the same results after calculations as the working directly on the raw data. In this paper, the main focus is on public key cryptographic algorithms based on homomorphic encryption scheme for preserving security. The case study on various principles and properties of homomorphic encryption is given and then various homomorphic algorithms using asymmetric key systems such as RSA, ElGamal, Paillier algorithms as well as various homomorphic encryption schemes such as Brakerski-Gentry-Vaikuntanathan (BGV), Enhanced homomorphic Cryptosystem (EHC), Algebra homomorphic encryption scheme based on updated ElGamal (AHEE), Non-interactive exponential homomorphic encryption scheme (NEHE) are investigated.

General Terms

Security, Homomorphic Encryption Algorithms, Homomorphic Encryption Schemes

Keywords

Cryptography, Homomorphic Encryption, Paillier algorithm, RSA, ElGamal, BGV, EHC, NEHE, AHEE

1. INTRODUCTION

Security is the prime requirement because of the increasing usage of the internet or public cloud for storing the data. Security is needed for preserving the integrity, confidentiality, availability of the information system resources [1]. There can be storage of the data in the encrypted format in any database but if the operations or the computations on the encrypted data are required to be performed then it is the necessary to decrypt those data but the decrypted data are not secure any more thus, a new idea of the cryptosystem was proposed that allows the direct computation on the encrypted data. This concept is called "privacy homomorphism" [2]. However, decryption is not performed; the result obtained is same as computations on plaintext. While exclusively manipulating encrypted data, implicit additions and multiplications on plaintext values can be performed by the workers by using homomorphic encryption [3].

There are two types of the cryptosystems public key cryptosystem and symmetric cryptosystem [4]. IDEA, DES, AES etc are the symmetric key algorithms and RSA, EL-Gamal etc. are various asymmetric cryptosystem [5].

In the section 2, theoretical background is given with the basic concepts of the homomorphic encryption. All four functions of homomorphic encryption are explained. The additive and multiplicative properties of the homomorphic encryption are described with the examples (Paillier, RSA and ElGamal). In the section 3, various homomorphic encryption schemes are described. These all schemes have property of mixed homomorphic encryption. After then, the comparison of various homomorphic encryption schemes and algorithms are given which gives the overall idea about all algorithms and schemes. In section 4, a survey paper is concluded.

2. THEOREITICAL BACKGROUND

Security is the prime requirement because cyber crimes are increasing nowadays. Today, the public environment is needed to be secure for preserving the security of data. There are many private environments are available but to store the data over those environments can be expensive than public area. Hence, everyone is convenient to store the data on public cloud i.e. Internet. There are many encryption algorithms are available [6]. Using them, the secure environment is created. Homomorphic encryption enables that secure environment in which the operations can be done on the already encrypted data and the same result can be obtained as on original data [7]. There are many homomorphic encryption schemes are described in this paper which makes use of this approach.

2.1 History of Homomorphic Encryption

In 1978 Ronald Rivest, Leonard Adleman and Michael Dertouzos suggested for the first time the concept of Homomorphic encryption [8]. Since then, little progress has been made for 30 years. The encryption system of Shafi Goldwasser and Silvio Micali was proposed in 1982 was a provable security encryption scheme which reached a remarkable level of safety, it was an additive Homomorphic encryption, but it can encrypt only a single bit. In the same concept in 1999 Pascal Paillier was also proposed a provable security encryption system that was also an additive Homomorphic encryption. Few years later, in 2005, Dan Boneh, Eu-Jin Goh and Kobi Nissim [9] invented a system of provable security encryption, with which unlimited number of additions but only one multiplication can be performed.

Homomorphic encryption is the encryption on the already encrypted data rather than on the original data with providing the result as it is done on the plain text. The complex mathematical operations can be performed on the cipher text without changing the nature of the encryption [10].

2.2 Functions of Homomorphic Encryption

Homomorphic Encryption H is a set of four functions [11] as shown in figure 1.

H = {Key Generation, Encryption, Decryption, Evaluation }

1. Key generation: client will generate pair of keys public key pk and secret key sk for encryption of plaintext.
2. Encryption: Using secret key sk client encrypt the plain text PT and generate Esk(PT) and along with public key pk this cipher text CT will be sent to the server.
3. Evaluation: Server has a function f for doing evaluation of cipher text CT and performed this as per the required function using pk.
4. Decryption: Generated Eval(f(PT)) will be decrypted by client using its sk and it gets the original result.

