RESEARCH ARTICLE OPEN ACCESS

State Of Art in Homomorphic Encryption Schemes

S. Sobitha Ahila[1], Dr.K.L.Shunmuganathan[2]
Assistant Professor,CSE Easwari engineering college
Professor, CSE R.M.K Engineering College

Abstract
The demand for privacy of digital data and of algorithms for handling more complex structures have increased exponentially over the last decade. However, the critical problem arises when there is a requirement for publicly computing with private data or to modify functions or algorithms in such a way that they are still executable while their privacy is ensured. This is where homomorphic cryptosystems can be used since these systems enable computations with encrypted data. A fully homomorphic encryption scheme enables computation of arbitrary functions on encrypted data.. This enables a customer to generate a program that can be executed by a third party, without revealing the underlying algorithm or the processed data. We will take the reader through a journey of these developments and provide a glimpse of the exciting research directions that lie ahead. In this paper, we propose a selection of the most important available solutions, discussing their properties and limitations.

Keywords— mobile agents, homomorphic encryption

I. INTRODUCTION

The goal of encryption is to ensure confidentiality of data in communication and storage processes. Recently, its use in constrained devices led to consider additional features such as the ability to delegate computations to un trusted computers. For this purpose, we would like to give the un trusted computer only an encrypted version of the data to process. The computer will perform the computation on this encrypted data, hence without knowing anything on its real value. Finally, it will send back the result, and user will decrypt it. For coherence, the decrypted result has to be equal to the intended computed value if performed on the original data. For this reason, the encryption scheme has to present a particular structure. Rivest et al. proposed in 1978 to solve this issuethrough homomorphic encryption Unfortunately, Brickell and Yacobi pointed out in some security flaws in the first proposals of Rivest et al . Since this first attempt,a lot of articles have proposed solutions dedicated to numerous application contexts: anonymity, privacy, electronic voting, electronic auctions, lottery protocols , protection of mobile agents , multiparty computation and so forth. The goal of this article is to provide a survey of partial and full homomorphic encryption techniques

In Section 2, we provide some basic and fundamental information on cryptography and various types of encryption schemes. In Section 3, we discuss some of basic definitions about homomorphic encryption schemes in the literature. Section 4 provides a brief presentation of applications of homomorphic cryptosystems. Section 5 presents a discussion on partial homomorphic encryption schemes. Section 6 presents a discussion on fully homomorphic encryption schemes which are the most powerful encryption schemes for providing a framework for computing over encrypted data. Finally, Section 7 concludes the chapter while outlining a number of research directions and emerging trends in this exciting filed of computation which has a tremendous potential of finding applications in the real-world deployments.

II. TOWARDS HOMOMORPHIC ENCRYPTION

A. Conventional Cryptography

In this Section, we will recall some important concepts on encryption schemes. Encryption schemes are designed to preserve confidentiality. The security of encryption schemes must not rely on the obfuscation of their codes, but it should only be based on the secrecy of the key used in the encryption process. Encryption schemes are broadly of two types

- symmetric encryption schemes
- asymmetric encryption schemes

In the following, we present a very brief discussion on each of these schemes.

1) Symmetric encryption schemes

In these schemes, the sender and the receiver agree on the key they will use before establishing any secure communication session. Therefore, it is not possible for two persons who never met before to use such schemes directly. This also implies that in order

to communicate with different persons, we must have a different key for each people. Requirement of large number of keys in these schemes make their key generation and management relatively more complex operations. However, symmetric schemes present the advantage of being very fast and they are used in applications where speed of execution is a paramount requirement. Symmetric-key encryption can use either stream ciphers or block ciphers.

- Stream ciphers encrypt the digits (typically bytes) of a message one at a time.
- Block ciphers take a number of bits and encrypt them as a single unit, padding the plaintext so that it is a multiple of the block size. Blocks of 64 bits have been commonly used. The Advanced Encryption Standard (AES) algorithm approved by NIST in December 2001 uses 128-bit blocks.

