Journal of Current Research in Engineering and Science
JCRES
Bi-Annual Online Journal (ISSN : 2581 - 611X)

Volume 7- Issue 1, January 2024
Paper : 98

A Comprehensive Study on Classical Homomorphic Encryption Algorithms for Data Encryption

Sonam Mittal[1] and Soni Singh[2]
Chitkara University Institute of Engineering and Technology,
Chitkara University,
Rajpura, Punjab, India
Sonam.mittal@chitkara.edu.in, and Soni.singh@chitkara.edu.in

***Abstract*— Different homomorphic encryption schemes are emerging as the new boom in the field of cryptography. It enables the user to perform computation on the encrypted text itself. The evolution of the homomorphic encryption scheme has not happened in one night, it took years of hard work by researchers. The paper studies various classical homomorphic encryption algorithms that served as a milestone in the journey of achieving a fully homomorphic encryption scheme. The chronological survey of classical homomorphic encryption is presented. A comparative analysis is done to analyze the homomorphic properties and ciphertext expansion ratio. A clear understanding of various classical homomorphic encryption with their algorithms is done and makes it easy for new researchers to understand the evolution of homomorphic encryption from the very first encryption algorithm, the RSA algorithm. It may help the latest researchers in the field of cryptography, to understand homomorphic encryption in-depth and its evolution.**

Keywords- ***Homomorphic Encryption, Data Encryption, RSA, GM, Cryptography Security, .***

I. Introduction

Cryptography is used as a technique for the secure transmission of data between two different parties, where the first party as a sender can encrypt the message and send it to the second party. On another end, the second party as a receiver receives the encrypts message and decrypts it to retrieve the original message [1]. In past few decades, a lot of research has been done in the field of cryptography for secure data transmission, storage, and computation. Researchers are successfully able to design various encryption algorithms for secure data transmission and storage. But performing secure computation over the data is still in the developing stage.

The idea of querying over the encrypted text in a database gives rise to the concept of homomorphism. The first encryption algorithm, RSA in 1978 is secure while data is in the resting stage or transition stage. But when data is in use, then Homomorphic Encryption (HE) comes into the scene to secure the data by performing a set of computations over encrypted data and maintains the data privacy (hiding the actual content from the data user).

II. Classical Homomorphic Encryption Models

This section discussed some classical HE schemes, which have created a new interest area among researchers in the domain of cryptography. The first encryption scheme RSA in 1978 and then showed gradual improvements as other encryption standards come into existence. Following chronological order, this section summarizes all algorithms in detail, considering their important parameters and properties.

A. Rivest-Shamir-Adleman Encryption Scheme

RSA is an asymmetric Public Key Encryption (PKE) algorithm, presented by Ron Rivest, Adi Shamir, and Leonard Adleman in 1978. The RSA's security rely on the problem of factoring large integers [2]. The large size of the keys makes it more secure but at the same time very slow on computation. The parameters used are modulus n, encryption exponent e, and decryption exponent d are used in different sub-algorithms as given below:

1) *Key Generation (KeyGen):*

a) Select two large prime numbers, p *and* q, and compute $n = p * q$ *and* $\phi(n) = (p-1)(q-1)$.

b) Choose an integer e, $(1 < e < \phi)$, such that $\gcd(e, \phi) = 1$.

c) Calculate the secret exponent, $d\,(1 < d < \phi)$, such that $d \equiv e^{-1}\, 1\, (mod\ \phi)$

d) The public and private key is (n, e) and (n, d).

2) Encryption (Encrypt):

a) Get a public key (n, e).

b) Generate the cipher text c from plain text m using $c = m^e mod\ n$

3) Decryption (Decrypt)

a) Use private key (n, d) to get plain text m using $m = c^d mod\ n$.

RSA algorithm shows the multiplicative homomorphic property as shown below. Suppose there are two cipher texts, CT_1 *and* CT_2 for plaintexts m_1 *and* m_2 respectively, such as $CT_1 = m_1^e\ mod\ n$ and $CT_2 = m_2^e\ mod\ n$ therefore, $CT_1 \times CT_2 = m_1^e\ mod\ n \times m_2^e\ mod\ n$.