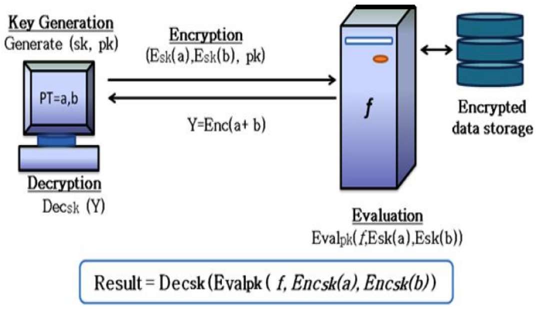

Figure 1: Homomorphic Encryption functions

2.3 Properties of Homomorphic Encryption

Homomorphic Encryption has mainly two properties,

Additive Homomorphic Encryption: A Homomorphic encryption is additive, if [12]:

$$E_k (PT1 \oplus PT2) = E_k (PT1) \oplus E_k (PT2)$$

As the encryption function is additively homomorphic, the following identities can be described:

The product of two cipher texts will decrypt to the sum of their corresponding plaintexts, D (E (m1, r1) · E (m2, r2) mod n2) = m1 + m2 mod n.

The product of a cipher text with a plaintext raising g will decrypt to the sum of the corresponding plaintexts, D (E (m1, r1) · gm2 mod n2) = m1 + m2 mod n. [13]

A remarkable feature of the Paillier cryptosystem is its homomorphic properties. In 1999, Pascal Paillier has introduced his cryptosystem [14]. The scheme is illustrated in the following.

Example: Paillier Cryptosystem (1999):

1. **Key generation:**
   Step 1: $n = pq$, the RSA modulus
   Step 2: $\lambda = \text{lcm}(p-1, q-1)$
   Step 3: $g \in Z /n^2 Z$ s.t. $n|\text{or } d_n^2(g)$
   Step 4: Public-key: (n, g), secret key: λ, μ
2. **Encryption of m :**
   Step 1: $m \in \{0, 1 \ldots n-1\}$, a message
   Step 2: $h \in_R Z/n Z$
   Step 3: $c = g^m h^n \bmod n^2$, a cipher text
3. **Decryption of c :**
   $m = L(c^\lambda \bmod n^2) L(g^\lambda \bmod n^2)^{-1} \bmod n$
   The constant parameter,
   $L(g^\lambda \bmod n^2)^{-1} \bmod n$ or $L(g^\alpha \bmod n^2)^{-1} \bmod n$ where $g=1+ n \bmod n^2$ can also be recomputed once for all.

Figure 2: Paillier Algorithm

Suppose there are two ciphers CT1 and CT2 such that:

$$CT1 = g^{m1} x_1^n \bmod n^2$$

$$CT2 = g^{m2} x_2^n \bmod n^2$$

$$CT1 \cdot CT2 = g^{m1} x_1^n . g^{m2} x_2^n \bmod n^2$$

Additive Property is: $g^{m1+m2}(x_1 x_2)^n \bmod n^2$

Pascal Paillier, the French mathematician, has proposed the new cryptographic algorithm named “Paillier Cryptosystem Algorithm” in 1999. It has an additive homomorphic property. Paillier cryptosystem is on the basis of “decisional composite residuosity assumption (DCRA)”. Therefore, the Paillier cryptosystem has various applications, for example, e-voting systems, threshold schemes, etc. [15].

Multiplicative Homomorphic Encryption: Homomorphic encryption is multiplicative, if [12]:

$$E_k (PT1 \otimes PT2) = E_k (PT1) \otimes E_k (PT2)$$

Rivest, Shamir and Adleman published their public key cryptosystem in 1978 [16].

