THE ASIAN BULLETIN OF BIG DATA MANAGMENT
Vol. 6. Issue 1 (2026)
https://doi.org/10.62019/jj2yek34

Homomorphic Encryption Techniques: Advances, Challenges, and Future Directions in Distributed Computing Security

Muhammad Hussain, Shafi Ullah, Muhammad Imran Ghafoor, Muhammad Sohaib Roomi, Muhammad Yusuf, Shanila Azhar

Chronicle

Article history
Received: Feb 12, 2026
Received in the **revised format:** Feb 28, 2026
Accepted: March 15, 2026
Available online: March 30 2026

Muhammad Hussain & Muhammad Yusuf are currently affiliated with the Department of Computer Science, Balochistan University of Information Technology, Engineering and Management Sciences (BUITEMS), Quetta, Pakistan.
Email: hussainjan781@gmail.com
Email: yousafyousafzai111@gmail.com

Shafi Ullah & Shanila Azhar are currently affiliated with the Department of Computer Engineering, Balochistan University of Information Technology, Engineering and Management Sciences (BUITEMS), Quetta, Pakistan.
Email: Shafi.ullah@buitms.edu.pk
Email: Shanila.azhar@buitms.edu.pk

Muhammad Imran Ghafoor & Muhammad Sohaib Roomi are currently affiliated with the Department of Engineering & Technology, Superior University, Lahore Pakistan .
Email: engr.imranbhatti09@ieee.org
Email: sohaib4039@gmail.com

Corresponding Author*

Abstract

The emergence of Big Data and cloud computing have increased the need to have a strong cryptographic system capable of securing data throughout the processing process and not just at rest or in transit. One way to meet this need is through homomorphic encryption (HE), which allows arbitrary computations to be performed on ciphertexts and ensures that the result obtained on the plaintexts is semantically equivalent. This essay describes and revisits in detail the homomorphic encryption methods: multiplicative, XOR-based (GoldwasserMicali), additive (Paillier), and fully homomorphic encryption (FHE); focusing on their mathematical basis, real-world applications and real-world uses in healthcare, electronic voting, private information retrieval, Big Data analytics, drone security and privacy-preserving machine learning. We also analyse the security of critical infrastructure systems to intrusion threats and the complement of HE to the advanced intrusion detection frameworks. The major issues of computational overhead, noise growth and key-size expansion, and limited expressiveness of Somewhat Homomorphic Encryption (SHE) are examined, and the recent progress in Learning with Errors (LWE)- and Ring-LWE (RLWE)-based FHE algorithms are outlined. Open research directions and convergence of HE with federated learning, blockchain and IoT security are also described in the paper. This work contains 95 cross-disciplinary references to foundational cryptography and up-to-date applied systems, and has become a point of entry and a reference to those researchers constructing privacy-preserving data-processing pipelines.

Keywords: Homomorphic encryption; fully homomorphic encryption; Big Data security; cloud computing; Paillier cryptosystem; RSA; ElGamal; Goldwasser–Micali; Learning with Errors; Ring-LWE; privacy-preserving computation; intrusion detection systems; IoT security.

INTRODUCTION

The development of the Big Data technology and the enhancement of cloud computing created some severe security and privacy issues [1]–[4]. Large amounts of sensitive data: social networks, smartphones, IoT devices, healthcare records, industrial sensors, etc. have become the regular practice today to be processed and stored on shared cloud infrastructure. Traditional encryption models secure data at rest and in transit, but cloud computing may need to perform calculations on that data, which means that it has to be decrypted temporarily. This leaves a weak point that can be tapped by the enemies [5]–[8]. Homomorphic encryption (HE) fills this void by being a cryptographic object that enables computations to be calculated directly on ciphertexts [5], [15]–[17]. The decryption of the resulting ciphertext gives the same result as would have been obtained by the same operations applied to the original plaintext- a property of the mathematical structure of the underlying algebraic

system. This feature changes the way organisations can delegate computing to unreliable cloud providers without the loss of data confidentiality [24], [25]. There has been a substantial amount of literature dedicated to HE in the past 20 years, which has resulted in the first fully homomorphic encryption (FHE) scheme by Gentry in 2009 [15], [16]. It has since been developed into several generations of schemes, such as BV [14], BGV [13], and more more recent CKKS-style approximate HE, which eliminates the former limits on the noise growth, key size and computation latency [11]–[14], [26]–[29]. To complement these, an expansion of the application space beyond simple data outsourcing to privacy-sensitive machine learning [40], [41], biometric authentication [35], genomic analytics [31]-[32], and electronic voting [36], [37] occurs.

This paper makes the following contributions: (i) a structured taxonomy and comparative analysis of multiplicative, additive, XOR, and fully homomorphic encryption schemes with mathematical treatment; (ii) an updated survey of application domains with references to recent implementations; (iii) an examination of the HE–IoT/Big Data convergence; and (iv) a discussion of open challenges and future research directions, including post-quantum security, hardware acceleration, and integration with federated and privacy-preserving machine learning pipelines [49]–[60], [79], [85].