Journal of Current Research in Engineering and Science
JCRES
Bi-Annual Online Journal (ISSN : 2581 - 611X)

So, multiplicative property states,$CT_1 \times CT_2 = (m_1 \times m_2)^e \; mod \; n$.

B. *Goldwasser-Micali scheme (GM Scheme)*

GM scheme [3], encryption process is very as it uses only product and square mathematical operations, whereas decryption is more complex due to exponential operation and having complexity $O\left(k.l(p^2)\right)$, where, $l(p)$ presents the number of bits in p. Single bit input, and the ciphertext expansion are two major drawback of this scheme. As, a single bit of plaintext is encrypted in an integer modulo n, that is, $l(n)$ bits, i.e., encryption of 1 bit, generates 1024 bits long ciphertext [4]. The GM scheme is as follows:

1) *Key Generation (KeyGen):*
   a) Select $p \; and \; q$, must be large prime numbers
   b) Calculate $N = p.q$.
   c) Choose an integer $a \in \left(\frac{\mathbb{Z}}{N\mathbb{Z}}\right)$, to satisfy, $\left(\frac{a}{p}\right) = \left(\frac{a}{q}\right) = -1$
   d) As output receives, (N, a) and $(\, p, \; q)$ as public and private key.
2) *Encryption (Encrypt):*
   a) Choose plaintext $m \in \{0, 1\}$.
   b) Choose random r such that $1 < r < N$.
   c) Compute cipher text c, such that $c = a^m \, r^2 \; mod \; N$ or we can say $c = \begin{cases} r^2 \; mod \; N, \; if \; m = 0 \\ ar^2 \; mod \; N \quad if \; m = 1 \end{cases}$
3) *Decryption (Decrypt):*
   a) Compute $\left(\frac{c}{p}\right)$.
   b) Decrypt ciphertext to get plain text m such that

$$m = \begin{cases} 0, & if \; \left(\dfrac{c}{p}\right) = 1 \\ 1 & if \; \left(\dfrac{c}{p}\right) = -1 \end{cases}$$

The additive homomorphic property for ciphertexts $CT_1 \; and \; CT_2$, for plaintexts $m_1 \; and \; m_2$ is: $CT_1 = Encrypt(m_1, r_1)$ and $CT_2 = Encrypt(m_2, r_2)$, and $CT_1 + CT_2 = Encrypt(m_1 + m_2, r_1 + m_2)$

C. *EL-Gamal's Scheme*

A new signature scheme is proposed by Taher ElGamal in 1984. This scheme is also used to the Diffie-Hellman (DH) key distribution method. The calculating discrete logarithms over finite fields is the problem on which the security of the scheme lies. The author successfully achieve the same level of security as of RSA algorithm but the size of the public key is more and shows the multiplicative homomorphic property [5]. The scheme is explained as follows:

1) *Key Generation (KeyGen):*
   a) Choose a large prime number, p.
   b) Choose x from a Group $G =< Zp^*, X >$ such that, $1 \le x \le p - 1$.
   c) Choose g, a primitive root as a generator in the Group $G =< Zp^*, X >$.
   d) Calculate $y = g^x \; mod \; p$
   e) The public and private keys are, (g, y, p) and x.
2) *Encryption (Encrypt):*
   a) Choose plaintext $m \in \{0, 1\}$.
   b) Choose a random number r, from a Group $G =< Zp^*, X >$ such that, $1 \le r \le p - 1$.
   c) Compute Ciphertext, $C_1 = g^r \; mod \; p$ and $C_2 = (m.y^r) \; mod \; p$
3) *Decryption (Decrypt):*
   a) Decrypt ciphertext to get plain text m such that, $m = [C_2(C_1^x)^{-1}] \; mod \; p$

D. *Benaloh's Scheme*

The author in the year 1994 [6], generalizes the GM scheme to manage inputs of $l(k)$ bits, where k is a prime number. Encryption works identical to the GM algorithm in which selection of an integer $r \in \mathbb{Z}_N$ and computing ciphertext, $c = a^m \, r^2 \; mod \; N$, but the decryption is more complicated having less ciphertext expansion than GM algorithm. For input size $l(k)$ and output $l(n)$, the ciphertext expansion is $\frac{l(n)}{l(k)}$. This makes the scheme more efficient as encryption cost is almost similar to the GM scheme. For every decryption step, a constant overhead is there, i.e., $O\left(\sqrt{k}.l(k)\right)$. A smaller value k, limits the expansion in the ciphertext. The scheme shows the additive homomorphic property.

