International Journal of Science and Research (IJSR)
ISSN: 2319-7064
SJIF (2022): 7.942

Introduction: Cloud Storage Security and Homomorphic Encryption in Cloud Computing

Pushpjeet Cholkar[1], Dr. Margi Patel[2]

[1]M. Tech Scholar, CSE, Indore Institute of Science and Technology, Indore, Madhya Pradesh, India

[2]Guide, Indore Institute of Science and Technology, Indore, Madhya Pradesh, India

Abstract: ***Cloud computing, is one of the key hybrids of Internet-based computing that allows on-demand sharing of resources, including processing and data storage to globally separated machines and other smart electronic devices. This coherent model supports ubiquitous, on-demand access to a shared pool of configurable computing resources. Cloud data storage solutions constitute a significant application area in the cloud computing domain. In this work, cloud storage security is area of focus. These security measures are configured to protect cloud storage, support regulatory compliance and protect customers' privacy as well as setting authentication rules for individual users and devices. Additionally, the overview of homomorphic encryption technique is used for cloud storage security purpose.***

Keywords: Homomorphic encryption, Cloud security, Cloud Computing and Cloud Storage, Energy efficiency and quality of service (QoS)

1. Introduction

Cloud computing is the delivery of hosted services, including software, hardware, and storage, over the Internet. The benefits of rapid deployment, flexibility, low up-front costs, and scalability, have made cloud computing virtually universal among organizations of all sizes, often as part of a hybrid/multi-cloud infrastructure architecture. Cloud security refers to the technologies, policies, controls, and services that protect cloud data, applications, and infrastructure from threats [1].

Cloud security, also known as cloud computing security, consists of a set of policies, controls, procedures and technologies that work together to protect cloud-based systems, data, and infrastructure. These security measures are configured to protect cloud data, support regulatory compliance and protect customers' privacy as well as setting authentication rules for individual users and devices [2,3]. From authenticating access to filtering traffic, cloud security can be configured to the exact needs of the business. And because these rules can be configured and managed in one place, administration overheads are reduced and IT teams empowered to focus on other areas of the business.

In ancient Greek, the term (homos) was used to mean “same,” while (morphe) was used for “shape” (Liddell and Scott 1896). Then the term homomorphism was coined and used in different areas. In abstract algebra, homomorphism is defined as a map preserving all the algebraic structures between the domain and range of an algebraic set (Malik et al. 2007). The map is simply a function, i.e., an operation that takes the inputs from the set of domains and outputs an element in the range (e.g., addition, multiplication). In the cryptography field, homomorphism is used as an encryption type. Homomorphic Encryption (HE) is a kind of encryption scheme that allows a third party (e.g., cloud, service provider) to perform certain computable functions on the encrypted data while preserving the features of the function and format of the encrypted data. Indeed, this homomorphic encryption corresponds to a mapping in the abstract algebra. As an example for an additively HE scheme, for sample messagesm1 andm2, one can obtain E(m1 + m2) by using E(m1) and E(m2) without knowing m1 and m2 explicitly, where E denotes the encryption function. Normally, encryption is a crucial mechanism to preserve the privacy of any sensitive information. However, the conventional encryption schemes cannot work on the encrypted data without decrypting it first. In other words, the users have to sacrifice their privacy to make use of cloud services such as file storing, sharing, and collaboration. Moreover, untrusted servers, providers, and popular cloud operators can keep physically identifying elements of users long after users end the relationship with the services (McMillan 2013). This is a major privacy concern for users. In fact, it would be perfect if there existed a scheme that would not restrict the operations to be computed on the encrypted data while it would be still encrypted. From a historical perspective in cryptology, the term homomorphism was used for the first time by Rivest et al. (1978a) in 1978 as a possible solution to the computing without decrypting problem. This given basis in Rivest et al. (1978a) has led to numerous attempts by researchers around the world to design such a homomorphic scheme with a large set of operations. In this work, the primary motivation is to survey the HE schemes focusing on the most recent improvements in this field, including partially, somewhat, and fully HE schemes.