Example 1: RSA Cryptosystem (1978):

1. Key Generation

Step 1: each user generates a public/private key pair by selecting, Two large primes at random - p, q
Step 2: computing their system modulus N= p.q and ø(N)=(p-1)(q-1)
Step 3: selecting at random the encryption key e Where, 1<e<ø(N), gcd(e,ø(N))=1
Step 4: publish their public encryption key: KU= {e,N} nkeep secret private decryption key: KR={d,p,q}

2. Encryption

Step 1: obtains public key of recipient KU={e,N}
Step 2: computes: $C=M^e$ mod N, where $0\leq M<N$

3. Decryption

Step 1: uses their private key KR={d,p,q}
Step 2: computes: $M=C^d$ mod N

Figure 3: RSA Algorithm

Following figure shows the homomorphic property of the RSA. Suppose there are two cipher texts, CT1 and CT2.

$$CT1 = m1^e \bmod n$$

$$CT2 = m2^e \bmod n$$

$$CT1 \cdot CT2 = m1^e \cdot m2^e \bmod n$$

So, multiplicative property: $(m1 \cdot m2)^e$ mod n

Example 2: ElGamal encryption algorithm [18]

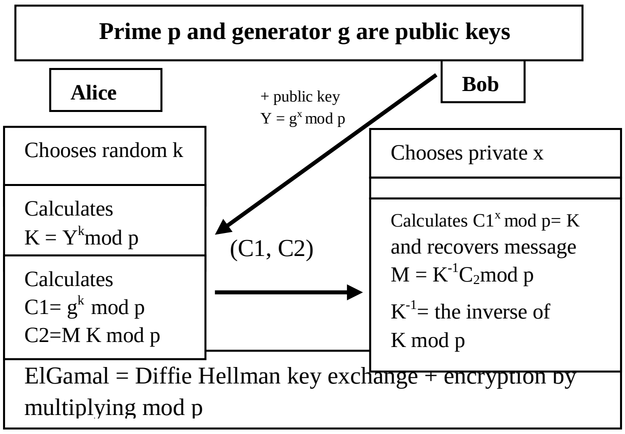

Figure 4: ElGamal Encryption-Decryption process

ElGamal encryption algorithm is proposed by the "Taher Elgamal" in 1984. ElGamal encryption algorithm is publickey algorithm and is multiplicative homomorphic [17]. ElGamal encryption-decryption process is shown in following Figure 4.

3. HOMOMORPHIC ENCRYPTION SCHEMES

In this section, the survey of various homomorphic encryption schemes like Algebra homomorphic encryption scheme based on updated ElGamal (AHEE), Non-interactive exponential homomorphic encryption algorithm (NEHE), homomorphic Cryptosystem (EHC), Brakerski-Gentry-Vaikuntanathan (BGV) etc is done.

3.1 BGV Encryption Scheme

Dealing with integer vectors (whose security is dependent on the hardness of decisional LWE (Learning with Errors) [19]) and dealing with the integer polynomials (whose security is dependent on the hardness of the decisional R-LWE (Ring LWE) [20]) are two versions of the cryptosystem. BGV is an asymmetric encryption scheme which can be used for the encryption of the bits.

Encrypt(Plaintext *m*, PublicKey *Pub*): Ciphertext *c*

Decrypt(Ciphertext *c*, PrivateKey *Priv*): Plaintext *m*

Level shifting operations

Rescale(Cipertext *c*): Ciphertext *c'*
SwitchKey(Augmented Cipertext *c*): Ciphertext *c'*

Homomorphic operations

Add(Ciphertext *c*1, Ciphertext *c*2): Ciphertext *csum*
Mul(Ciphertext *c*1, Ciphertext *c*2): Ciphertext *cmul*

Figure 5: Basic Encryption functions [21]

3.2 Gorti's Enhanced Homomorphic Cryptosystem (EHC)

EHC is the new Enhanced Homomorphic Cryptosystem used for homomorphic Encryption / Decryption with IND-CCA secure. There are numerous applications of this type of homomorphic encryption in the real time. Homomorphic encryption has the basic concept that the computer will perform the computations on the already encrypted data without having any knowledge of its real value. And at last this computed encrypted message or data will be sent back as a result and decrypted.

This decrypted result must be equal to the intended computed value if performed on the real data. For this reason, a particular structure has to be presented by the encryption scheme [22].

Chose large prime number ‘ p ‘ and another prime number ‘ q ‘
Calculate m = p * q
Generate a random number ‘ r ‘.
r,q and m Kept secret. Secret values r,q and m

Shared key : p

Encryption
Encrypt(X,m,p,q,r)
Assume X € *Zp*
Compute $Y = (X + r*pq) \pmod{m}$
Output Y € *Zc*

Decryption
Decrypt(Y,p)
input Y € *Zc*
compute X = Y mod p output X € *Zp*

Figure 6: Encryption/Decryption of EHC schemes [22]

3.3 Non-interactive Exponential Homomorphic Encryption Scheme [NEHE]

Non-interactive evaluation of encrypted exponential functions and polynomial functions can be implemented in following way [23].