E. *Naccache-Stern scheme*

This scheme [7] is an improved version of Benaloh's scheme. To reduce the ciphertext expansion further, the author uses a greater value of a parameter k than Benaloh's scheme. The scheme has reduced the cost of the decryption given by $O(l(n)^5 \log\left(l(n)\right)$. The ciphertext expansion value is 4, and can be further reduced by changing the selected parameters such as k. The scheme is partially additive homomorphic.

F. *Okamoto-Uchiyama scheme*

To improvise the performance the author changed the base group G such as $G = \mathbb{Z}^*_{p^2}$ [8]. By taking p and q as large two prime numbers, calculate n i.e., $n = p^2 q$ and changed group G, help to reduce the expansion of ciphertext. The scheme is partial additive and its security relies on the factorization of n against passive attacks. To make the scheme more secure against active attacks random oracle model can be used [9]. The scheme is more susceptible to a chosen-ciphertext attack and hence readily breaks the factorization problem. The scheme is considered suitable for EPOC systems as accepted by IEEE standard.

G. Paillier Scheme

It is one of the most well-known improved HE schemes as the author reduced the ciphertext expansion to its minimum value from 3 to 2, in comparison with other existing HE schemes [10]. As usual, the author uses p and q as two large prime numbers to calculate $n = p.q$ with $\gcd\left(n, \emptyset(n)\right)$. Group, $G = \mathbb{Z}^*_{n^2}$, and parameter H selection, led to a value of $k = l(n)$. The encryption cost is not high but the decryption process is more complex.

The scheme is explained as follows:

1) *Key Generation (KeyGen):*
   a) Take $p\ and\ q,$ as two large prime numbers.
   b) Calculate $n = p.q$ and choose $g = n + 1$.
   c) Calculate $\emptyset(n) = (p-1).(q-1),$ where $\emptyset(n)$ is Euler's Totient function.
   d) Calculate $\mu = \emptyset(n)^{-1}\ mod\ n$
   e) The public and secret keys are (g, n) and μ respectively.
2) *Encryption (Encrypt):*
   a) Select plaintext $m < n$, where $m \in \mathbb{Z}_n$
   b) Pick a random number $r < n$, such that $\gcd(r, n) = 1$, where$r \in \mathbb{Z}^*_n$.
   c) Calculate Ciphertext, $C = g^m.r^n\ mod\ n^2$.
3) *Decryption (Decrypt):*
   a) Ciphertext, $C < n^2,$ and decrypt it using the private key, to retrieve the plain text m such that,

$$m = \left|\frac{\left[C^{\emptyset(n)}\ mod\ n^2\right] - 1}{n}.\mu\ mod\ n\right|$$

The Chinese Remainder Theorem (CRT) is used to manage the decryption process to make the scheme more efficient. Due to reduced, encryption costs and ciphertext expansion, the scheme gets more popular and great acceptance. This scheme is additive homomorphic i.e., $Encrypt(m_1) + Encrypt(m_1) = Encrypt([m_1 + m2 mod\ n.$

H. Damgard - Jurik Scheme

The author proposed the generalization for Pailliler's scheme by using the group $\boldsymbol{G = G = \mathbb{Z}^*_{n^{s+1}}}$, $\boldsymbol{S > 0}$, the larger the value of $\boldsymbol{S}$, the lower will be the ciphertext expansion. The key generation algorithm is identical to the Paillier scheme [11]. To encrypt the plaintext $\boldsymbol{m} \in \mathbb{Z}^*_{\boldsymbol{n}}$,
The scheme is explained as follows:

1) *Key Generation (KeyGen):*
   a) Select two large prime numbers, $p\ and\ q$.
   b) Calculate $n = p.q$ and choose $\lambda = lcm(p - 1, q - 1)$
   c) $g = n + 1^j\ x\ mod\ n^{(S+1)}$.
   d) Choose b, using the CRT, such as $b\ mod\ n \in \mathbb{Z}^*_{\boldsymbol{n}}$ and $b = 0\ mod\ \lambda$.
   e) The public key (g, n) and the private key is b.
2) *Encryption (Encrypt):*
   a) Choose plaintext $m < n$, where $m \in \mathbb{Z}^*_n$.
   b) Choose a random number, $r \in \mathbb{Z}^{*(S+1)}_n$.
   c) Compute the ciphertext, $C = g^m.r^{n^S}\ mod\ n^{(S+1)} \in \mathbb{Z}_{n^{S+1}}$
3) *Decryption (Decrypt):*
   a) Ciphertext, $C < n^2,$ and decrypt it using the private key, to obtain the plain text m such that,

$$m = C^b\ mod\ n^{(S+1)}$$

The ciphertext expansion value can be computed by $1 + \frac{1}{S}$, and it can be further reduced to 1, for a large value of S. The scheme has less ciphertext expansion rate but has more computational overhead. The overall cost is more than Paillier's scheme. It satisfies, $Encrypt(m_1) + Encrypt(m_1) = Encrypt([m_1 + m_2]mod\ n)$.

I. Galbraith scheme

The author presented an adaptive version of Paillier scheme in the milieu of elliptic curves over rings $\frac{\mathbb{Z}}{N^n\mathbb{Z}}$ for $n \geq 3$ [12]. Ciphertext expansion equals 3. For S=1, the ratio of encryption cost and decryption cost is approx. to 7 and 14 respectively. For bigger value of S, the cost of encryption and decryption may be reduced. The security of the scheme improves with the increase in value of S.

J. Boneh-Goh-Nissim Scheme

The author presented a new homomorphic PKE scheme in 2005 [13]. The scheme is based on finite groups of composite order that use a bilinear map. The scheme is explained as follows:

1) *Key Generation (KeyGen):*
   a) Select two large prime numbers, $p\ and\ q$.
   b) Calculate $n = p.q$ and generate a bilinear group, G having order n.
   c) Choose two random generators g and g_1, such that $g, g_1 \in G$.
   d) Compute $h = {g_1}^q$, a random generator from a subgroup of $G,$ having order p.
   e) The public and secret keys are (G, n, g, h) and p.
2) *Encryption (Encrypt):*
   a) Choose plaintext $m \in \{0,1 \dots q\}$, where $m < q$.

b) Choose a random number, $r \in \{0,1 \dots n-1\}$.

c) Compute the ciphertext, $C = g^m . h^r \in \mathrm{G}$

3) *Decryption (Decrypt):*

a) Decrypt the ciphertext using secret key, p, to get plain text m such that, $C^p = (g^m . h^r)^p = (g^q)^m$

Finally, to compute m, find a discrete log of C^p, i.e., $C^p\ base\ g^q$. The security of the scheme relies on subgroup decisional problem. If $n = p.q,$ is an element from a group $G,$ of composite order then it is hard to find out from which subgroup has order p or q, it belongs. The author also shows the evaluation of the 2-DNF (Disjunctive Normal Form formula on ciphertexts [14] [15].

K. *Castagnos's scheme*

The author improved the performance of existing HE schemes, by proposing a new probabilistic encryption scheme using quadratic field quotations [16]. For the value of S=1, scheme achieves the ciphertext expansion and ratio of encryption/decryption as 3 and 2 respectively. The security of the system relies on the new decisional problem and LUC problem related to Lucas sequence [16]. The encryption cost is much smaller than the El-Gammal. The computational cost of the scheme is also reduced as the LUC sequence and LUC function help to perform the computation faster. The scheme is faster than the Galbraith scheme [12].

