Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X
IMPACT FACTOR: 7.056

IJCSMC, Vol. 14, Issue. 10, October 2025, pg.21 – 26

HOMOMORPHIC ENCRYPTION: AN OVERVIEW

Solomon Sarpong[1]; Samuel Opuni-Basoa[2]; Kwaafo Akoto Awuah-Mensah[3]

[1,2,3]Department of Physical and Mathematical Sciences, University of Environment and Sustainable Development, Somanya, Ghana
[1] ssarpong@uesd.edu.gh; [2] sopbasoa@uesd.edu.gh; [3] kaawuah-mensah@uesd.edu.gh

DOI: https://doi.org/10.47760/ijcsmc.2025.v14i10.002

Abstract: The tendency of having accessibility to information anytime and anywhere has contributed to individuals or organizations shifting from the storage of information to Internet-based cloud services. This shift has made data more vulnerable as data in the cloud cannot be operated on unless the computation is outsourced to a third-party which can be compromised. Furthermore, in order to keep data secure and accessible securely, homomorphic encryption techniques can be used. Homomorphic encryption supports computation on ciphertexts without decryption first. This paper gives a comprehensive survey of homomorphic encryption.

Keywords: Homomorphic, non-repudiation, semantic security, sparse subset sum, side-channel attack.

I. INTRODUCTION

Cryptography ensures the security, confidentiality and integrity of information whether at rest or in transit. Hence, it ensures secure communication between two or more parties. Confidentiality: only the intended recipient can understand the information communicated. Integrity: whether the information is at rest or in transit, it cannot be modified without being detected. Non-repudiation: There cannot be deniability about the creation or transmission of the information by the creator. Authentication: both the sender and receiver know the identity and origin/destination of the information.

Homomorphism as an English word, originated from the Greek word – *homosmorphe*. *Homos* means "same" and *morphe* means "shape". In cryptography, homomorphic encryption (HE) is a technique for converting plaintext to cyphertext. HE enables users to mathematically operate on encrypted information without decrypting it first. As a form of encryption, HE allows computation on cyphertexts with the resulting cyphertext same as the computation on the plaintext. Thus, ***A*** can compute the product of two encrypted numbers which can be decrypted by ***B*** to obtain the individual numbers without ***B*** knowing the actual values of the individual numbers. That is, two encrypted numbers can be added or multiplied and the output decrypted without knowing the individual numbers. This property helps to protect the privacy and security of the encrypted data whether at rest or in transit. Furthermore, HE ensures privacy and confidentiality of information by ensuring that,

operations can be performed on the encrypted information without the need for it to be decrypted [1]. [2] has presented various trends and challenges of HE.

HE enables confidential information to be transmitted across computer systems/stored in databases. Hence, in homomorphic encryption, information can be converted into cyphertext which can be operated on as if it was the plaintext [3]. There are two types of homomorphic encryption schemes namely: additive and multiplicative. In the additive homomorphic schemes, $E(m_1 + m_2) = E(m_1) \oplus E(m_2)$ without knowing m_1 and m_2. Examples of additive homomorphic encryption schemes include: Paillier, Goldwasser-Micali, Benaloh and Damgard-Jurik, Schmidt-Samoa-Takagi and Okamota-Uchiyama cryptosystems. On the other hand, in multiplicative homomorphic encryption schemes, $E(m_1 \otimes m_2) = E(m_1) \otimes E(m_2)$ without knowing m_1 and m_2. The multiplicative homomorphic encryption schemes include: RSA, Sanders-Young-Yung and ElGamal cryptosystems.

The objective of homomorphic encryption is to manage and protect the information of users that is stored on untrusted devices. Due to the unique nature of homomorphic encryption, it is used in cryptosystems such as multi-party computation to secure computation. The security of cryptosystems of which homomorphic encryption is also part is based on the intractability of: Integer factorization problem, discrete logarithm problem and quadratic residousity problem. The diverse applications of HE is the motivation for this paper as it seeks to give a comprehensive survey of homomorphic encryption by providing the basic definitions, properties of primitive cryptographic techniques, the security and applications. This paper highlights the types of homomorphic encryption; some cryptographic primitives; the correctness of HE schemes, limitations and some attacks on homomorphic encryption.