Step 1) Sander and Tschudin [24][25] described the problem of non-interactive evaluation of encrypted functions (EEF):

- Alice (the originating host) has an algorithm to compute a function f.
- Bob (the remote host) has an input x and is willing to compute f(x).
- Alice wants Bob to learn nothing “substantial” about f.
- Bob should not need to interact with Alice during the computation of f(x).

Step 2) The protocol for non-interactive EEF proposed by

Sander and Tschudin[25] is shown in figure 7 as follows:

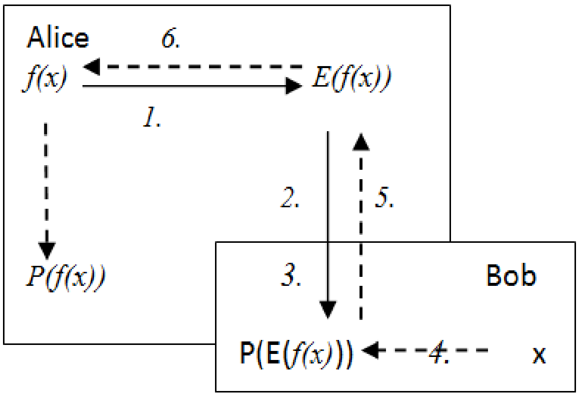

Figure 7: Encryption/Decryption of NEHE

1. Alice will encrypt f.
2. Alice will creates a program P(E(f)) which implements E(f).
3. Alice will send P(E(f)) to Bob.
4. Bob will execute P(E(f)) at x.
5. Bob will send P(E(f))(x) to Alice.
6. Alice will decrypt P(E(f))(x) and obtain f(x).

RSA, a public key algorithm is reviewed briefly in order to introduce and prove the exponential homomorphic encryption. Its security is based on the difficulty of factoring large integer [26].

3.4 Algebra Homomorphic Encryption Scheme Based On Updated ElGamal (AHEE)

This is the modified form of the digital signature standard DSS presented by the NIST in America[27]. Te security of the AHEE is IND-CPA which is the highest level of the security of AHEE. Additive homomorphism of this algorithm refers the same k for encryption but uses the random number of k in E1() which makes AHEE able to resist plaintext attack. The AHEE is the subset of the fully homomorphism. AHEE has been proved to be secure. This description of fully homomorphism is advanced by Rivest, Adleman and Dertouzos is as follows [28].

Sander and Tschudin defined additive and multiplicative homomorphisms on Integer Ring(Homomorphic Encryption Scheme, namely HES)[28-32] Homomorphic operations:

Step 1: select any two prime numbers say p and q

Step 2: calculate the product of those two prime numbers. Say N = p * q. where p and q being confidential and N is public.

Step 3: select random number x and a root g of GF(p). where g and x are smaller than p.

Step 4: calculate $y = g^x \bmod p$. use this y for the encryption.

Step 5: encryption will be performed in following two steps:

1. Select random integer number r and apply following homomorphic encryption.

   $E_1(M) = (M+r*p) \bmod N.$

2. Select random integer number k, and the encryption algorithms are:

   $E_g(M) = (a,b) = (g^k \bmod p,\ y^k E_1(M) \bmod p)$

Step 6: Decrypted algorithm D_g () is $M=b\times (a^x)^{-1}$ (mod p).

Figure 8: AHEE Homomorphic scheme

Multiplicative: Eg (M1M2) = Eg (M1)·Eg (M2), or

M1.M2=Dg (Eg (M1)·Eg (M2)).

Additive: Eg (M1+M2) = Eg (M1) ⊕Eg (M2), or

M1+M2=Dg (Eg (M1) ⊕Eg (M2)).

Above equations show multiplicative and additive Homomorphic Encryption properties of Algebra Homomorphic Encryption Scheme Based On Updated ElGamal (AHEE).

Comparison of various Homomorphic Encryption algorithms and schemes is described in table 1. In this table comparison is done based on additive, multiplicative and mixed Homomorphic properties of various algorithms and schemes.