III. Historical Evolution of Homomorphic Encryption

Privacy homomorphism i.e., the idea to perform computation over encrypted data was first introduced by Rivest, Shamir, and Adleman in their cryptographic algorithm, i.e., RSA which later turned into a concept of today, i.e., Homomorphic Encryption. The RSA's security rely on the problem of factoring large integers. Due to the deterministic nature of the scheme, it is not much secure as generated ciphertext can easily be guessed for its respective plaintext, as always, the same ciphertext generates for the same plaintext. The multiplicative property of RSA is also not used in a real-life application and various improvements to the basic RSA algorithm also tend to lose its homomorphic property [17].

The GM scheme by Goldwasser & Micali (1982) [3] is the idea of the RSA algorithm i.e., compute, modulo $n = p.q$, a product of two large primes. The scheme's security relies on the intractable quadratic residuosity problem [15]. The algorithm's core premise is to partition a subset of integers modulo n into two secret parts $F_0\ and\ F_1$. Determining the subset and its partition is critical that can be done using group theory. The subset is the group G of invertible integers modulo n with a Jacobi symbol equal to 1 with regard to n. With these parameter values, G may be divided into two parts: H and G\H. A new signature scheme is presented by Taher ElGamal in 1985 [5]. The scheme is an asymmetric encryption scheme having the same sized numbers and achieving the same level of security but the size of the public key is greater than RSA. The algorithm is partially multiplicative homomorphic encryption [5]. The scheme can be cracked up in sub-exponential time [14] due to its deterministic nature like RSA but still can be considered a hard problem [18].

The author, Benaloh in the year 1994 [6], generalizes the GM scheme to improve it in terms of ciphertext size. The scheme is probabilistic and additive homomorphic. The decryption algorithm is more complex and smaller ciphertext. For every decryption step, a constant overhead is there, i.e., $O\left(\sqrt{k}.\,l(k)\right)$. In 1998, Naccache & Stern introduced another scheme to improve Benaloh's scheme in terms of computational efficiency [7] and also reduce the ciphertext size by choosing a greater value of $k,$ than Benaloh's scheme. The decryption algorithm is also modified to reduce the cost [7]. The scheme is partially additive homomorphic. In the same year, author improves the performance of the earlier PHE scheme by changing the group G, that helps to reduce the expansion of ciphertext for the proposed scheme [8]. To make the scheme more secure against active attacks random oracle model can be used [9]. The scheme is partially additive homomorphic.

In 1999, Paillier presented probabilistic PHE scheme, based on the concept of the GM scheme that was additive homomorphic [10]. With the change, Group, $G = \mathbb{Z}^*_{n^2}$, and specified parameter selection, the author, achieves the least ciphertext expansion, and reduced encryption cost. It is one of the most popular and improved HE schemes. Decisional Composite Residuosity Assumption (DCRA) problem, makes the scheme secure. In 2001, Damgård and Jurik [11] presented the generality of Pailliler's scheme by modifying the group. The scheme is more intensive in terms of computation but reduces the ciphertext expansion. The overall cost to execute the scheme is more than Paillier's scheme. The scheme is additive homomorphic. In 2002, The author presented an adaptive version of Paillier scheme related to the elliptic curves over rings while preserving the homomorphic property of the scheme [12]. Overall cost can be decreased with improved security. Cramer and Shoup [19] presented another variant of the PHE scheme, based on Paillier's scheme but stronger than that in 2002**.** With some specified algebraic properties, the scheme is more secure against adaptive chosen-ciphertext attacks. Specified parameter selection helps to reduce the public key and ciphertext size.

In 2003, Bresson et al. [1] presented a somewhat modified version of Cramer Shoup's HE schemes that is additive homomorphic. Two separate decryption versions

Journal of Current Research in Engineering and Science
JCRES
Bi-Annual Online Journal (ISSN : 2581 - 611X)

based on two different trapdoors are given. The scheme is additive homomorphic and security does not rely on a factorization problem but instead on a residuosity-related assumption problem.