Examples of popular and well-respected symmetric algorithms Include Twofish, Serpent, AES (Rijndael), Blowfish, CAST5, RC4, 3DES, and IDEA

2) Asymmetric encryption schemes:

In these schemes, every participant has a pair of keys private and public. While the private key of a person is known to only her, the public key of each participant is known to everyone in the group. Such schemes are more secure than their symmetric counterparts and they don't need any prior agreement between the communicating parties on a common key before establishing a session of communication.RSA ,ElGamal, Diffie–Hellman key exchange protocol, DSS (Digital Signature Standard), which incorporates the Digital Signature Algorithm, Paillier cryptosystem, Cramer–Shoup cryptosystem and YAK authenticated key agreement protocol

B. Security of encryption schemes

Security of encryption schemes was first formalized by Shannon with the notion of perfect secrecy/unconditional secrecy, which characterizes encryption schemes for which the knowledge of a ciphertext does not give any information about the corresponding plaintext and the encryption key. One-Time Pad encryption scheme is perfectly secure under certain conditions. However, no other encryption scheme has been proved to be unconditionally secure. For asymmetric schemes, we can rely on their mathematical structures to estimate their security strength in a formal way. These schemes are based on some well-identified mathematical problems which are hard to solve in general, but easy to solve for the one who knows the trapdoor – i.e., the owner of the keys. However, the estimation of the security level of these schemes may not be always correct due to several reasons. First, there may be other ways to break the system than solving the mathematical problems on which these schemes are based. Second, most of the security proofs are performed in an idealized model called random oracle model, in which involved primitives, for example, hash functions, are considered truly random. This model has allowed the study of the security level of numerous asymmetric ciphers. However, we are now able to perform proofs in a more realistic model called standard model (Canetti et al., 1998; Paillier, 2007). This model eliminates some of the unrealistic assumptions in the random oracle model and makes the
security analysis of cryptographic schemes more practical.
Usually, to evaluate the attack capacity of an adversary, we distinguish among several contexts

- cipher-text only attacks (where the adversary has access only to some ciphertexts)
- known-plaintext attacks (where the adversary has access to some pairs of plaintext messages and their corresponding ciphertexts)
- chosen-plaintext attacks (the adversary has access to a decryption oracle that behaves like a black-box and takes a ciphertext as its input and outputs the corresponding plaintexts).

C. *Probabilistic encryption:*

Almost all the well-known cryptosystems are deterministic. This means that for a fixed encryption key, a given plaintext will always be encrypted into the same ciphertext under these systems. However, this may lead to some security problems.RSA scheme is a good example for explaining this point. Let us consider the following points with reference to the RSA cryptosystem:

- A particular plaintext may be encrypted in a too much structured way. With RSA,messages 0 and 1 are always encrypted as 0 and 1, respectively.
- It may be easy to compute some partial information about the plaintext: with RSA,the cipher text c leaks one bit of information about the plaintext m, namely, the so called Jacobi symbol .
- When using a deterministic encryption scheme, it is easy to detect when the same message is sent twice while processed with the same key.

In view of the problems stated above, we prefer encryption schemes to be probabilistic. In case of symmetric schemes, we introduce a random vector in the encryption process (e.g., in the pseudo-random generator for stream ciphers, or in the operating mode for block ciphers) – generally called initial vector (IV). This vector may be public and it may be transmitted in a clear-text form. However, the IV must be changed every time we encrypt a message. In case of asymmetric ciphers, the security analysis is

more mathematical and formal, and we want the randomized schemes to remain analyzable in the same way as the deterministic schemes. Researchers have proposed some models to randomize the existing deterministic schemes, as the optimal asymmetric encryption padding (OAEP) for RSA (or any scheme that is based on a trapdoor one-way permutation) [2] A simple consequence of this requirement of the encryption schemes to be preferably probabilistic appears in the phenomenon called expansion. Since for a plaintext we require the existence of several possible ciphertexts, the number of ciphertexts is greater than the number of possible plaintexts. This means the ciphertexts cannot be as short as the plaintexts; they have to be strictly longer. The ratio of the length of the ciphertext and the corresponding plaintext (in bits) is called expansion. The value of this parameter is of paramount importance in determining security and efficiency tradeoff of a probabilistic encryption scheme. In Paillier's scheme, an efficient probabilistic encryption mechanism has been proposed with the value of expansion less than 2 .