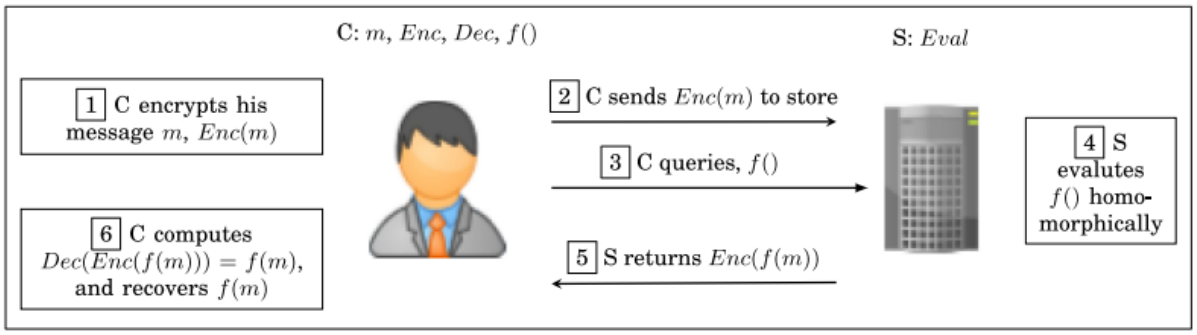

Figure 1: A simple client-server HE scenario, where C is Client and S is Server.

2. Literature Review

Wu et al.[1] introduction to cloud storage. It covers the key technologies in Cloud Computing and Cloud Storage, several different types of clouds services, and describes the advantages and challenges of Cloud Storage after the introduction of the Cloud Storage reference model.

M. Lakshmi et al.[2] presents the key technologies and virtual storage architecture in cloud. Cloud storage is more advantageous than traditional storage because of its availability, scalability, performance, portability and its functional requirements. Implementing virtualization in the cloud storage improves the scalability, availability but at the same time providing security in the virtual environment is complex. So apart from virtualization, emphasis should be given regarding security in virtual storage.

Odun-Ayo et al. [3] focused on Cloud storage. A review of Cloud storage systems, architecture, models and challenges was done. A comparison of some of the storage features offered by two popular Cloud Storage Service Providers IBM and Amazon was also done. In conclusion it is important to note that despite certain Cloud challenges particularly in terms of security and privacy, Cloud storage is still being adopted at a tremendous rate; and research works are still on-going in a bid to further push the boundaries of Cloud storage adaptation.

Lee et al.[4] proposed a solution to the double curl equation with generalized Coulomb gauge based on the vectorial representation of the magnetic vector potential. Finally, the original equation can be rewritten in a generalized form and solved in a more natural and accurate way using finite-element method.

Yang et al.[5] propose an efficient fine-grained outsourced data deletion scheme based on invertible Bloom filter, which can also achieve public and private verifiability of the storage and deletion results. If the cloud server does not honestly maintain/delete the data and generate corresponding evidences, users can easily detect the cloud server's malicious behaviors with an overwhelming probability. Meanwhile, in data deletion and deletion result verification processes, the computational complexity is independent of the number of outsourced data blocks, which makes the proposed scheme be suitable for large-scale data deletion scenario. Moreover, we provide the detailed security analysis and performance evaluation, which can respectively demonstrate the security and practicability of the proposed scheme.

Yang et al.[6] propose an efficient large universe regular language searchable encryption scheme for the cloud, which is privacy-preserving and secure against the off-line keyword guessing attack (KGA). A notable highlight of the proposal over other existing schemes is that it supports the regular language encryption and deterministic finite automata (DFA) based data retrieval. The large universe construction ensures the extendability of the system, in which the symbol set does not need to be predefined. Multiple users are supported in the system, and the user could generate a DFA token using his own private key without interacting with the key generation center. Furthermore, the concrete scheme is efficient and formally proved secure in standard model. Extensive comparison and simulation show that this scheme has function and performance superior than other schemes.