Boneh et al. presented a new homomorphic PKE scheme in 2005 [13]. BGN scheme is somewhat homomorphic, as it shows additive homomorphic property and also supports only one multiplication operation on ciphertexts. It provides a foundation for the construction of Fully Homomorphic Encryption (FHE). The author also shows the evaluation of the 2-DNF formula on ciphertexts using, his scheme, popularly known as the BGN scheme, and ciphertext size also remains constant [14][15].

Earlier in 1982 and 1999, two somewhat HE encryption schemes are introduced by Yao [20] and Sander et al. [21] respectively. Yao's tried to compute operations over encrypted text using a circuit, later popularly known as Yao's garbled circuit. The author proposed a solution to the 'Millionaires Problem' where the wealth of two rich people can be compared without revealing another one. The proposed scheme is not much efficient as ciphertext shows linear growth with each computation and also has more computational overhead. The overall scheme is not favored due to its too much complexity [15]. Sander et al. (SYY) [21], proposed an idea to evaluate operation over encrypted text but on a different set, i.e., a semigroup, NC1. NC1 semigroup possesses fewer properties in comparison to a group and has poly-logarithmic depth and polynomial-sized circuits. The circuit is comprised of polynomially-many AND gates and one OR/NOT gate that improves the depth of the evaluation circuit. Ciphertext size grows exponentially with each operation.

In 2007, Ishai and Paskin (IP) [22] proposed another PKE scheme to evaluate the branching programs (binary decision programs with labels 0 or 1), over encrypted text for a limited length of the program. The details between the server and client are hidden, like the size of the input to the server from the client, i.e., the size of the branching program. The scheme is improved in terms of the number of computations and ciphertext size, also independent of the function or length of the branching program. In 2007, Kawachi et al. [23], presented a PHE scheme over a cyclic group. The scheme is additive homomorphic. The pseudo-homomorphism is an algebraic condition that permits homomorphic operations on encrypted text and decryption results in original plaintext with a small decryption error. The scheme's security is dependent on the hardness of lattice problems [15].

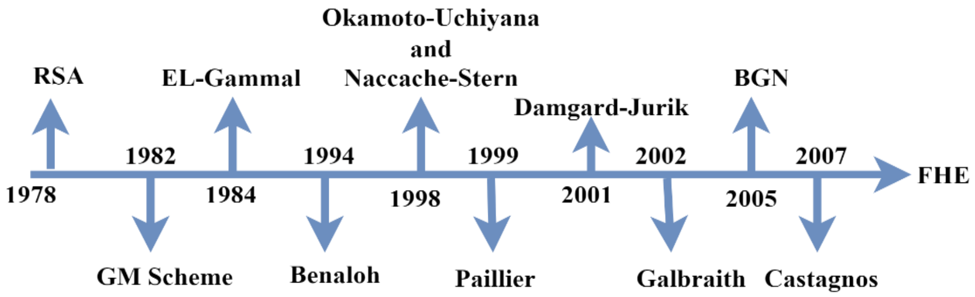

Figure 1 Evolution of Classical Homomorphic Encryption Schemes

IV. Comparative Analysis and Result

Fig. 1 shows the evolution of classical homomorphic encryption schemes. In fact, researchers have been working on the problem of HE since 1978 starting with the RSA algorithm [2] but haven't got any scheme for a long time. Table 2.1 shows the analysis of various classical HE schemes.

Table 2. 1 Analysis of Classical Homomorphic Encryption Schemes

Year	Scheme	Additive	Multiplicative	Ciphertext Expansion Ratio
1978	RSA	✕	✓	1
1982	GM	✓	✕	$log_2(n)$
1984	EL-Gamal	✕	✓	2
1994	Benaloh	✓	✕	$log_2(n)/log_2(r)$
1998	Naccache & Stern	✓	✕	$log_2(n)/log_2(r)$
1998	Okamoto & Uchiyama	✓	✕	3
1999	Paillier	✓	✕	2
2001	Damgard-Jurik	✓	✕	$log_2(n^{S+1})/log_2($
2002	Galbraith	✓	✕	3
2005	BGN	✓	✓	$log_2(n)/log_2(r)$
2007	Castagnos	✓	✕	3