Applications of different Homomorphic algorithms and schemes are also compared.

4. CONCLUSION

This paper presents the basic concept of the homomorphic encryption and the various encryption algorithms as per the properties of the homomorphic encryption; Paillier can be used for preserving the additive property of homomorphic encryption while ElGamal and RSA can be used for multiplicative property. This paper can be useful for those who are wishing to carry out research in the direction of the cryptographic algorithms used for homomorphic encryption. This survey can be helpful to know which and how various cryptographic algorithms are being used for applying homomorphic encryption for privacy preservation. There are various homomorphic encryption schemes described in this paper which can be used for mixed homomorphic encryption property. At last the comparison of all homomorphic encryption algorithms and schemes is done which may help to extend current research techniques.

5. ACKNOWLEDGMENTS

The authors are grateful to Sankita J. Patel who helped us to make clear the concept of homomorphic encryption. Authors also thank the anonymous reviewers for their constructive feedback.

Table 1. Comparison of various Homomorphic Encryption schemes

	Add-Homo	Multi-Homo	Mixed-Homo	Applications
Paillier	√	x	x	e-voting system, threshold scheme
RSA	x	√	x	To secure internet, Banking and credit card transaction
ElGamal	x	√	x	In Hybrid systems
BGV	x	x	√	For the security of integer polynomials.
EHC	x	x	√	Efficient Secure Message Transmission in Mobile Ad Hoc Networks

NEHE	x	x	√	Active networks, e-commerce based on mobile agent, computing grid
AHEE	x	x	√	Secure multi-party computation, electronic voting and mobile cipher

6. REFERENCES

[1] William, Stallings, and William Stallings. Cryptography and Network Security, 4/E. Pearson Education India, 2006.

[2] Lee, Hyungjick, Jim Alves-Foss, and Scott Harrison. "The use of encrypted functions for mobile agent security." In System Sciences, 2004. Proceedings of the 37th Annual Hawaii International Conference on, pp. 10-pp. IEEE, 2004.

[3] Coron, Jean-Sébastien, Tancrede Lepoint, and Mehdi Tibouchi. "Practical multilinear maps over the integers." Advances in Cryptology–CRYPTO 2013. Springer Berlin Heidelberg, 2013. 476-493

[4] Rashmi Nigoti,Manoj jhuria, Dr. shailendra singh " A survey of cryptographic algorithms for cloud computing", in Madhya Pradesh, India, IJETCAS 13-123,2013.

[5] Erfani, Shervin. "Security management system and method."U.S. Patent No. 6,542,993. 1 Apr. 2003.

[6] Diffie, Whitfield and Martin E. Hellman. "New directions in cryptography". Information Theory, IEEE Transactions on 22.6 (1976): 644-654.

[7] Zvika Brakerski and Vinod Vaikuntanathan, "Effcient Fully Homomorphic Encryption from (Standard) LWE", IeeeXplore-2011 BrakerskiV-FOCS 2011.

[8] Rivest, Ronald L., Len Adleman, and Michael L. Dertouzos. "On data banks and privacy homomorphisms." Foundations of secure computation 4, no. 11 (1978): 169-180.

[9] Dan Boneh, Eu-Jin Goh, and Kobbi Nissim. "Evaluating 2-DNF formulas on ciphertexts". In Theory of Cryptography Conference, TCC'2005, volume 3378 of Lecture Notes in Computer Science, pages 325-341. Springer, 2005.

[10] Boneh, Dan, Eu-Jin Goh, and Kobbi Nissim. "Evaluating 2-DNF formulas on ciphertexts." In Theory of cryptography, pp. 325-341. Springer Berlin Heidelberg, 2005.

[11] Yang, Jing, Mingyu Fan, Guangwei Wang, and Zhiyin Kong. "Simulation Study Based on Somewhat Homomorphic Encryption." Journal of Computer and Communications 2 (2014): 109.

[12] Tebaa, Maha, Saïd El Hajji, and Abdellatif El Ghazi. "Homomorphic encryption applied to the cloud computing security." In Proceedings of the World Congress on Engineering, vol. 1, pp. 4-6. 2012.

[13] Melchor, Carlos Aguilar, et al. "Improving Additive and Multiplicative Homomorphic Encryption Schemes Based on Worst-Case Hardness Assumptions}." IACR Cryptology ePrint Archive 2011 (2011): 607.