The rest of this paper is organized as follows. Section II reviews Big Data and its security landscape. Section III discusses the scope and techniques for Big Data security. Section IV provides the formal model of HE. Section V presents the four major HE types. Section VI surveys critical infrastructure security. Section VII maps HE to application domains. Section VIII traces the evolution of FHE. Section IX concludes with future directions.

Big Data and Its Security

Big Data is the colossal amount of information gathered on the Internet, social networks, smartphones, and IoT devices [4], [8], [60]. Big Data is huge, multifaceted and can be structured, semi structured or unstructured [23]. It is characterized by five Vs as follows:

- Volume: the size of the records, transactions and tables in a dataset expressed in terabytes or petabytes [4].
- Velocity: the rate of real-time or near-real-time at which data is produced and processed [4].
- Variety: data in diverse forms: structured, unstructured, semi-structured, textual, spatial or temporal [4].
- Value: the statistical inferences, events, correlations and hypotheses which can be drawn out of data [4].
- Veracity: trustworthiness, authenticity, accountability and availability of data produced [4], [60].

One of the most critical issues with Big Data is its security. Scholars claim that the occurrence of security incidents in Big Data systems can be mostly explained by the utilization of interconnected computers and heterogeneous storage models all of which increase the attack surface [6], [7]. The fact that cloud-hosted Big Data

infrastructure is dynamically changing further complicates the implementation of access controls, key management and audit logging [2], [3], [65].

CHALLENGES AND CONCERNS IN BIG DATA TECHNOLOGY

The main issues regarding the Big Data technology are [1]-[8]: (i) Big Data applications security; (ii) incident response plans; (iii) data leaks and prevention; (iv) physical and personal security; (v) identity and access control; (vi) segregation and protection; (vii) legacy systems integration; (vii) Big Data applications and legacy systems ambiguity; and (viii)-(xiii) migration difficulties

The answers to these dilemmas are based on the specifications and functions of the platforms. An example is that bandwidth switching and reduction of traffic in cloud environments can address security threats that are caused by latency [8]. Lazy preclusion of encroachment in distributed Big Data file systems (e.g., HDFS) reduces privacy leaks in the transmission process [51], [52]. Also gaining extreme popularity as a threat vector in cloud-hosted Big Data systems is social engineering, which involves using human factors to steal credentials [57], [58].

SCOPE OF BIG DATA SECURITY

Big Data security has attracted intense research interest, with most efforts focused on securing cloud-based deployments. Future Big Data applications must comply with the CIA triad—Confidentiality, Integrity, and Availability—to enhance both security posture and user acceptance [2], [3], [17]. Growing demand for data security has driven the development of advanced encryption algorithms suited to the rapid resource-deployment model of cloud computing [8], [30].

TECHNIQUES FOR BIG DATA SECURITY

Several techniques have been developed to implement security in Big Data systems, especially in cloud computing environments:

- An encryption algorithm for Big Data based on Parallelized Disjunctive Query (PDQ) enables efficient secure search over encrypted datasets [7].
- Encryption techniques that secure Big Data and IoT systems while maintaining privacy, efficiency, and sustainability of cloud computing support infrastructure [8].
- A Homomorphic Re-encryption Scheme (HERS) protects raw data while enabling Privacy-Preserving Data Processing (PPDP). By combining RSA with a HERS layer, re-encryption is performed on the cloud side without ever exposing plaintext to the cloud provider [24].

- Data anonymization techniques—including k-anonymity, l-diversity, and t-closeness—implemented on distributed processing frameworks such as MapReduce and Apache Spark constitute a complementary line of defense [49]–[55].
- Attribute-Based Encryption (ABE), Hierarchical Identity-Based Encryption (HIBE), and Hierarchical Attribute Set-Based Encryption (HASBE) extend fine-grained access control to multi-tenant Big Data environments [46], [47].

IV. FORMAL MODEL OF HOMOMORPHIC ENCRYPTION

Consider a message space $(M, \circ)$ that is a finite semi-group or group with σ as a security parameter. A homomorphic cryptosystem on message space M is a quadruple (K, E, D, A) of probabilistic polynomial-time algorithms [5], [22]:

- **Key Generation (K):** On providing initiation parameter 1^σ, the key generation scheme produces an encryption and a decryption key pair (k_e, k_d) = k ∈ K, where K represents the key space.
- **Encryption (E):** On providing 1^σ, k_e, and an element in the message space m ∈ M, the encryption scheme produces a ciphertext c in cipher-space C: c ∈ C.

Decryption (D): The decryption scheme is deterministic. It requires 1^σ, k, c ∈ C to reproduce m ∈ M such that ∀m ∈ M if c = E(1^σ, k_e, m) then Prob[D(1^σ, k, c) ≠ m] is negligible (≤ negl(σ)).