Yu et al.[7] give the security analysis of a PDP scheme. We show this scheme does not guarantee the storage correction. The malicious cloud can forge a proof to pass the verification from TPA even if it has deleted the whole user's file

Wang et al. [8] proposes a novel multiuser computational offloading scheme for a fog-based scenario. Specifically, a controlling user (CU) distributes its computational tasks to multiple trusted helping users (HUs) by exploiting both nonorthogonal multiple access (NOMA) and beamforming. First, social relationships between the CU and HUs are exploited for the selection of trusted HUs. Our numerical results demonstrate that by combining the social relationships, the computational capacities, and the channel conditions, the proposed multiuser computational offloading scheme relying on social trust, NOMA, and beamforming improves the energy consumption, sum rate, and transmission latency.

Bai et al. [9] propose a multiedgechain structure that accommodates thousands of edge data and promotes on-chain data efficiency to achieve cross-chain edge data sharing for heterogeneous blockchain systems. Moreover, aiming at the profits of computing resource scheduling in the IIoE, a two-stage Stackelberg game strategy with an optimal scheduling demand and reward is provided considering the edge user's preferences and risk factors. Finally, the simulation results verify the superiority of the proposed scheme, regarding the game equilibrium, utility

optimization, and data sharing efficiency of cloud–edge collaboration.

Qiu et al. [10] propose a dynamic threshold-based access scheme for security provisioning of C-V2X computation-offloading network by considering an imperfect CSI. In this scheme, an optimized access threshold is set to update adaptively in terms of channel estimation error for balancing both the security and reliability of the offloading link. Furthermore, the proposed scheme can maximize the secrecy throughput under a connection outage constraint of the D2D-V links, with the total area spectral efficiency optimized under the security performance criterion for the offloading link. Numerical results are provided for validating the proposed theoretical analysis. A useful design insight is provided for attaining an optimal configuration of C-V2X computation-offloading network.

Tiwari et al. [11] propose an efficient online secretary-based algorithm for choosing the best suitable candidate FN for offloading the data. To show the effectiveness of Devote, we obtained the numerical results for assessing its performance, while collating it with the benchmark schemes. We analyze different performance metrics, such as service delay, economy, and user satisfaction, which show that devote incurs less service delay, as compared to other systems, while achieving user satisfaction of 88.4%.

Mostafa et al. [12] aims to maximize the number of users being served and minimize the total energy consumption subject to delay tolerance constraints. The joint computation and communication resource allocation problem is solved optimally for both non-orthogonal multiple access (NOMA) and orthogonal multiple access (OMA) schemes. The joint user pairing and fog access point assignment problem for NOMA is proved to be NP-hard. For both NOMA and OMA, heuristic and optimal algorithms based on graph matching are designed. The optimal algorithms, though of high complexity, allow NOMA and OMA to be compared at their full potential and serve as benchmarks for evaluating the heuristic algorithms. Simulation results show that NOMA significantly outperforms OMA in terms of outage probability and energy consumption, especially for tight delay tolerance constraints and large computational tasks. Simulation results also demonstrate that our proposed NOMA and OMA schemes significantly outperform the swap-enabled matching algorithm widely used in the literature.

Cao et al. [13] multiple computing paradigms are emerging, such as mobile transparent computing (TC), edge computing, and fog computing. These paradigms employ more resourceful edge devices, e.g., small-scale servers, smart phones, and laptops, to assist the low-end IoT devices. By offloading the computing-intensive tasks to the edge devices, it is expected to converge the data collection at IoT devices and the data processing at edge devices to provision computing-intensive and delay-sensitive services.

Ibrar et al. [14] providing a promising approach to enable ultra-reliable and delay-sensitive applications with high vehicle mobility over SDV-F. We propose ART Net, an AI-based Vehicle-to-Everything (V2X) framework for resource distribution and optimized communication using the SDV-F architecture. ART Net offers ultra-reliable and low-latency communications, particularly in highly dynamic environments, which is still a challenge in IoV. ART Net is composed of intelligent agents/controllers, to make decisions intelligently about (i) maximizing resource utilization at the fog layer, and (ii) minimizing the average end-to-end delay of time-critical IoV applications. Moreover, ART Net is designed to assign a task to fog nodes based on their states. Our experimental results show that considering a dynamic IoV environment, ART Net can efficiently distribute the fog layer tasks while minimizing the delay.