[14] Paillier, Pascal. "Public-key cryptosystems based on composite degree residuosity classes." In Advances in cryptology—EUROCRYPT'99, pp. 223-238. Springer Berlin Heidelberg, 1999.

[15] Sakurai, Kouichi, and Tsuyoshi Takagi. "On the security of a modified Paillier public-key primitive." Information Security and Privacy. Springer Berlin Heidelberg, 2002.

[16] El Gamal, Taher. "A public key cryptosystem and a signature scheme based on discrete logarithms." In Advances in Cryptology, pp. 10-18. Springer Berlin Heidelberg, 1985.

[17] Taher elgamal, member, "A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms", IEEE transactions on information theory, vol. it-31, no. 4, july 1985.

[18] Vaidehi, E. "Computing Aggregation Function Minimum/Maximum using Homomorphic Encryption Schemes in Wireless Sensor Networks (WSNs)." California State University, East Bay Hayward, CA, USA. (2007).

[19] Regev, Oded. "The learning with errors problem." In Blavatnik School of Computer Science, Tel Aviv University Invited survey in CCC (2010).

[20] Lyubashevsky, Vadim, Chris Peikert, and Oded Regev. "On ideal lattices and learning with errors over rings." Journal of the ACM (JACM) 60, no. 6 (2013): 43.

[21] Fau, Simon, et al. "Towards practical program execution over fully homomorphic encryption schemes." P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC), 2013 Eighth International Conference on. IEEE, 2013.

[22] Rao, Gorti VNKV Subba, and Garimella Uma. "An Efficient Secure Message Transmission in Mobile Ad Hoc Networks using Enhanced Homomorphic Encryption Scheme." GJCST-E: Network, Web & Security 13.9 (2013).

[23] Chen, Liang, Zhang Tong, Wen Liu, and Chengmin Gao. "Non-interactive Exponential Homomorphic Encryption Algorithm." In Cyber-Enabled Distributed Computing

and Knowledge Discovery (CyberC), 2012 International Conference on, pp. 224-227. IEEE, 2012..

[24] Sander, Tomas, and Christian F. Tschudin. "Towards mobile cryptography." In Security and Privacy, 1998. Proceedings. 1998 IEEE Symposium on, pp. 215-224. IEEE, 1998.

[25] Sander, Tomas, and Christian F. Tschudin. "Protecting mobile agents against malicious hosts." In Mobile agents and security, pp. 44-60. Springer Berlin Heidelberg, 1998.

[26] Chen, Liang, Zhang Tong, Wen Liu, and Chengmin Gao. "Non-interactive Exponential Homomorphic Encryption Algorithm." In Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), 2012 International Conference on, pp. 224-227. IEEE, 2012.

[27] Smid, Miles E., and Dennis K. Branstad. "Response to comments on the NIST proposed Digital Signature Standard." Advances in Cryptology—Crypto'92. Springer Berlin Heidelberg, 1993.

[28] Xiang, Guangli, Benzhi Yu, and Ping Zhu. "A algorithm of fully homomorphic encryption." In Fuzzy Systems and Knowledge Discovery (FSKD), 2012 9th International Conference on, pp. 2030-2033. IEEE, 2012.

[29] Sander, Tomas, and Christian F. Tschudin. "Towards mobile cryptography." In Security and Privacy, 1998. Proceedings. 1998 IEEE Symposium on, pp. 215-224. IEEE, 1998.

[30] Xiang, Guangli, Benzhi Yu, and Ping Zhu. "A algorithm of fully homomorphic encryption." In Fuzzy Systems and Knowledge Discovery (FSKD), 2012 9th International Conference on, pp. 2030-2033. IEEE, 2012.

[31] Zhu, Ping, Yanxiang He, and Guangli Xiang. "Homomorphic Encryption Scheme of the Rational." In Wireless Communications, Networking and Mobile Computing, 2006. WiCOM 2006. International Conference on, pp. 1-4. IEEE, 2006.

[32] Chen, Liang, and Chengmin Gao. "Public Key Homomorphism Based on Modified ElGamal in Real Domain." In 2008 International Conference on Computer Science and Software Engineering, vol. 3, pp. 802-805. 2008.