- **Homomorphic Property (A):** A is a scheme that requires 1^σ, k, and c_1, c_2 ∈ C to produce a third ciphertext c_3 ∈ C such that ∀m_1, m_2 ∈ M: $m_3 = m_1 \circ m_2$, c_1 = E(1^σ, k_e, m_1), c_2 = E(1^σ, k_e, m_2) ⟹ D(1^σ, k, A(1^σ, k, c_1, c_2)) = m_3.

An encryption scheme is considered Partially Homomorphic (PHE) if it supports either addition or multiplication (but not both) for an unlimited number of operations. A Somewhat Homomorphic Encryption (SHE) scheme supports both operations but for a bounded depth of computation. FHE lifts this restriction by supporting arbitrary circuits of unbounded depth [15], [16], [28].

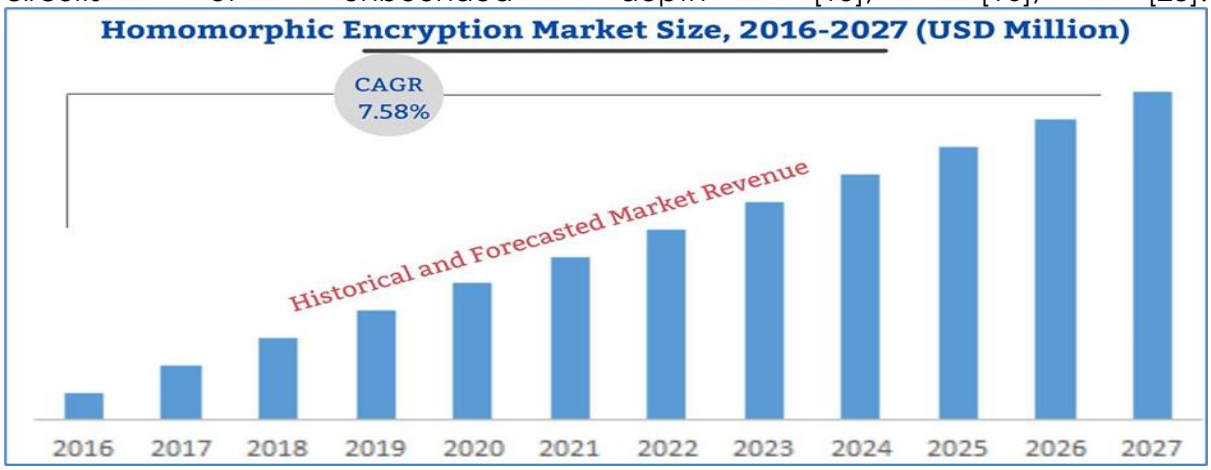

Figure 1.
Homomorphic Encryption Method

TYPES OF HOMOMORPHIC ENCRYPTION TECHNIQUES

There are four major types of homomorphic encryption: multiplicative homomorphic encryption, XOR homomorphic encryption, additive homomorphic encryption, and fully homomorphic encryption (FHE) [5], [17], [22], [28]. Each is discussed below.

Multiplicative Homomorphic Encryption

Multiplicative homomorphic encryption is exemplified by the RSA public-key cryptosystem and the ElGamal cryptosystem [22], [28]. RSA encryption operates in four steps:

1. A random number generator produces a pair of large prime numbers u and v; their product N = uv forms the modulus.
2. The public key component (N, L) is shared between communicating parties; each party retains its private key.
3. The communicating parties agree on a padding scheme; the sender encrypts message S as E(S) = S^L mod N.
4. The recipient decrypts the received ciphertext using the private key to retrieve the original message.

The homomorphic property of unpadded RSA requires:

$E(S_1) \cdot E(S_2) = S_1^L \cdot S_2^L \bmod N = (S_1 S_2)^L \bmod N = E(S_1 S_2)$ (1)

RSA encryption can also be used in digitally signing data packets, thereby thwarting Man-In-The-Middle (MITM) attacks [5], [22]. Multiplicative homomorphism is also realizable through the ElGamal cryptosystem, which employs asymmetric public-key encryption. The ElGamal keys are defined over a cyclic group G of prime order d with generator T [22]:

Public key: (G, T, d, L), where $L = T^p$ (2)

Private key: $p \in \{1, 2, \ldots, d-1\}$ (3)

XOR Homomorphic Encryption (Goldwasser–Micali)

XOR homomorphic encryption is based on the Goldwasser–Micali (GM) scheme, an asymmetric encryption technique grounded in quadratic residuosity assumptions [5], [28]. Given two large prime numbers a and b, the scheme computes the composite modulus $V = a \cdot b$. The non-residue t and the key pair are then generated as [22]:

$t^{((a-1)/2)} \equiv -1 \pmod a \implies t$ *is a quadratic non-residue mod* V (4)

Public key: (t, V) (5)

Private key: (a, b) (6)

Goldwasser–Micali achieves XOR (bitwise) homomorphism through multiplication of ciphertexts modulo N. Its probabilistic nature provides semantic security, but it is limited to single-bit encryption, making ciphertext expansion a practical concern [5], [28].