Malik et al. [15] present an overview of massive IoT and 6G-enabling technologies. We discuss different energy-related challenges that arise while using fog computing in 6G-enabled massive IoT. We categorize different energy-efficient fog computing solutions for IoT and describe the recent work done in these categories. Finally, we discuss future opportunities and open challenges in designing energy-efficient techniques for fog computing in the future 6G massive IoT network.

Zhu et al. [16] formulate a weighted sum minimization problem of task completion time and energy consumption at the local fog for achieving efficient task computation. Further, a deep learning-based joint offloading decision and resource allocation (DL-JODRA) algorithm is developed to address such problem by jointly optimizing offloading action, local CPU, bandwidth and external CPU occupation ratios. The optimal offloading decision based comprehensive optimization consideration of network resources further improves the network efficiency. Finally, the extensive simulation results demonstrate that the proposed DL-JODRA can achieve optimal offloading decision with low computation resource requirement and gain significant reduction on network costs (i.e., delay and energy) comparing with benchmark methods.

Tang et al. [17] investigate the decentralized partially observable offloading problem in the EH-enabled IoT fog system, in which multiple IoT devices cooperate to maximize the network performance while meeting their QoE requirements. We formulate the optimization problem as a decentralized partially observable Markov decision process (Dec-POMDP) in which each IoT device makes the task offloading decisions according to its local observation of the environment. The Lagrangian approach and the policy gradient method are adopted to find the optimal solution for the proposed problem. Due to the high complexity of solving the Dec-POMDP, a learning-based decentralized offloading algorithm with low complexity is presented to find the approximate optimal solution. Finally, extensive experimental evaluation and comparison are carried out to show the effectiveness of the proposed scheme.

Mebrek et al. [18] studies the energy efficiency and quality of service (QoS) issues in IoT-Fog-Cloud systems by proposing a joint optimization of resource allocation and workload dispatching over a fog-cloud system. The joint communication and computing optimization problem is formulated by a Nash Equilibrium problem (NEP), which

allows the trade-off between consumed energy by the system and QoS. To break the curse of large-scale systems, we propose a Reinforcement Learning-based algorithm that allows users to learn the optimal policy without having a priori knowledge of the dynamic statistics of the system. Finally, we conduct simulation experiments based and comparison two benchmarks. Evaluations and comparisons demonstrate the efficiency of our proposal.

Sen et al. [19] propose a heuristic for solving the problem and design the reinforcement model based on the output of the proposed heuristic. Our simulation results show that RILTA can reduce the task processing time and energy consumption with higher timeliness guarantee in comparison to other existing methods by 13 - 22% and 1 - 10% respectively.

Erkin et al.[20]propose encrypting private data and processing them under encryption to generate recommendations. By introducing a semitrusted third party and using data packing, we construct a highly efficient system that does not require the active participation of the user. We also present a comparison protocol, which is the first one to the best of our knowledge, that compares multiple values that are packed in one encryption. Conducted experiments show that this work opens a door to generate private recommendations in a privacy-preserving manner

Alaya et al.[21] presenting different known cryptosystems based, in a great part of its construction, on the homomorphic encryption, all joined with other techniques to enhance the cryptosystem performance and the privacy ratio. In addition, the homomorphic encryption has the feature to be highly adequate with any field it is used in, giving numerous advantages and a tremendous performance. Hence, the following survey presents different domains using homomorphic encryption and a final comparison between the adopted techniques.

Munjal et al.[22] an attempt is made to present a systematic review of homomorphic cryptosystems with its categorization and evolution over time. In addition, this paper also includes a review of homomorphic cryptosystem contributions in healthcare.