III. HOMOMORPHIC ENCRYPTION SCHEMES

A. Homomorphism:

A function $f : G \rightarrow H$ from one group G to another H is a (group) homomorphism if the group operation is preserved in the sense that

$$f(g1 *_G g2) = f(g1) *_H f(g2)$$

for all g1, g2 ∈ G. Let e_G be the identity in G and e_H the identity in H . A group homomorphism f maps e_G to e_H : $f(e_G) = f(e_H)$ .Note that f must preserve the inverse map due to:

$f(g)f(g^{-1}) = f(gg^{-1}) = f(e_G)$,

therefore: $f(g)^{-1} = f(g^{-1})$.

The kernel of a homomorphism f is

$\ker f = \{g \in G : f(g) = e_h\}$

The image of f is like the image of any function

$\text{im } f = \{h \in H : \exists\ g \in G \text{ such that } f(g) = h\}$

If a group homomorphism f : G → H is subjective, then H is said to be a homomorphic image of G.

B. Homomorphic encryption

Homomorphic encryption is a form of encryption which allows specific types of computations to be carried out on ciphertext and obtain an encrypted result which decrypted matches the result of operations performed on the plaintext.

A public-key encryption scheme E = (KeyGen, Enc, Dec) is homomorphic if for all k and all (pk, sk) output from KeyGen(k), it is possible to define groups M, C so that:The plaintext space M, and all ciphertexts output by Enc_{pk} are elements of C. For any m1, m2 ∈ M and c1, c2 ∈ C with $m1 = Dec_{sk}(c1)$ and $m2 = Dec_{sk}(c2)$ it holds that

$$Dec_{sk}(c1 * c2) = m1 * m2$$

where the group operations * are carried out in C and M, respectively. In other words, a homomorphic cryptosystem is a PKS with the additional property that there exists an efficient algorithm (Eval) to compute an encryption of the sum or/and the product, of two messages given the public key and the encryptions of the messages, but not the messages themselves.

C. Fully homomorphic encryption

This scheme is able to output a ciphertext that encrypts $f(m_1,..., m_t)$, where f is any desired function, which of course must be efficiently computable. No information about $m_1, ..., m_t$ or $f(m_1, ..., m_t)$, or any inter- mediate plaintext values should leak. The inputs, outputs and intermediate values are always encrypted and therefore useless for an adversary.

A public key encryption scheme (KeyGen, Enc, Dec) is fully homomorphic if there exists an additional efficient algorithm Eval that, for a valid public key pk, a permitted circuit C and a set of ciphertexts Ψ = {c1, ..., ct} where

ci ← Enc_{pk} (mi), outputs c ← $Eval_{pk}$ (C, Ψ) under pk.

IV. APPLICATIONS OF HOMOMORPHIC ENCRYPTION SCHEMES

A. Protection of mobile agents

The protection of mobile agents by homomorphic encryption can be used in two ways:

(i) computing with encrypted functions
(ii) computing with encrypted data.

Computation with encrypted functions is a special case of protection of mobile agents. In such scenarios, a secret function is publicly evaluated in such a way that the function remains secret. Using homomorphic cryptosystems the encrypted function can be evaluated which guarantees its privacy. Homomorphic schemes also work on encrypted data to compute publicly while maintaining the privacy of the secret data. This can be done encrypting the data in advance and then exploiting the homomorphic property to compute with encrypted data.

B. Multiparty computation

In multiparty computation schemes, several parties are interested in computing a common, public function on their inputs while keeping their individual inputs private. This problem belongs to the area of computing with encrypted data.

C. Election schemes:

In election schemes, the homomorphic property provides a tool to obtain the tally given the encrypted votes without decrypting the individual votes.

D. Mix-nets

Mix-nets are protocols that provide anonymity for senders by collecting encrypted messages from several users. A desirable property to build such mix-nets is re- encryption which is achieved by using homomorphic encryption.