Additive Homomorphic Encryption (Paillier)

Additive homomorphic encryption is based on the Paillier cryptosystem, an asymmetric probabilistic scheme [5], [25], [28]. Key generation proceeds by selecting two large primes a and b such that GCD(ab, (a−1)(b−1)) = 1, computing L = ab and λ = LCM(a−1, b−1), selecting a generator $T \in Z^*_{L^2}$, and deriving a parameter μ as follows [22], [28]:

Modulus: $L = ab$ (7)

LCM parameter: $\lambda = LCM(a-1, b-1)$ (8)

Generator condition: $(T^\lambda \bmod L^2 - 1) / L$ *must be invertible mod* L (9)

μ: $\mu = (\mathcal{L}(T^\lambda \bmod L^2))^{-1} \bmod L$, *where* $\mathcal{L}(x) = (x-1)/L$ (10)

Public key: (L, T) (11)

Private key: (λ, μ) (12)

The additive homomorphic property of Paillier means that $D(E(m_1) \cdot E(m_2) \bmod L^2) = m_1 + m_2 \bmod L$. This makes Paillier ideal for secure aggregation, electronic voting tallying, and privacy-preserving statistics [36], [37], [44].

Fully Homomorphic Encryption (FHE)

The strongest type of homomorphic encryption is FHE, which allows unlimited multiplications and additions of ciphertexts [15], [16]. It can be built off of LWE and

RLWE assumptions. According to the LWE problem, it is computationally infeasible to recover a secret s, given a random (matrix) A ∈ Z_q n x m and a randomized output b = As + e (mod q) where e is a small error vector [39]. The FHE of RWWE works in the polynomial rings, and therefore, with the help of batching and NTT-accelerated multiplication of polynomials, it can achieve large performance gains [11], [13], [14], and [39]. Noise management is the main issue in FHE. Every ciphertext has a small error that is added during encryption; homomorphic operations add this noise, and, when it surpasses a certain threshold, the ciphertext has been rendered undecryptable [15]. The bootstrapping method used by Gentry reinitiates the noise by homomorphically evaluating the decryption circuit but it is costly. Later (BV, BGV, CKKS) modulus switching, key switching, and scale-invariant schemes were added to increase the depth of the computable circuit, without the need to bootstrap it, [13], [14], [26], [27].

VI. SECURITY OF CRITICAL INFRASTRUCTURE ENCRYPTION

Critical infrastructure such as smart grids, connected vehicles, industrial control systems, and healthcare cyber-physical systems have higher threat profiles that even conventional encryption is inadequate to handle [19], [20]. Recently, researchers have investigated domain-specific Intrusion Detection Systems (IDS) that are specific to the following domains:

- A secure cloud-based service availability framework with built-in mechanism of intrusion detection against security violation of an automated system of smart vehicles connected [19].
- Adaptively Supervised and Clustered Hybrid IDS (ASCH-IDS) of wirelessly connected sensor clusters that is used to monitor critical infrastructures based on Receiver Operating Characteristic (ROC) analysis [20].
- Machine-to-Machine (M2M) and IoT devices: HE-based authentication protocols to avoid exposing credentials when using over-the-air applications [56], [61].
- Access control with the help of blockchain and HE to smart homes and industrial IoT settings [59], [76].

The integration of HE and IDS models suggests a promising research opportunity: it is possible to detect anomalies directly in encrypted traffic logs, without having to decrypt sensitive information to perform the analysis [63], [64], [80].

VII. HOMOMORPHic encryption has applications in the following ways.

The subsections below give a survey of the main areas of application of HE, based on the literature on the topic, as well as recent applications.

A. Drone Security

An authenticated encryption structure that is linearly homomorphic is used to authenticate and secure ground-controlled multirotor drones by verifying the integrity of flight-control commands without decryption [9], [21]. The same approach can be applied to autonomous vehicle fleets in which command-and-control integrity is a safety-critical aspect [81].

B. Decreasing Storage Requirements.

A hybrid public-key encryption scheme that integrates Somewhat Homomorphic Encryption (SHE) and lightweight symmetric-key primitives can significantly decrease

the ciphertext expansion [21]. These schemes are of special interest to resource-constrained IoT endpoints and edge nodes [60], [61], [80].

C. Outsourcing Computer and Storage.

HE helps organizations that handle Big Data to delegate computation and storage to cloud providers without presenting plaintext [2], [3], [7]. Functional encryption algorithms produce public keys that enable authorized parties to analyze certain functions on ciphertexts, enabling fine-grained access control to analytics-as-a-service models [30], [51], [84].

D. Private Information Retrieval (PIR)

PIR protocols and private query processing use HE. Users exploit HE to present queries to search engines or databases, in such a way that the server does not know anything about the query other than its answer [25], [29]. On-line decryption can be supported by streaming versions in network-coding applications [24].