II. TYPES OF HOMOMORPHIC ENCRYPTION

The idea of homomorphic encryption was conceived by (Rivest et al., 1978). With its ability to allow calculations over encrypted data, it has found applications in a lot of cryptosystems. There have been several forms of HE since its proposition by (Rivest et al., 1978) until Gentry (2009) proposed a public key encryption scheme of the HE namely, fully homomorphic encryption (FHE) [3]. This scheme has the capability of any kind of operation. HE is classified by the number of mathematical operations that can be performed on an encrypted data. In lieu of this, it can either be somewhat homomorphic encryption (SHE), partially homomorphic encryption (PHE) and fully homomorphic encryption (FHE).

A. Somewhat Homomorphic Encryption (SHE)

This cryptosystem allows addition and multiplication operations a limited number of times on encrypted data as each operation adds noise. Hence, the data is irretrievable after a certain amount of noise is added. Boneh-Goh-Nissin cryptosystem (BGN) [5] is an example of SHE. The security of BGN relies on the assumption of the subgroup decision problem [3], and [6]

B. Partially Homomorphic Encryption (PHE)

This cryptosystem allows either addition or multiplication operations on encrypted data. Examples of PHE include: Goldwasser-Micali cryptosystem [7] that has semantic security based on the quadratic residuousity problem; Benaloh cryptosystem [8] with its security based on higher residuousity problem; Naccache-Stern cryptosystem [9], [10] has security also based on higher residuousity problem; Okamoto-Uchiyama cryptosystem [11] has security based on *p*-subgroups assumption which is identical to the quadratic residuousity problem and higher residuousity problem; Paillier cryptosystem [12] has semantic security based on the ssumption of the decisional composite residuousity problem; Damgard-Jurik cryptosystem [13] has semantic security based on the assumption of the decisional composite residuousity problem; Riverst-Shamir-Adlemann (RSA) cryptosystem [14], [15], [16]; ElGamal cryptosystem [17], [18] has semantic security based on the hardness of the Diffie-Hellman problem.

C. Fully Homomorphic Encryption

This cryptosystem allows addition and multiplication operations on encrypted data any number of times. The semantic security of this cryptosystem relies on sparse subset sum problem (SSSP) and assumptions [6] and [19].

III. CRYPTOGRAPHIC PRIMITIVES

This section takes a look at some cryptographic primitives.

A. Paillier Cryptosystem

Paillier cryptosystem [12] is an example of partial homomorphic encryption. This HE is an additive homomorphic cryptosystem based on the composite residousity problem. This cryptosystem requires a

quadruple: (*keyGen*, *Enc*, *Dec*, *Add*). *KeyGen(λ)*: compute *p* and *q* of length *k* bits randomly such that, $N = pq$ and $\rho = N^{-1}(mod\ \varphi(N))$ where $\varphi(N) = (p-1)(q-1), p$ and q are two large random primes of length *k* units. The values for $pk = N$ and $sk = \rho$ can be published. $Enc(\rho k, m \in Z)$: to encrypt a message *m*, compute $c = (1+mN)r^N(modN^2)$ where $r \xleftarrow{random} \{1,\ldots, N-1\}$. $Dec(sk, c)$: to decrypt the cyphertext, c compute $m = \frac{(cr^{-N}(mod\ N^2))-1}{N}$ where $r = c^{\rho}(mod\ N)$.

$Add(c_1, c_2)$: compute $c_1c_2 = E(m_1)E(m_2) = (1+m_1N)r_1^N(1+m_2N)r_2^N(mod\ N^2) = [1+(m_1+m_2)N + m_1m_2N^2](r_1r_2)^2(mod\ N^2) = [1+(m_1+m_2)N]r^N(mod\ N^2) = E(m_1+m_2)$

.