E. Data aggregation in wireless sensor networks

Homomorphic encryption schemes can be applied to protect privacy of input data while computing an arbitrary aggregation function in a wireless sensor network

V. PARTIAL HOMOMORPHIC ENCRYPTION SCHEMES

In this Section, we describe some homomorphic encryption systems which have created substantial interest among the researchers in the domain of cryptography.

A. Goldwasser-Micali scheme

In Goldwasser-Micali scheme as for RSA, we use computations modulo n = pq, a product of two large primes. Encryption is simple, with a product and a square, whereas decryption is heavier, with an exponentiation. Nevertheless, this step can be done in $O(l(p)^2)$[10]. The basic principle of GM is to partition a well-chosen subset of integers modulo n into two secret parts: M_0 and M_1.Then, encryption selects a random element of M_b to encrypt b, and decryption allows to know in which part the randomly selected element lies. The core point lies in the way to choose the subset, and to partition it intoM_0 andM_1. GM uses group theory to achieve the following: the subset is the group G of invertible integers modulo n with a Jacobi symbol, with respect to n, equal to 1. The partition is generated by another group $H \subset G$, composed of the elements that are invertible modulo n with a Jacobi symbol, with respect to a fixed factor of n, equal to 1; with these settings, it is possible to split G into two parts: H and $G \setminus H$ The generalizations of Goldwasser-Micali play with these two groups; they try to fit two groups G and H such that G can be split into more than k = 2 parts.Some limitations are encrypting k bits leads to a cost of $O(k \cdot l(p)^2)$. This is not very efficient even if it is considered as practical. Concerns about the expansion: a single bit of plaintext is encrypted in an integer modulo n, that is, l(n) bits. Thus, the expansion is really huge.

B. Benaloh's scheme

Benaloh is a generalization of GM, that enables to manage inputs of l (k) bits, k being a prime satisfying some particular constraints. Encryption is similar as in the previous scheme (encrypting a message $m \in \{0, \ldots, k-1\}$ means picking an integer $r \in \mathbf{Z}^*$ and computing $c = g^m r^k \bmod n$ but decryption is more complex. The input and out- put sizes being, respectively, of l(k) and l(n) bits, the expansion is equal to l(n)/l(k)[11]. This is better than in the GM case. Moreover, the encryption is not too expensive as well. The overhead in the decryption process is estimated to be $O(\sqrt{k}.l(k))$ for pre-computation which remains constant for each dynamic decryption step. But Value of k has to be taken very small, which in turn limits the gain obtained on the value of expansion.

C. Naccache-Stern scheme

This scheme is an improvement of Benaloh's scheme. Using a value of the parameter k that is greater than that used in the Benaloh's scheme, it achieves a smaller expansion and thereby attains a superior efficiency[8].The encryption step is precisely the same as in Benaloh's scheme. However, decryption is different. The value of expansion is same as that in Benaloh's scheme, i.e.
l (n)/ l (k).However, the cost of ecryption is less and is given by: $O(l(n)^5 \log(l(n))$ The authors claim that it is possible to choose the values of the parameters in the system in such a way that the achieved value of expansion is 4.

D. Okamoto-Uchiyama scheme

Considering n = p2 q, p and q still being two large primes, and the group $G = \mathbf{Z}_{p^*}{}^2$,they achieve k = p. Thus, the expansion is equal to 3[12].

E. Paillier scheme

One of the most well-known homomorphic encryption schemes is due to Paillier[13]. It is an improvement of the previous one, that decreases the expansion from 3 to 2. Paillier came back to n = pq, with $\gcd(n, \varphi(n)) = 1$, but considered the group $G = \mathbf{Z}^* \mathrm{P}^2$, and a proper choice of H led him to k = e (n). The encryption cost is not too high. Decryption needs one exponentiation modulo n2 to the power λ (n), and a multiplication modulo n. Paillier showed in his paper how to manage decryption efficiently through the Chinese Remainder Theorem. With smaller expansion and lower cost compared with the previous ones, this scheme is really attractive.In 2002, Cramer and Shoup proposed a general approach to gain security against adaptive chosen-ciphertext attacks for certain cryptosystems with some particular algebraic properties . Applying it to

Paillier's original scheme, they proposed a stronger variant. Bresson et al. proposed in a slightly different version that may be more accurate for some applications.