It has been found that the author, Castagnos improved the performance of existing HE schemes, by proposing a new probabilistic encryption scheme using quadratic field quotations [16]. The security of the system relies on the new decisional problem and LUC problem. The scheme is faster than the Galbraith scheme and performs computation faster. Melchor et al. [24], presented a new chained encryption scheme, i.e., a combination of BGN [13] and the Kawachi's scheme [23]. As BGN supports unlimited additions and one multiplication and Kawachi's scheme is additive, so the new combined scheme supports homomorphic unlimited additions and two multiplication evaluations up to a constant depth circuit. However, the ciphertext size increases exponentially for multiplication whereas remains constant for an addition operation [14][25].

V. CONCLUSION AND FUTURE SCOPE

The paper presents the study of different classical homomorphic encryption schemes before Gentry's fully homomorphic encryption scheme in 2009 which is given by researcher, Craig Gentry. The Study shows that the various classical encryption schemes are either additive or multiplicative except for the BGN scheme which is somewhat homomorphic. This study may help the researchers to understand the history of homomorphic encryption schemes and their evolution to fully homomorphic encryption schemes.

REFERENCES

[1] E. Bresson, D. Catalano, and D. Pointcheval, "A Simple Public-Key Cryptosystem with a Double Trapdoor Decryption Mechanism and Its Applications," in Lecture Notes in Computer Science (Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2003, vol. 2894, pp. 37–54, doi: 10.1007/978-3-540-40061-5_3.

[2] R. R. L. . L. A. and M. L. Dertouzos., "On Data Banks and Privacy Homomorphisms," Found. Secur. Comput., vol. 4, no. 11, pp. 169-180., 1978, doi: 10.1075/jpcl.15.2.04lef.

[3] S. Goldwasser and S. Micali, "Probabilistic Encryption," Journal of Computer and System Sciences, vol. 28, no. 2. pp. 270–299, 1984, doi: 10.1016/0022-0000(84)90070-9.

[4] N. Jain, "Implementation and Analysis of Homomorphic Encryption Schemes," Int. J. Cryptogr. Inf. Secur., vol. 2, no. 2, pp. 27–44, 2012, doi: 10.5121/ijcis.2012.2203.

[5] T. Elgamal, "A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms," IEEE Trans. Inf. Theory, vol. 31, no. 4, pp. 469–472, 1985, doi: 10.1109/TIT.1985.1057074.

[6] J. Benaloh, "Dense Probabilistic Encryption.," in Proceedings of the workshop on selected areas of cryptography, Berlin, Heidelberg, 1994, pp. 120–128.

[7] D. Naccache and J. Stern, "A New Public Key Cryptosystem Based on Higher Residues," in Proceedings of the ACM Conference on Computer and Communications Security, 1998, pp. 59–66, doi: 10.1145/288090.288106.

[8] T. Okamoto and S. Uchiyama, "A New Public-Key Cryptosystem As Secure As Factoring," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 1403, pp. 308–318, 1998, doi: 10.1007/BFb0054135.

[9] R. Canetti, O. Goldreich, and S. Halevi, "The Random Oracle Methodology, Revisited," J. ACM, vol. 51, no. 4, pp. 557–594, 2004, doi: 10.1145/1008731.1008734.

[10] P. Paillier, "Public-Key Cryptosystems Based on Composite Degree Residuosity Classes," in Advances in Cryptology - EUROCRYPT '99, International Conference on the Theory and Application of Cryptographic Techniques, 1999, pp. 223–238, doi: 10.1007/3-540-48910-X_16.

[11] I. B. Damgård and M. J. Jurik, "A Generalisation, a Simplification and some Applications of Paillier's Probabilistic Public-Key System," BRICS Rep. Ser., vol. 7, no. 45, pp. 119–136, 2000, doi: 10.7146/brics.v7i45.20212.