Tebba et al.[23] analyzes the application of different Homomorphic Encryption cryptosystems (RSA, Paillier, El Gamal, Goldwasser-Micali, Boneh-Goh-Nissim and Gentry) on a Cloud Computing platform. They are compared based on four characteristics; Homomorphic Encryption type, Privacy of data, Security applied to and keys used.
Dowlin et al. [24] introduces homomorphic encryption to the bioinformatics community, and presents an informal "manual" for using the Simple Encrypted Arithmetic Library (SEAL), which we have made publicly available for bioinformatic, genomic, and other research purposes.

Makkaoui et al. [25] examine the challenges facing Homomorphic Encryption methods to allow suppliers of cloud to perform operations on encrypted data, and provide the same results after treatment, as if they were performing calculations on raw data.

Gahi et al. [26] present a relational database system based on homomorphic encryption schemes to preserve the integrity and confidentiality of the data. Our system executes SQL queries over encrypted data. We tested our system with a recently developed homomorphic scheme that enables the execution of arithmetic operations on ciphertexts. We show that the proposed system performs accurate SQL operations, yet its performance discourages a practical implementation of this system.

Zhange et al. [27] construct three concrete secure cloud storage protocols using RSA-based, Paillier-based and DGHV-based HESs, which are multiplicatively, additively and fully HESs, respectively. We conduct extensive theoretical analysis and experimental evaluations to validate the practicability of the proposed protocol.

Seth et al. [28] ensure the protection of information, thus we offered a method to amend the Paillier Homomorphic algorithm without compromising the protection of existing technique. In our prospect work, we plan to propose an efficient Multicloud architecture so that information is stored, maintained and retrieved efficiently by employing a modified Paillier approach.

Comparison of Various Algorithms

Ref	Year	Algorithm used	Dataset used	Complexity	time	Results	Limitations
[8]	2018	UPRA	ZFBF matrix	low	-	Energy consumption 0.4 joule	Need to improve the real-time processing capability
[9]	2021	DLPBFT consensus algorithm	-	high	-	Memory utilization 1.2 joule	requireddigital asset trading among different fields.
[10]	2020	C-V2X computation-offloading network	D2D-V links	low	0.2 s	Throughput 0.12	Need to investigate the PLS analysis for non-orthogonal multiple access (NOMA) scheme
[11]	2021	efficient online secretary-based algorithm	SARSA and QL	high	-	Accuracy 88.4%.	the system spends a lot of time processing data neglecting the criticality
[12]	2022	non-orthogonal multiple access (NOMA)		low	-	Energy consumption 0.4 joule	Need multi-objective optimization technique to get better results.
[14]	2020	ARTNet	IoV environment,	high	0.5 s	Energy consumption 0.15 j	High communication cost
[15]	2022	provides an overview of massive IoT applications		-	-	-	discuss open challenges and highlight future research opportunities for

							improving energy efficiency.
[16]	2020	Deep learning with gradient descent	-	Low	-	Energy and time optimization	Need to improve Renewable energy queue
[17]	2020	Decentralized Markov decision process with policy gradient algorithm	-	Low	0.3 s	Reduced electricity cost	Energy harvesting enabled IoT devices do not have complete system information.
[18]	2019	Q-learning based reinforcement learning scheme with ascendant gradient	-	High	0.4 s	Energy efficient resource allocation	Distributed resource allocation decision by users without having prior knowledge of the system dynamics
[19]	2019	Q-learning based reinforcement learning scheme	-	High	0.5 s	Energy and time optimization	Energy efficient task scheduling among the three tiers i.e., cloud, fog and edge devices

Classification of Homomorphic Encryption

Partially Homomorphic Encryption (PHE) Schemas PHE was first attested use of homomorphic encryption introduce by Rivest in 1976.but it was called "privacy homomorphism" [1]. PHE which allows performing single operation either addition or multiplication 'n' number of times on encrypted data, that mean which allows any type of operation without any limitation. There are several algorithms well knowing for PHE [12] such as:

RSA Algorithm (1976):

Key Generation: Step 1: select p and q primes random numbers.