F. Damgard-Jurik scheme

Damgard and Jurik propose a generalization of Paillier's scheme to groups of the form Z*n s+l for s > 0. In this scheme, choice of larger values of s will achieve lower values of expansion [14]. This scheme can be used in a number of applications. For example, we can mention the adaptation of the size of the plaintext, the use of threshold cryptography, electronic voting, and so on. To encrypt a message, m $\in Z_n$ one picks $r \in \mathbf{Z}^*$ n at random and computes $g^m r^{n^s} \in \mathbf{Z}_n s+1$. The authors show that if one can break the scheme for a given value s = σ, then one can break it for s = σ − 1. They also show that the semantic security of this scheme is equivalent to that of Paillier. To summarize, the expansion is of 1 + 1/s, and hence can be close to 1 if s is sufficiently large. The ratio of the encryption cost of this scheme over Paillier's can be estimated to be (1/6)s(s + 1)(s + 2). The same ratio for the decryption step equals (1/6)(s + 1)(s + 2).Note that even if this scheme is better than Paillier's according to its lower expansion, it remains more costly. Moreover, if we want to encrypt or decrypt k blocks of l (n) bits, running Paillier's scheme k times is less costly than running Damgard-Jurik's scheme once.

G. Galbraith scheme

This is an adaptation ρf the existing homomorphic encryption schemes in the context of elliptic curves [15]. Its expansion is equal to 3. For s = l the ratio of the encryption cost for this scheme over that of Paillier's scheme can be estimated to be about 7, while the same ratio for the cost of decryption cost is about 14 for the same value of s. However, the most important advantage of this scheme is that the cost of encryption and decryption can be decreased using larger values of s. In addition, the security of the scheme increases with the increase in the value of s as it is the case in Damgard-Jurik's

H. Castagnos scheme

Castagnos explored the possibility of improving the performance of homomorphic encryption schemes using quadratic fields quotations [16]. This scheme achieves an expansion value of 3 and the ratio of encryption/decryption cost with s = l over Paillier's scheme can be estimated to be about 2.

VI. FULLY HOMOMORPHIC ENCRYPTION SCHEMES

A. Gentry's scheme

He proposed fully homomorphic encryption consists of several steps: It constructs a somewhat homomorphic scheme that supports evaluating low-degree polynomials on the encrypted data. It squashes the decryption procedure so that it can be expressed as a low-degree polynomial which is supported by the scheme It applies a bootstrapping transformation to obtain a fully homomorphic scheme[6].

B. *Brakerski and Vaikuntanathan, Scheme*

They have constructed a somewhat homomorphic encryption scheme based on RLWE[1,3,4]. The scheme inherits the simplicity and efficiency, as well as the worst case relation to ideal lattices. Moreover, the scheme enjoys key dependent message security (KDM security, also known as circular security), since it can securely encrypt polynomial functions (over an appropriately defined ring) of its own secret key. The authors argue that all known constructions of fully homomorphic encryption employ a bootstrapping technique that enforces the public key of the scheme to grow linearly with the maximal depth of evaluated circuits. This is a major drawback with regard to the usability and the efficiency of the schemes. However, the size of the public key can be made independent of the circuit depth if the somewhat homomorphic scheme can securely encrypt its own secret key[9]. With the design of this scheme, the authors have solved an open problem - achieving circular secure somewhat homomorphic encryption. The authors have also shown how to transform the proposed scheme into a fully homomorphic encryption scheme following Gentry's blueprint of squashing and bootstrapping.

C. *Smart and Vercauteren scheme*

They present a fully homomorphic encryption scheme has smaller key and ciphertext sizes [17]. The construction proposed by the authors follows the fully homomorphic construction based on ideal lattices proposed by Gentry. It produces a fully homomorphic scheme form a somewhat homomorphic scheme. For somewhat homomorphic scheme, the public and the private keys consist of two large integers (one of which shared by both the public and the private key), and the ciphertext consists of one large integer.