[12] S. D. Galbraith, "Elliptic Curve Paillier Schemes," J. Cryptol., vol. 15, no. 2, pp. 129–138, 2002, doi: 10.1007/s00145-001-0015-6.

[13] D. Boneh, E. J. Goh, and K. Nissim, "Evaluating 2-DNF formulas on ciphertexts, Berlin; Heidelberg," in Lecture Notes in Computer Science, 2005, vol. 3378, pp. 325–341, doi: 10.1007/978-3-540-30576-7_18.

[14] J. Sen, "Homomorphic Encryption: Theory and Application," in Theory and Practice of Cryptography and Network Security Protocols and Technologies, J. Sen, Ed. BoD – Books on Demand, 2013, p. 158.

[15] A. Acar, H. Aksu, A. S. Uluagac, and M. Conti, "A Survey on Homomorphic Encryption Schemes: Theory and Implementation," ACM Comput. Surv., vol. 51, no. 4, pp. 1–35, 2018, doi: 10.1145/3214303.

[16] G. Castagnos, "An Efficient Probabilistic Public-Key Cryptosystem Over Quadratic Fields Quotients," Finite Fields Their Appl., vol. 13, no. 3, pp. 563–576, Jul. 2007, doi: 10.1016/J.FFA.2006.05.004.

[17] A. Wood, K. Najarian, and D. Kahrobaei, "Homomorphic Encryption for Machine Learning in Medicine and Bioinformatics," ACM Comput. Surv., vol. 53, no. 4, pp. 1–35, 2020, doi: https://doi.org/10.1145/3394658.

[18] P. Martins, L. Sousa, and A. Mariano, "A Survey on Fully Homomorphic Encryption: An Engineering Perspective," ACM Comput. Surv., vol. 50, no. 6, pp. 1–33, 2017, doi: 10.1145/3124441.

[19] R. Cramer and V. Shoup, "Universal Hash Proofs and a Paradigm for Adaptive Chosen Ciphertext Secure Public-Key Encryption," in Lecture Notes in Computer Science (Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2002, vol. 2332, pp. 45–64, doi: 10.1007/3-540-46035-7_4.

[20] A. C. Yao, "Protocols for Secure Computations," in 23rd Annual Symposium on Foundations of Computer Science (sfcs 1982), Chicago, IL, USA, USA, 1982, pp. 160–164, doi: 10.1109/SFCS.1982.38.

[21] T. Sander, A. Young, and M. Yung, "Non-Interactive Cryptocomputing for NC," in 40th Annual Symposium on Foundations of Computer, New York City; NY; USA, 1999, pp. 554–566, doi: 10.1109/SFFCS.1999.814630.

[22] Y. Ishai and A. Paskin, "Evaluating Branching Programs on Encrypted Data," in Theory of Cryptography. TCC 2007. Lecture Notes in Computer Science, 2007, vol. 4392, pp. 575–594, doi: https://doi.org/10.1007/978-3-540-70936-7_31.

[23] A. Kawachi, K. Tanaka, and K. Xagawa, "Multi-bit Cryptosystems Based on Lattice Problems," in Lecture Notes in Computer Science (Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2007, vol. 4450, pp. 315–329, doi: 10.1007/978-3-540-71677-8_21.

[24] C. A. Melchor, P. Gaborit, and J. Herranz, "Additively Homomorphic Encryption with d-Operand Multiplications," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2010, vol. 6223, pp. 138–154, doi: 10.1007/978-3-642-14623-7_8.

[25] K. Aulakh, and K. R. Ramachandran, R.K., "A Detailed Survey of Fully Homomorphic Encryption Standards to Preserve Privacy over Cloud Communications." in 2020 Indo–Taiwan 2nd International Conference on Computing, Analytics and Networks (Indo-Taiwan ICAN), 2020, pp. 207-211, IEEE.

[26]. J. Kaur, and K. R. Ramkumar, K.R., "The recent trends in cyber security: A review.", Journal of King Saud University-Computer and Information Sciences, vol. 34, no.8, pp.5766-5781, 2022, https://doi.org/10.1016/j.jksuci.2021.01.018.