Step 2: calculate n = p.q and φ(n) = (p − 1)(q − 1).
Step 3: select e such that gcd(e, φ(n)) = 1.
Step 4: determine d such that e.d ≡ 1modφ(n).
Step 5: the public key pk = (e, n) and secret key is sk = (d)

- Encryption: Compute c = E(m) = me (mod n)
- Decryption: Compute m = D(E) = c d (mod n)
- Homomorphic Property: The homomorphic property of RSA shows following E(m1 ∗ m2) directly without ever decrypting it. The RSA is only support homomorphic over multiplicative, it does not support homomorphic over addictive of chiphertexts. Suppose m1, m2 ∈ME(m1)∗E(m2) = [me 1 (mod n)] ∗ [me 2 (mod n)] = (m1 ∗ m2) e (mod n) = E(m1 ∗ m2)
- Elgamal Algorithm (1985):

Key Generation:

Step 1: create an efficient cyclic group 'G' of order 'q' with generator 'g'.
Step 2: choose a random value x ∈ {1, 2, . . . q − 1}.
Step 3: compute h = g x .
Step 4: the public key ispk = (G, h, q, g) and x as private key.

• Encryption:
Step 1: chose random numberr∈ {1, 2, . . . q − 1}.
Step 2: compute c1 = g r and calculate the shared secret key is S = h r .
Step 3: convert the secret massage m into m 0 ∈ G.
Step 4: calculate c2 = m 0 ∗S
Step 5: the ciphertext pair are c = E(m) = (c1, c2) = (g r , m 0 ∗h r) = (g r , m 0 ∗(g x r)

• Decryption:
Step 1: compute shared secret key s = c x 1where x is secret key
Step 2:D(E) = c2.s−1 = mgxr.g−xr = m

• Homomorphic Property: E(m1) ∗ E(m2) = (g r1 , m 0 1 .hr1) ∗ (g r2 , m 0 2 .hr2)= g r1+r2 , m 0 1∗m 0 2h r1+r2) = E(m1 ∗ m2)

3) Pallier Cryptosystem (1999):

• Key Generation:
Step 1: choose p and q prime random number equal length such that gcd(pq,(p − 1)(q − 1)) = 1
Step 2: compute n = pq and λ = lcm(p − 1, q − 1) the lcm means Least Common Multiple
Step 3: choose integer random g ∈ Z ∗ such that gcd(L(g λmodn2), n) = 1 with L function define as follow L(u) = (u − 1)/n
Step 4: the public key pk = (n, g) and secret key is sk = (p, q)

• Encryption:
Step 1: select random number r ∈ Z ∗
Step 2: compute c = E(m) = g mr n(mod n 2)

- Decryption: Compute m = D(E) = (L(c λ (modn2))/(L(g λ (modn2))
- Homomorphic Property: E(m1) ∗ E(m2) = [g (m1)r n 1 (modn2)] ∗ [g (m2)r n 2 (modn2)] = g (m1 + m2)(r1 + r2) n(modn2) = E(m1 + m2)

B. Somewhat Homomorphic Encryption (SWHE) Schemas
SWHE allows performing different operations with limited number of times. There are several SWHE well known examples such as BGN encryption scheme which was the first practical SWHE developed by Beneh-Goh-Nissim BNG Algorithm (2005)

C. Fully Homomorphic Encryption (FHE) Schemas
FHE combines the advantage of PHE with SWHE, which allows to perform unlimited amount of operation for unlimited number of times. FHE was first practically proposed by Craig Gentry in 2009.

Applications of homomorphic Encryption

1) Securing Data Stored in the Cloud.
2) Enabling Data Analytics in Regulated Industries.
3) Improving Election Security and Transparency

3. Conclusion

Cloud security, also known as cloud computing security, consists of a set of policies, controls, procedures and technologies that work together to protect cloud-based systems, data, and infrastructure. This work presents the

types of cloud storage environments, and importance of cloud storage security. Additionally, homomorphic encryption technique is proposed for cloud storage security purpose.

References

[1] J. Wu, L. Ping, X. Ge, Y. Wang and J. Fu, "Cloud Storage as the Infrastructure of Cloud Computing," 2010 International Conference on Intelligent Computing and Cognitive Informatics, 2010, pp. 380-383, doi: 10.1109/ICICCI.2010.119.