D. Gentry and Halev scheme

They presented a novel implementation approach for the variant of Smart and Vercauteren proposition which had a greatly improved key generation phase. In particular, the authors have noted that key generation (for cyclotomic fields) is

essentially an application of a Discrete Fourier Transform (DFT), followed by a small quantum of computation, and then application of the inverse transform. The key generation method of Gentry and Halevi is fast [7].

E. Stehle and Steinfield scheme

They improved Gentry's fully homomorphic scheme and obtained a faster fully homomorphic scheme with O(n3.5) bits complexity per elementary binary addition/multiplication gate However, the hardness assumption of the security of the scheme is stronger than that of Gentry's scheme. The improved complexity of the proposed scheme stems from two sources[18]. First, the authors have given a more aggressive security analysis of the sparse subset sum problem (SSSP) against lattice attacks as compared to the analysis presented in (Gentry, 2009). The SSSP along with the ideal lattice bounded distance decoding (BDD) problem are the two problems underlying the security of Gentry's fully homomorphic scheme. On the contrary, the finer analysis of Stehle and Steinfield for SSSP takes into account the complexity of approximate SVP, thereby making it more consistent with the assumption underlying the analysis of the BDD problem.

F. Chunsheng scheme

Chunsheng proposed a modification of the fully homomorphic encryption scheme of Smart and Vercauteren. The author has applied a self-loop bootstrappable technique[19] so that the security of the modified scheme only depends on the hardness of the polynomial coset problem and does not require any assumption of the sparse subset problem.In addition, the author have constructed a non- self-loop fully homomorphic encryption scheme that uses cycle keys. In a nutshell, the security of the improved fully homomorphic encryption scheme in this work is based on use of three mathematical approaches: (i) hardness of factoring integer problem, (ii) solving Diophantine equation problem, and (iii) finding approximate greatest common divisor problem.

G. Boneh & Freeman scheme

Boneh and Freeman propose a linearly homomorphic signature scheme that authenticates vector subspaces of a given ambient space [20]. The scheme has several novel features that were not present in any of the existing similar schemes. First, the scheme is the first of its kind that enables authentication of vectors over binary fields; previous schemes could not authenticate vectors with large or growing coefficients. Second, the scheme is the only scheme that is based on the problem of finding short vectors in integer lattices, and therefore, it enjoys the worst-case security guarantee that is common to lattice-based cryptosystems. The scheme can be used to authenticate linear transformations of signed data, such as those arising when computing mean and Fourier transform or in networks that use network coding.

VII. CONCLUSIONS

We presented in this paper a state of the art on homomorphic encryption schemes discussing their parameters, performances and security issues. As we saw, these schemes are not well suited for every use, and their characteristics must be taken into account. Nowadays, such schemes are studied in wide application contexts, but the research is still challenging in the cryptographic community to design more powerful secure schemes. Performing computations using fully homomorphic encryption scheme nowadays takes quite a long time, but as techniques evolve things will quickly change. Researchers believe in the possibility of advancing in fully homomorphic encryption area and bringing new related technologies to the wide market. It can be used whenever the need of doing computations on pieces of un-owned information appears.We therefore conclude that focusing on these topics would be a good idea for further research.

REFERENCES

[1] Brakerski, Z., Gentry, C., & Vaikuntanathan, (2011). Fully Homomorphic Encryption without Bootstrapping. In: Proceedings of the 3rd Innovations in Theoretical Computer Science Conference (ITCS'12), pp. 309-325, ACM Press, New York, NY, USA.

[2] R. Rivest, L. Adleman, and M. Dertouzos, "On data banks and privacy homomorphisms," in Foundations of Secure Computa- tion, pp. 169–177, Academic Press, 1978.

[3] Brakerski, Z. & Vaikuntanathan, V. (2011). Fully Homomorphic Encryption from Ring-LWE and Security for Key Dependent Messages. In: Advances in Cryptology-Proceedings of CRYPTO'11, Lecture Notes in Computer Science (LNCS), Vol 6841, Springer-Verlag, pp. 505-524.