B. ElGamal Cryptosystem

[20] is an example of multiplicative encryption and it is based on the Diffie-Hellman key exchange protocol. The security of ElGamal cryptosystem is derived from the hardness of the discrete logarithm problem. This cryptosystem requires a quadruple: $(keyGen, Enc, Dec, Multi)$.

$keyGen(\lambda)$: a cyclic group G of order N and a generator $g \in Z_N^*$ are chosen. Compute $h = g^x$ where x is a random integer from $\{1,\ldots, N-1\}$. The public and secret keys are (G, N, g, h) and x respectively.

$Enc(pk, m \in Z)$: to encrypt a message, *m*, compute $s = h^y, c_1 = g^y$ and $c_2 = ms$ where $y \in \{1,\ldots, N-1\}$. The cyphertext is $C = (c_1, c_2)$ compute $s' = c_1^x$ and $m = c_2(s')^{-1}$ to retrieve the message, *m*.

$Multi(c_1, c_2)$: compute $c_1c_2 = E(m_1)E(m_2) = (g^{x_1}, m_1h^{x_2}) * (g^{x_1}, m_2h^{x_2}) = (g^{x_1+x_2}, m_1m_2h^{x_1x_2}) = E(m_1m_2)$

.

C. RIVEST, SHAMIR AND ADLEMAN (RSA)

RSA [21] is multiplicatively homomorphic. The RSA is homomorphic encryption cryptosystem whose security is based on the hardness of the factoring problem. With a security parameter *λ*, RSA requires a quadruple: $(KeyGen, Enc, Dec, Multi)$.

$KeyGen(\lambda)$: randomly choose two large primes *p* and *q* and compute the following $N = p$, $\emptyset(N) = (p-1)(q-1)$ and $gcd(d, \emptyset(N)) = 1$ where *d* is a large integer. The multiplicative inverse of *d* is computed such that, $ed \equiv 1 mod\ \emptyset(N)$. The public and secret keys are $pk = (e, N)$ and $sk = (d, p, q)$ respectively.

$Enc(pk, m \in Z_N)$: to encrypt message $m \in \{1,\ldots, N-1\}$, compute $c = E(m) = m^e(mod\ N)$.

$Dec(sk, c)$: the message *m* can be obtained from the cyphertext *c* by computing $m = c^d(mod\ N)$.

$Multi(c_1, c_2)$: $c_1c_2 = E(m_1)E(m_2) = [m_1^e(mod\ N)][m_2^e(mod\ N)] = (m_1m_2)^e(mod\ N) = E(m_1m_2)$.

IV. CORRECTNESS OF HOMOMORPHIC ENCRYPTION

Assume $p \in N^+$ be a large integer as the secret key. Let *a* and *b* be two arbitrary positive integers with $a, b \ll p \in N^+$.

An additive homomorphic encryption is correct if: $a + b < p$.

$(a' + b') = (a + r_1p) + (b + r_2p) = a + b + (r_1 + r_2)p$

$a + b + (r_1 + r_2)p\ mod\ p = (a + b + rp)\ mod\ p = a + b$

A multiplicative homomorphic encryption is correct if: $a \otimes b < p$.

$a' \otimes b' = (a + r_1p)(b + r_2) = ab + a(r_2p) + b(r_1p) + r_1r_2p^2$

$(ab + a(r_2p) + b(r_1p) + r_1r_2p^2)\ mod\ p = ab$

V. LIMITATIONS OF HOMOMORPHIC ENCRYPTION

One of the main limitations of the computational overhead, which makes it slower than traditional encryption methods. Another limitation is the size of the plaintext and cyphertext, which can be significantly larger than the original information. Additionally, homomorphic encryption is currently only practical for specific types of computations such as addition and multiplication and cannot be used for more complex operations. Finally, homomorphic encryption is vulnerable to side-channel attacks, which can leak information about the encrypted information through its execution time or power consumption.