[2] M. Lakshmi Neelima and M. Padma, "a Study on Cloud Storage," Int. J. Comput. Sci. Mob. Comput., vol. 35, no. 5, pp. 966–971, 2014, [Online]. Available: https://ijcsmc.com/docs/papers/May2014/V3I5201499a81.pdf

[3] Odun-Ayo, O. Ajayi, B. Akanle, and R. Ahuja, "An Overview of Data Storage in Cloud Computing," in 2017 International Conference on Next Generation Computing and Information Systems (ICNGCIS), 2017, pp. 29–34. doi: 10.1109/ICNGCIS.2017.9.

[4] K. Lee, "Comments on 'Secure Data Sharing in Cloud Computing Using Revocable-Storage Identity-Based Encryption,'" IEEE Trans. Cloud Comput., vol. 8, no. 4, pp. 1299–1300, 2020, doi: 10.1109/TCC.2020.2973623.

[5] C. Yang, Y. Liu, X. Tao, and F. Zhao, "Publicly Verifiable and Efficient Fine-Grained Data Deletion Scheme in Cloud Computing," IEEE Access, vol. 8, pp. 99393–99403, 2020, doi: 10.1109/ACCESS.2020.2997351.

[6] Y. Yang, X. Zheng, C. Rong, and W. Guo, "Efficient Regular Language Search for Secure Cloud Storage," IEEE Trans. Cloud Comput., vol. 8, no. 3, pp. 805–818, 2020, doi: 10.1109/TCC.2018.2814594.

[7] J. Yu and R. Hao, "Comments on 'SEPDP: Secure and Efficient Privacy Preserving Provable Data Possession in Cloud Storage,'" IEEE Trans. Serv. Comput., vol. 14, no. 6, pp. 2090–2092, 2021, doi: 10.1109/TSC.2019.2912379.

[8] L. Wang, M. Guan, Y. Ai, Y. Chen, B. Jiao, and L. Hanzo, "Beamforming-Aided NOMA Expedites Collaborative Multiuser Computational Offloading," IEEE Trans. Veh. Technol., vol. 67, no. 10, pp. 10027–10032, 2018, doi: 10.1109/TVT.2018.2853675.

[9] F. Bai, T. Shen, Z. Yu, K. Zeng, and B. Gong, "Trustworthy Blockchain-Empowered Collaborative Edge Computing-as-a-Service Scheduling and Data Sharing in the IIoE," IEEE Internet Things J., vol. 9, no. 16, pp. 14752–14766, 2022, doi: 10.1109/JIOT.2021.3058125.

[10] B. Qiu, H. Xiao, A. T. Chronopoulos, D. Zhou, and S. Ouyang, "Optimal Access Scheme for Security Provisioning of C-V2X Computation Offloading Network With Imperfect CSI," IEEE Access, vol. 8, pp. 9680–9691, 2020, doi: 10.1109/ACCESS.2020.2964795.

[11] M. Tiwari, S. Misra, P. K. Bishoyi, and L. T. Yang, "Devote: Criticality-Aware Federated Service Provisioning in Fog-Based IoT Environments," IEEE Internet Things J., vol. 8, no. 13, pp. 10631–10638, 2021, doi: 10.1109/JIOT.2021.3049326.

[12] S. Mostafa, C. W. Sung, and Y. Guo, "Joint Computation and Communication Resource Allocation With NOMA and OMA Offloading for Multi-Server Systems in F-RAN," IEEE Access, vol. 10, pp. 24456–24466, 2022, doi: 10.1109/ACCESS.2022.3152531.

[13] J. Cao, D. Zhang, H. Zhou, and P.-J. Wan, "Guest Editorial Emerging Computing Offloading for IoTs: Architectures, Technologies, and Applications," IEEE Internet Things J., vol. 6, no. 3, pp. 3987–3993, 2019, doi: 10.1109/JIOT.2019.2921217.