E. Two Party and Multi-Party Computations.

HE can securely execute two-party computation in which parties that distrust each other collectively and privately compute a function without knowing each other's respective inputs (e.g., Alice has input A, Bob has input B; either does not know the other's private value) [13], [45]. Examples of commercial applications include supply-chain collaboration and privacy-preserving joint analytics [29].

F. Zero-Knowledge Proof Protocols.

The homomorphic nature of HE can be used to build zero-knowledge proof systems, where a prover can prove its knowledge of a secret without disclosing it [9], [13]. This will be especially useful in privacy-preserving credential systems and identity management via blockchain [76].

G. Big Data and Cloud computing.

The encryption algorithms are functional to aid analytics in Big Data by building public keys that allow calculation of particular aggregate statistics of encrypted data [30]. Cryptosystems based on OkamotoUchiyama, Paillier, and ElGamal, and entirely homomorphic have been experimented in pipelines of cloud-based Big Data [27], [29], [84]. Healthcare is a well-known application: HE can be used to run privacy-preserving queries on Electronic Medical Records (EMR) and analyze patient genomics securely, without breaking the HIPAA or GDPR limitations [31] -35], [65], [72].

H. Electronic Voting

Homomorphic encryption is a backbone to verifiable electronic voting. Paillier-based tallying schemes enable the summing of encrypted ballots without decryption of votes, maintaining ballot secrecy and enabling open checking [36]-[38], [48]. Logics such as Applied Pi Calculus have been actively studied as a method of formal verification of such systems [36].

I. Privacy-Protecting Machine Learning.

Machine learning has been incorporated into HE to train and make inferences on encrypted data [40], [41]. Privacy-preserving distributed control systems can be achieved via reinforcement learning using HE-protected reward signals [38].

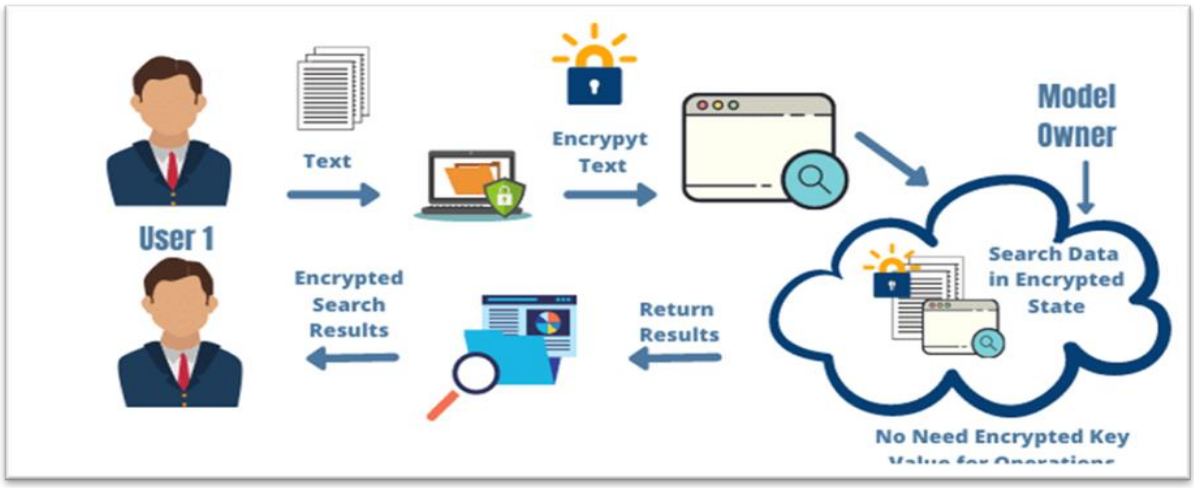

Figure 2.
Homomorphic Encryption Applications

Federated learning with HE is such that model gradients exchanged between clients and aggregators are confidential [79], [83], [85].

BACKGROUND AND EVOLUTION OF FULLY HOMOMORPHIC ENCRYPTION

Through a constructive scheme was only elusive until 2009 when Gentry constructed the first plausible FHE construction using ideal lattices [15], [16]. The first practical (although incomplete) implementation was then done by Smart and Vercauteren [28]. The evolution of FHE since the first plan proposed by Gentry to contemporary versions is the following milestones:

- Formulation of an FHE scheme using the Approximate Greatest Common Divisor (AGCD) problem on integers, making the scheme run on integer arithmetic [18].

- Compression of the size of the public-key (e.g. 2.25 GB with a lattice dimension 32,768) to manageable sizes by squashing and modulus reduction [13], [29].

- Module switching (key switching) was introduced to further decrease the size of the public-key and the cost of bootstrapping [13].

- Introduction of batching with the Chinese Reminder Theorem (CRT) and addition of permutation operations, allowing SIMD-like parallel processing of multiple plaintext slots [11], [13].

- Weakening of the LWE assumption to AGCD, and making integer based FHE based on a lattice-based hardness foundation [18].

- Ring-LWE (RLWE)-based FHE with key and modulus switching development, and scale-invariant (BGV-style) and approximate arithmetic (CKKS) schemes to minimize noise growth per multiplication [11], [13], [14].