VI. SOME ATTACKS ON HOMOMORPHIC ENCRYPTION

[22] presented various attacks on homomorphic encryption based on the weakness in the underlying mathematical assumptions. Side-channel attacks [19]: this type of attack is based on the assumption that, an adversary has access to some information about the secret key of the encryption algorithm. Side-channel attack is carried out when an adversary lunches timing attack on the system. That is the adversary checks the time it takes for the encryption/decryption to be undertaken. This form of attack can be thwarted when the homomorphic encryption scheme has leakage resilience semantic properties.

Black box attacks [23]: this form of attack on homomorphic encryption happens when an adversary manipulates encrypted data that has been accessed without success because the secret key is not known. However, the aim of the adversary is to gain some information about the plaintext when the output of the homomorphic encryption is examined. This form of attack can be thwarted by randomized encoding i.e. adding random numbers to the plaintext before encryption.

Lattice attack [24]: this form of attack occurs when an adversary tries to exploit the vulnerabilities in the lattice structures of a lattice-based cryptosystem with the aim to compromise the secret key.

VII. APPLICATIONS OF HOMOMORPHIC ENCRYPTION

Homomorphic encryption is used in healthcare to secure patients' data to prevent unauthorized access. In financial services, it is used to secure and ensure confidentiality of the transactions of businesses and clients. Homomorphic encryption is also used in smart grid. Each generator is considered as a node. At each node, data such as electrical generation and usage, temperature of the physical equipment in use, energy flow, etc. are generated. In order to undertake any computation on the smart grid, the application of homomorphic encryption preserves the privacy and security of the nodes. Furthermore, homomorphic encryption is used in genomics – DNA and RNA sequences. Genomic data generated from individuals are as unique as fingerprint. Hence, it can be used to identify an individual.

Data need to be shared among scientists but there are restrictions on sharing genomic data however, the application of homomorphic encryption helps to overcome such issue. Homomorphic cryptosystems are applied in the following: security in cloud computing [13] and [25]; private use of untrusted webservers [25] and [26]; protection of mobile agents [25], [27], [28], [29]; multiparty computation [30], [10], [25]; secret sharing schemes (Beimel, 2011), [25], [32] ; threshold schemes [25], [33], [34], [25][27]; zero-knowledge proofs [25], (Dhokrat et al., 2024.), [36], [13] election schemes [25], [37], [38], [39], [40]; lottery protocols [25], [33], [41]. [42]

In order to work on encrypted data in a hospital and ensure its security and privacy,[43] proposed the use of homomorphic encryption. The need for secure data processing in IoT in digital healthcare systems has necessitated the use of edge computing coupled with information protection and data integrity. However, this does not provide enough security in the face of a powerful adversary. Hence, [44] proposed the use of Paillier homomorphic encryption and certificateless linear homomorphic ID-based signatures in HC-SDP models for data protection in digital healthcare systems. [45] applied neural network via homomorphic encryption in secure privacy-preserving access to information. Homomorphic encryption has been applied in various ways in industry such as cloud computing, electronic voting and IoT [46] and [47].

They, [48], reviewed the optimization of HE in privacy-preserving federated learning. A comprehensive performance comparison of different schemes on the same platform using different HE schemes was done using theoretical and engineering optimizations, [49]. An optimized HE-based convolutional neural network framework for real-time applications was proposed by [50].

VIII. CONCLUSION

This paper surveys homomorphic encryption. Homomorphic encryption enables us to perform computations on encrypted information without first decrypting it. Hence, there can be computations on a cyphertext by multiple parties without decrypting it first. In lieu of this, homomorphic encryption eliminates key distribution problems in multi-party computations. Hence, homomorphic encryption finds applications in several fields where computation needs to be done on encrypted data by multiple entities.

REFERENCES

[1]. M. Alkharji and H. Liu, "Homomorphic Encryption Algorithms and Schemes for Secure Computations in the Cloud," p. 19, 2016, doi: 10.13140/RG.2.2.19574.41285.