[4] Brakerski, Z. & Vaikuntanathan, V. (2011a). Efficient Fully Homomorphic Encryption from (Standard) LWE. In: Proceedings of the IEEE 52nd Annual Symposium on Foundations of Computer Science (FOCS'11), pp. 97-106, ACM Press, New York, NY, USA.

[5] ElGamal, T. (1985). A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms. IEEE

Transactions on Information Theory, Vol 31, Issue 4, July 1985, pp. 469-472

[6] Gentry, C. (2009). Fully Homomorphic Encryption Using Ideal Lattices. In: Proceedings of the 41st Annual ACM Symposium on Theory of Computing (STOC'09), pp. 169-178, ACM Press, New York, NY, USA.

[7] Gentry, C. & Halevi, S. (2011). Implementing Gentry's Fully-Homomorphic Encryption Scheme. In: Advances in Cryptology - Proceedings of EUROCRYPT'11, Lecture Note in Computer Science (LNCS), Vol 6632, Springer-Verlag, pp. 129-148.

[8] Naccache, D. & Stern, J.(1998). A New Public Key Cryptosystem Based on Higher Residues.In: Proceedings of the 5th ACM Conference on Computer and Communications Security(CCS'98), pp. 59-66, ACM Press, New York, NY, USA.

[9] Vaikuntanathan, V. (2011). Computing Blindfolded: New Developments in Fully Homomorphic Encryption. In: Proceedings of the IEEE 52nd Annual Symposium on Foundations of Computer Science (FOCS'11), pp. 5-16, IEEE Computer Society Press, Washington, DC, USA.

[10] Goldwasser, S. & Micali, S. (1984). Probabilistic Encryption. Journal of Computer and System Sciences, Vol 28, Issue 2, pp. 270-299, April 1984.

[11] J. Benaloh, Verifiable secret-ballot elections, Ph.D. thesis, Yale University, Department of Computer Science, New Haven, Conn, USA, 1988

[12] Okamoto, T. & Uchiyama, S. (1998). A New Public-Key Cryptosystem as Secure as Factoring. In: Advances in Cryptology-Proceedings of EUROCRYPT'98, Lecture Notes in Computer Science (LNCS), Vol 1403, Springer-Verlag, pp. 308-318

[13] P. Paillier, "Public-key cryptosystems based on composite de- gree residuosity classes," in Advances in Cryptology (EURO- CRYPT '99), vol. 1592 of Lecture Notes in Computer Science, pp. 223–238, Springer, New York, NY, USA, 1999.

[14] Damgard, I. & Jurik, M. (2001). A Generalisation, a Simplification and Some Applications of Paillier's Probabilistic Public-Key System. In: Proceedings of the 4th International Workshop on Practice and Theory in Public Key Cryptography (PKC'01), Lecture Notes in Computer Science (LNCS), Vol 1992, Springer-Verlag, pp. 119-136.

[15] Galbraith, S. D. (2002). Elliptic Curve Paillier Schemes. Journal of Cryptology, Vol 15, No 2, pp.129-138, August 2002.

[16] Castagnos, G. (2007). An Efficient Probabilistic Public-Key Cryptosystem over Quadratic Fields Quotients. Finite Fields and Their Applications, Vol 13, No 3, pp. 563-576, July 2007.

[17] Smart, N. & Vercauteren. (2012). Fully Homomorphic SIMD Operations. Design Codes and Cryptography, Springer, USA, July 2012.

[18] Stehle, D. & Steinfeld, R. (2010). Faster Fully Homomorphic Encryption. In: Advances in Cryptology – Proceedings of ASIACRYPT'10, Lecture Notes in Computer Science (LNCS), Vol 6477, Springer-Verlag, pp. 377-394.

[19] Gu Chun-sheng (2012)Attack on Fully Homomorphic Encryption over the Integers-International Journal of Information & Network Security (IJINS) Vol.1, No.4, October 2012, pp. 275~281

[20] Boneh, D. & Freeman, D. M. (2011). Linearly Homomorphic Signatures over Binary Fields and New Tools for Lattice-Based Signatures. In: Public Key Cryptography (PKC'11), Lecture Notes in Computer Science (LNCS), Vol 6571, Springer-Verlag, pp. 1-16.