- Hardware acceleration of HE with Karatsuba multiplication and Number Theoretic Transform (NTT), which can practically support homomorphic evaluation speeds [26].

The sub-exponential time attacks on lattice problems that LWE and RLWE are based on, the residual noise accumulation in deep circuits, and poor standardization are currently the main bottlenecks. The NIST Post-Quantum Cryptography standardization program has led to a fresh surge of interest in lattice-hardness assumptions, offering a roadmap to standardized post-quantum HE [11], [39].

In 2020 the global homomorphic encryption market has increased significantly, and verticals in healthcare, finance, and government procurement have driven commercial deployment [8], [23]. HE libraries (open-source) Microsoft SEAL, HEAAN, HElib, OpenFHE have reduced the barrier to entry, so that practitioners can integrate HE into production pipelines without requiring in-depth cryptographic knowledge [27], [40], [41].

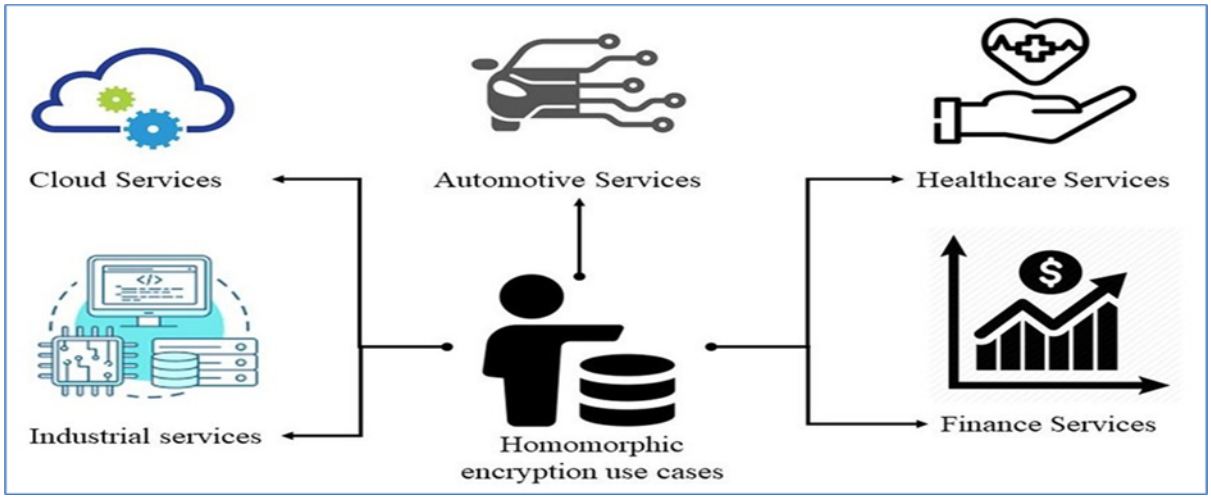

Figure 3.
Homomorphic Encryption Market Size

OPEN CHALLENGES AND FUTURE DIRECTIONS

Computational Overhead: FHE is many times slower than plaintext operations. The most significant research directions are hardware co-design (FPGA/ASIC accelerators) and algorithmic improvements (batching, approximate arithmetic) [26], [27], [84].

Noise Management and Bootstrapping: Bootstrapping is the most costly operation in FHE with an approximate cost of 8090 percent of total computation time. Approximate bootstrapping schemes (CKKS) provide a partial answer to applications that can withstand small numerical errors [13], [14].

Post-Quantum Security: LWE/RLWE-based HE is a construction with post-quantum security by default, but to select the parameter carefully to resist known lattice reduction attacks (BKZ, sieging) it is necessary to carefully consider the recommendations of NIST [11], [39].

Standardization: HE APIs and parameter sets are not standardized, which hinders interoperability. The HomomorphicEncryption.org Security Standard and continuing NIST work is a good step towards a single specification [5], [28]. HE in Federated and Distributed Learning: HE in federated learning enhances gradient privacy without a trusted aggregator, and combines HE with secure aggregation. Effective protocols to this application are still an open research issue [38], [79], [83], [85].

HE IoT and Edge computing: IoT devices have limited computational capabilities that require lightweight versions of HE. Leveled HE with compact parameters and NTRU-based schemes are candidate constructions [56], [60], [61], [80]. On-Chain computation: HE can be used in Smart contracts to achieve confidential on-chain computation. This synergy makes it possible to have privacy-preserving decentralized applications to vote, healthcare data markets, and supply-chain analytics [59], [76].