[2]. B. Alaya, L. Laouamer, and N. Msilini, "Homomorphic encryption systems statement: Trends and challenges," *Comput Sci Rev*, vol. 36, p. 100235, May 2020, doi: 10.1016/J.COSREV.2020.100235.

[3]. C. Gentry, "A Fully Homomorphic Encryption Scheme," *Dissertation*, no. September, p. 169, 2009, [Online]. Available: http://cs.au.dk/~stm/local-cache/gentry-thesis.pdf

[4]. R. Rivest, L. Adleman, and ML Dertouzos, "On data banks and privacy homomorphisms," *CiteseerRL Rivest, L Adleman, ML DertouzosFoundations of secure computation, 1978•Citeseer*, vol. 4(11), pp. 169–180, 1978, Accessed: Apr. 13, 2025. [Online]. Available: https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=c365f01d330b2211e74069120e88cff37eacbcf5

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

[6]. D. J. Wu, "Fully Homomorphic Encryption: Cryptography's holy grail," *XRDS: Crossroads, The ACM Magazine for Students*, vol. 21, no. 3, pp. 24–29, Mar. 2015, doi: 10.1145/2730906.

[7]. S. Goldwasser and S. Micali, "Probabilistic encryption & how to play mental poker keeping secret all partial information," *Proceedings of the Annual ACM Symposium on Theory of Computing*, pp. 365–377, May 1982, doi: 10.1145/800070.802212.

[8]. J. Benaloh, "Dense Probabilistic Encryption.," Proceedings of the Workshop on Selected Areas of Cryptography, Kingston.

[9]. J. M. Kukucka and M. Zuker, "An investigation of the theory and applications of homomorphic cryptography," *Rensselaer Polytechnic Institute*, 2013.

[10]. D. S. Maimut, A. Patrascu, and E. Simion, "Homomorphic Encryption Schemes and Applications for a Secure Digital World," *Journal of Mobile, Embedded and Distributed Systems*, vol. 4, no. 4, pp. 224–232, Dec. 2012, Accessed: Mar. 25, 2025. [Online]. Available: http://jmeds.eu/index.php/jmeds/article/view/83

[11]. T. Okamoto and S. Uchiyama, "A new public-key cryptosystem as secure as factoring," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 1403, pp. 308–318, 1998, doi: 10.1007/BFB0054135.

[12]. P. Paillier, "Public-Key Cryptosystems Based on Composite Degree Residuosity Classes," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 1592, pp. 223–238, 1999, doi: 10.1007/3-540-48910-X_16.

[13]. I. Damgård, N. Fazio, and A. Nicolosi, "Non-interactive Zero-Knowledge from Homomorphic Encryption," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 3876 LNCS, pp. 41–59, 2006, doi: 10.1007/11681878_3.

[14]. I. Ahmad and A. Khandekar, "Homomorphic Encryption Method Applied to Cloud Computing," *International Journal of Information & Computation Technology*, vol. 4, pp. 1519–1530, 2014, doi: 10.13140/RG.2.2.19574.41285.

[15]. P. V Parmar *et al.*, "Survey of Various Homomorphic Encryption algorithms and Schemes," *Int J Comput Appl*, vol. 91, no. 8, pp. 975–8887, 2014, doi: 10.5120/15902-5081.

[16]. R. Meissen and W. J. Martin, "A Mathematical Approach to Fully Homomorphic Encryption," Apr. 26, 2012, *Worcester Polytechnic Institute*. Accessed: Mar. 27, 2025. [Online]. Available: https://digital.wpi.edu/concern/student_works/2227mr290?locale=en

[17]. A. E. Khalid El Makkaoui and A. B.-H. Cina Motamed, "Data condentiality in the world of cloud," *J Theor Appl Inf Technol*, vol. 84, no. 3, Feb. 2016, Accessed: Mar. 27, 2025. [Online]. Available: https://www.jatit.org/volumes/Vol84No3/1Vol84No3.pdf

[18]. C. Fontaine and F. Galand, "A survey of homomorphic encryption for nonspecialists," *EURASIP J Inf Secur*, vol. 2007, 2007, doi: 10.1155/2007/13801.

[19]. M. Chase *et al.*, "Security of homomorphic encryption," 2017.

[20]. 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.

[21]. R. L. Rivest, A. Shamir, and L. Adleman, "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems," *Commun ACM*, vol. 21, no. 2, pp. 120–126, Feb. 1978, doi: 10.1145/359340.359342.

[22]. B. Li and D. Micciancio, "On the Security of Homomorphic Encryption on Approximate Numbers," 2021, doi: 10.1007/978-3-030-77870-5_23.

[23]. A. Borovik and Ş. Yalçınkaya, "Homomorphic Encryption and Some Black Box Attacks," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2020. doi: 10.1007/978-3-030-52200-1_11.

[24]. G. Chunsheng, "Attack on Fully Homomorphic Encryption over the Integers," *CoRR*, vol. abs/1202.3321, 2012, doi: 10.11591/ijins.v1i4.798.

[25]. 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, Sep. 2018, doi: 10.1145/3214303.

[26]. K. Schmidt-Samoa and T. Takagi, "Paillier's Cryptosystem Modulo p 2 q and Its Applications to Trapdoor Commitment Schemes," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 3715 LNCS, pp. 296–313, 2005, doi: 10.1007/11554868_21.

[27]. T. Sander and C. F. Tschudin, "Towards mobile cryptography," *Proceedings of the IEEE Computer Society Symposium on Research in Security and Privacy*, pp. 215–224, 1998, doi: 10.1109/SECPRI.1998.674837.

[28]. M. A. Will, B. Nicholson, M. Tiehuis, and R. K. L. Ko, "Secure voting in the cloud using homomorphic encryption and mobile agents," in *Proceedings - 2015 International Conference on Cloud Computing Research and Innovation, ICCCRI 2015*, 2016. doi: 10.1109/ICCCRI.2015.30.

[29]. L. Hyungjick, J. Alves-Foss, and S. Harrison, "The use of encrypted functions for mobile agent security," *Proceedings of the Hawaii International Conference on System Sciences*, vol. 37, no. C, pp. 4757–4766, 2004, doi: 10.1109/hicss.2004.1265700.

[30]. R. Kanagavalli, "A Survey of Homomorphic Encryption Schemes in Cloud Data Storage," *International Journal of Recent Development in Engineering and Technology Website: www.ijrdet.com*, vol. 3, no. 1, 2014, Accessed: Mar. 27, 2025. [Online]. Available: www.ijrdet.com

[31]. A. Beimel, "Secret-Sharing Schemes: A Survey," *International Workshop on Coding and Cryptology*, vol. 6639 LNCS, pp. 11–46, 2011, doi: 10.1007/978-3-642-20901-7_2.

[32]. E. F. Brickell, "Some Ideal Secret Sharing Schemes," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 434 LNCS, pp. 468–475, 1990, doi: 10.1007/3-540-46885-4_45.

[33]. J. Sen, "Homomorphic Encryption — Theory and Application," *Theory and Practice of Cryptography and Network Security Protocols and Technologies*, Jul. 2013, doi: 10.5772/56687.

[34]. D. Boneh *et al.*, "Threshold Cryptosystems from Threshold Fully Homomorphic Encryption," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 10991 LNCS, pp. 565–596, 2018, doi: 10.1007/978-3-319-96884-1_19.

[35]. J. Dhokrat, N. Pulgam, T. Maktum, and V. Mane, "A Framework for Privacy-Preserving Multiparty Computation with Homomorphic Encryption and Zero-Knowledge Proofs," 2024, doi: 10.31449/inf.v48i21.6562.