CONCLUSION

Big Data and cloud computing represent revolutionary technologies in any industry, and security issues have kept the widespread adoption of these solutions to sensitive workloads at bay. The paper has provided an overview of the history of homomorphic encryption, starting with its original cryptographic design and its four main types of homomorphic encryption (multiplicative, XOR, additive, and fully homomorphic) and mapped these schemes to a wide variety of application fields such as healthcare, electronic voting, PIR, federated learning, IoT security, and critical infrastructure protection. Notable accomplishments are: (i) FHE using LWE/RLWE is the most general privacy-preserving computational model and post-quantum secure, but still computationally expensive; (ii) Paillier and ElGamal PHE are practical in terms of additive and multiplicative workloads respectively, and attractive to voting, aggregation, and limited analytics; (iii) hardware acceleration and approximate arithmetic (CKKS

The further studies need to be oriented at standardizing the HE parameters, minimizing the bootstrapping latency, and production of lightweight HE versions to be applied in edge/iot applications. The emerging ecosystem of open-source HE libraries and the NIST post-quantum cryptography standard give a solid base to the future generation of privacy-preserving Big Data systems.

DECLARATIONS

Acknowledgement: We appreciate the generous support from all the contributors to the research and their different affiliations.
Funding: No funding body in the public, private, or nonprofit sectors provided a particular grant for this research.
Availability of data and material: In the approach, the data sources for the variables are stated.
Authors' contributions: Each author participated equally in the creation of this work.
Conflicts of Interest: The authors declare no conflict of interest.
Consent to Participate: Yes
Consent for publication and Ethical approval: Because this study does not include human or animal data, ethical approval is not required for publication. All authors have given their consent.

REFERENCES

A. Acar, H. Aksu, A. S. Uluagac, and M. Conti, "A survey on homomorphic encryption schemes," ACM Comput. Surv., vol. 51, no. 4, pp. 1–35, 2018.
A. Jahangeer, S. U. Bazai, S. Aslam, S. Marjan, M. Anas, and S. H. Hashemi, "A review on the security of IoT networks: From network layer's perspective," IEEE Access, vol. 11, pp. 71073–71087, 2023.
A. Rohilla, "Homomorphic cryptosystem," Int. J. Comput. Netw. Inf. Secur., pp. 44–51, 2017.
C. Gentry, "A fully homomorphic encryption scheme," Ph.D. dissertation, Stanford University, 2009.

C. Gentry, "Fully homomorphic encryption using ideal lattices," in Proc. 41st Annu. ACM Symp. Theory Comput. (STOC), 2009.
C. Hazay et al., "Efficient RSA key generation and threshold paillier in the two-party setting," Topics in Cryptology CT-RSA, 2018.
D. Mittal, D. Kaur, and A. Aggarwal, "Secure data mining in cloud using homomorphic encryption," in IEEE Int. Conf. Cloud Computing Emerging Markets (CCEM), 2014.
D. S. Galav, S. M. Ghosh, and P. Shrivastav, "Data confidentiality for secure cloud computing through homomorphic encryption," Int. J. Adv. Res. Comput. Sci., vol. 6, no. 1, pp. 56–97, 2015.
D. Singh, Dayanand, and A. Arya, "Security challenges in Big Data," Int. J. Comput. Sci. Eng., vol. 6, no. 7, pp. 981–985, 2018.
D. Wang, B. Guo, Y. Shen, S. J. Cheng, and Y. H. Lin, "A faster fully homomorphic encryption scheme in big data," in IEEE 2nd Int. Conf. Big Data Analysis (ICBDA), 2017, pp. 345–349.
G. S. Cetin et al., "Private queries on encrypted genomic data," BMC Med. Genomics, vol. 10, 2017.
I. Tabassum, S. U. Bazai, Z. Zaland, S. Marjan, M. Z. Khan, and M. I. Ghafoor, "Cyber Security's Silver Bullet—A Systematic Literature Review of AI-Powered Security," in 3rd Int. Informatics and Software Engineering Conf. (IISEC), IEEE, 2022, pp. 1–7.
M. Fahsi, S. M. Benslimane, and A. Rahmani, "A framework for homomorphic, private information retrieval protocols in the cloud," Int. J. Modern Educ. Comput. Sci., vol. 7, no. 5, pp. 16–23, 2015.
M. M. Potey, C. Dhote, and D. H. Sharma, "Homomorphic encryption for security of cloud data," Procedia Comput. Sci., 2016.
M. V. Dijk and C. Gentry, "Fully homomorphic encryption over the integers," in Advances in Cryptology (EUROCRYPT), 2010.
N. Chakraborty and G. K. Patra, "Functional encryption for secured Big Data analytics," Int. J. Comput. Appl., vol. 107, no. 16, pp. 19–22, 2014.
N. G. Tsoutsos and M. Maniatakos, "Efficient detection for malicious and random errors in additive encrypted computation," IEEE Trans. Comput., vol. 67, no. 1, pp. 16–31, 2018.
O. Regev, "The learning with errors problem," in Proc. Annu. IEEE Conf. Computational Complexity, 2010.
P. R. Kumar, P. H. Raj, and P. Jelciana, "Exploring data security issues and solutions in cloud computing," Procedia Comput. Sci., vol. 125, pp. 691–697, 2018.
P. V. Parmar, S. B. Padhar, S. N. Patel, N. I. Bhatt, and R. H. Jhaveri, "Survey of various homomorphic encryption algorithms and schemes," Int. J. Comput. Appl., vol. 91, no. 8, pp. 26–32, 2014.
M. Rahman, (2023). *Identifying Evidence-Based Strategies to Strengthen the Ability of Social Enterprises to Scale Health Impact in Low-and Middle-Income Countries (Doctoral dissertation, Doctoral dissertation, Duke University)* (Doctoral dissertation, Doctoral dissertation, Duke University).
R. Ullah, S. U. Bazai, U. Aslam, and S. A. A. Shah, "Utilizing blockchain technology to enhance smart home security and privacy," in Proc. Int. Conf. Information Technology and Applications, Springer, 2023, pp. 491–498.
S. Noor, S. U. Bazai, M. I. Ghafoor, S. Marjan, S. Akram, and F. Ali, "Generative adversarial networks for anomaly detection: A systematic literature review," in 4th Int. Conf. iCoMET, IEEE, 2023, pp. 1–6.
S. Otoum, B. Kantarci, and H. Mouftah, "Adaptively supervised and intrusion-aware data aggregation for wireless sensor clusters in critical infrastructures," in IEEE Int. Conf. Commun. (ICC), 2018.
S. Tahir, L. Steponkus, S. Ruj, M. Rajarajan, and A. Sajjad, "A parallelized disjunctive query based searchable encryption scheme for big data," Future Gener. Comput. Syst., 2018.
S. Tareen, S. U. Bazai et al., "Phishing and intrusion attacks: An overview of classification mechanisms," in 3rd Int. Informatics and Software Engineering Conf. (IISEC), IEEE, 2022, pp. 1–5.
S. U. Bazai and J. Jang-Jaccard, "In-memory data anonymization using scalable and high performance RDD design," Electronics, vol. 9, no. 10, p. 1732, 2020.

S. U. Bazai and J. Jang-Jaccard, "SparkDA: RDD-based high-performance data anonymization technique for Spark platform," in Int. Conf. Network and System Security, 2019, pp. 646–662.

S. U. Bazai, J. Jang-Jaccard, and H. Alavizadeh, "Scalable, high-performance, and generalized subtree data anonymization approach for Apache Spark," Electronics, vol. 10, no. 5, p. 589, 2021.

S. U. Bazai, J. Jang-Jaccard, and X. Zhang, "A privacy preserving platform for MapReduce," in Int. Conf. Applications and Techniques in Information Security, 2017, pp. 88–99.

S. U. Bazai, M. I. Ghafoor, U. A. Bhatti, M. A. Siddiqui et al., "Efficient autonomous navigation in dynamic indoor environment using VLP-16 LiDAR and sensor fusion with TD3," in Int. Conf. FIT, IEEE, 2025, pp. 1–6.

S. Yakoubov, V. Gadepally, N. Schear, E. Shen, and A. Yerukhimovich, "A survey of cryptographic approaches to securing big-data analytics in the cloud," in IEEE High Performance Extreme Computing Conf. (HPEC), 2014.

T. B. Patil, G. K. Patnaik, and A. T. Bhole, "Big data privacy using fully homomorphic non-deterministic encryption," in Proc. 7th IEEE Int. Advanced Computing Conf. (IACC), 2017, pp. 138–143.

T. Dugan and X. Zou, "Privacy-preserving evaluation techniques and their application in genetic tests," Smart Health, vol. 1–2, pp. 2–17, 2017.

T. Graepel, K. Lauter, and M. Naehrig, "ML confidential: Machine learning on encrypted data," in Information Security and Cryptology (ICISC), 2012.

V. Migliore et al., "A high-speed accelerator for homomorphic encryption using the Karatsuba algorithm," ACM Trans. Embed. Comput. Syst., vol. 16, no. 5s, 2017.

V. Thayananthan and A. Albeshri, "Big data security issues based on quantum cryptography and privacy with authentication for mobile data center," Procedia Comput. Sci., vol. 50, pp. 149–156, 2015.

W. Ding, Z. Yan, and R. H. Deng, "Encrypted data processing with homomorphic re-encryption," Inf. Sci., vol. 409–410, pp. 35–55, 2017.

X. Wang and Z. Zhang, "Data division scheme based on homomorphic encryption in WSNs for health care," J. Med. Syst., vol. 39, no. 12, 2015.

X. Yi, A. Bouguettaya, D. Georgakopoulos, A. Song, and J. Willemson, "Privacy protection for wireless medical sensor data," IEEE Trans. Depend. Secure Comput., vol. 13, no. 3, pp. 369–380, 2016.

Z. Brakerski and V. Vaikuntanathan, "Efficient fully homomorphic encryption from (standard) LWE," in Proc. Annu. IEEE Symp. Foundations Comput. Sci. (FOCS), 2011.

Z. Brakerski, C. Gentry, and V. Vaikuntanathan, "(Leveled) Fully homomorphic encryption without bootstrapping," ACM Trans. Comput. Theory, vol. 6, no. 3, pp. 1–36, 2014.