[36]. J. Groth, "Minimizing Non-interactive Zero-Knowledge Proofs Using Fully Homomorphic Encryption," *Cryptology ePrint Archive*, 2011, Accessed: Apr. 03, 2025. [Online]. Available: https://eprint.iacr.org/2011/012

[37]. T. Sharma, "E-Voting using Homomorphic Encryption Scheme," *Int J Comput Appl*, vol. 141, no. 13, pp. 14–16, May 2016, doi: 10.5120/IJCA2016909652.

[38]. M. Hirt and K. Sako, "Efficient Receipt-Free Voting Based on Homomorphic Encryption," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 1807, pp. 539–556, 2000, doi: 10.1007/3-540-45539-6_38.

[39]. I. Jabbar and S. N. Alsaad, "Design and Implementation of Secure Remote e-Voting System Using Homomorphic Encryption," *International Journal of Network Security*, vol. 19, no. 5, pp. 694–703, Sep. 2017, doi: 10.6633/IJNS.201709.19(5).06.

[40]. A. A. Abu Aziz, H. N.Qunoo, and A. A. Abu Samra, "Using Homomorphic Cryptographic Solutions on E-voting Systems," *International Journal of Computer Network and Information Security*, vol. 10, no. 1, pp. 44–59, Jan. 2018, doi: 10.5815/IJCNIS.2018.01.06.

[41]. B. Liang, G. Banegas, and A. Mitrokotsa, "Statically aggregate verifiable random functions and application to e-lottery," *Cryptography*, vol. 4, no. 4, pp. 1–20, Dec. 2020, doi: 10.3390/CRYPTOGRAPHY4040037.

[42]. P. A. Fouque, G. Poupard, and J. Stern, "Sharing Decryption in the Context of Voting or Lotteries," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 1962, pp. 90–104, 2001, doi: 10.1007/3-540-45472-1_7.

[43]. K. Munjal and R. Bhatia, "A systematic review of homomorphic encryption and its contributions in healthcare industry," *Complex & Intelligent Systems*, vol. 9, no. 4, p. 1, Aug. 2022, doi: 10.1007/S40747-022-00756-Z.

[44]. H. S. Trivedi and S. J. Patel, "Homomorphic cryptosystem-based secure data processing model for edge-assisted IoT healthcare systems," *Internet of Things (Netherlands)*, vol. 22, Jul. 2023, doi: 10.1016/J.IOT.2023.100693.

[45]. B. Pulido-Gaytan *et al.*, "Privacy-preserving neural networks with Homomorphic encryption: Challenges and opportunities," *Peer Peer Netw Appl*, vol. 14, no. 3, pp. 1666–1691, May 2021, doi: 10.1007/S12083-021-01076-8.

[46]. V. Bansal, "Survey on Homomorphic Encryption," *2021 5th International Conference on Information Systems and Computer Networks, ISCON 2021*, 2021, doi: 10.1109/ISCON52037.2021.9702486.

[47]. S. Ullah *et al.*, "Homomorphic Encryption Applications for IoT and Light-Weighted Environments: A Review," *IEEE Internet Things J*, 2024, doi: 10.1109/JIOT.2024.3472029.

[48]. Q. Xie *et al.*, "Efficiency optimization techniques in privacy-preserving federated learning with homomorphic encryption: A brief survey," *IEEE Internet Things J*, vol. 11, no. 14, 2024, doi: 10.1109/JIOT.2024.3382875.

[49]. H. Yang, S. Shen, W. Dai, L. Zhou, Z. Liu, and Y. Zhao, "Phantom: A CUDA-Accelerated Word-Wise Homomorphic Encryption Library," *IEEE Trans Dependable Secure Comput*, 2024, doi: 10.1109/TDSC.2024.3363900.

[50]. H. Yang, S. Shen, S. Jiang, L. Zhou, W. Dai, and Y. Zhao, "XNET: A Real-Time Unified Secure Inference Framework Using Homomorphic Encryption," *IACR Cryptol. ePrint Arch.*, Jan. 2023.