[14] M. Ibrar et al., "ARTNet: Ai-Based Resource Allocation and Task Offloading in a Reconfigurable Internet of Vehicular Networks," IEEE Trans. Netw. Sci. Eng., vol. 9, no. 1, pp. 67–77, 2022, doi: 10.1109/TNSE.2020.3047454.

[15] U. M. Malik, M. A. Javed, S. Zeadally, and S. ul Islam, "Energy-Efficient Fog Computing for 6G-Enabled Massive IoT: Recent Trends and Future Opportunities," IEEE Internet Things J., vol. 9, no. 16, pp. 14572–14594, 2022, doi: 10.1109/JIOT.2021.3068056.

[16] X. Zhu, S. Chen, S. Chen, and G. Yang, "Energy and Delay Co-aware Computation Offloading with Deep Learning in Fog Computing Networks," in 2019 IEEE 38th International Performance Computing and Communications Conference (IPCCC), 2019, pp. 1–6. doi: 10.1109/IPCCC47392.2019.8958729.

[17] Q. Tang, R. Xie, F. R. Yu, T. Huang, and Y. Liu, "Decentralized Computation Offloading in IoT Fog Computing System With Energy Harvesting: A Dec-POMDP Approach," IEEE Internet Things J., vol. 7, no. 6, pp. 4898–4911, 2020, doi: 10.1109/JIOT.2020.2971323.

[18] Mebrek, M. Esseghir, and L. Merghem-Boulahia, "Energy-Efficient Solution Based on Reinforcement Learning Approach in Fog Networks," in 2019 15th International Wireless Communications & Mobile Computing Conference (IWCMC), 2019, pp. 2019–2024. doi: 10.1109/IWCMC.2019.8766441.

[19] T. Sen and H. Shen, "Machine Learning based Timeliness-Guaranteed and Energy-Efficient Task Assignment in Edge Computing Systems," in 2019 IEEE 3rd International Conference on Fog and Edge Computing (ICFEC), 2019, pp. 1–10. doi: 10.1109/CFEC.2019.8733153.

[20] K. Munjal and R. Bhatia, "A systematic review of homomorphic encryption and its contributions in healthcare industry," Complex Intell. Syst., 2022, doi: 10.1007/s40747-022-00756-z.

[21] Z. Erkin, T. Veugen, T. Toft, and R. L. Lagendijk, "Generating Private Recommendations Efficiently Using Homomorphic Encryption and Data Packing," IEEE Trans. Inf. Forensics Secur., vol. 7, no. 3, pp. 1053–1066, 2012, doi: 10.1109/TIFS.2012.2190726.

[22] B. Alaya, L. Laouamer, and N. Msilini, "Homomorphic encryption systems statement: Trends and challenges," Comput. Sci. Rev., vol. 36, p. 100235, 2020, doi: https://doi.org/10.1016/j.cosrev.2020.100235.

[23] M. Tebaa and S. El Hajji, "Secure Cloud Computing through Homomorphic Encryption," vol. 5, no. December, pp. 29–38, 2014, [Online]. Available: http://arxiv.org/abs/1409.0829

[24] N. Dowlin, R. Gilad-Bachrach, K. Laine, K. Lauter, M. Naehrig, and J. Wernsing, "Manual for Using Homomorphic Encryption for Bioinformatics," Proc. IEEE, vol. 105, no. 3, pp. 552–567, 2017, doi: 10.1109/JPROC.2016.2622218.

[25] K. El Makkaoui, A. Ezzati, and A. B. Hssane, "Challenges of using homomorphic encryption to secure cloud computing," in 2015 International Conference on Cloud Technologies and Applications (CloudTech), 2015, pp. 1–7. doi: 10.1109/CloudTech.2015.7337011.

[26] Y. Gahi, M. Guennoun, and K. El-khatib, "A Secure Database System using Homomorphic Encryption Schemes," Security, no. c, pp. 54–58, 2011, [Online]. Available: http://www.thinkmind.org/index.php?view=article&articleid=dbkda_2011_3_20_30074

[27] J. Zhang, Y. Yang, Y. Chen, J. Chen, and Q. Zhang, "A general framework to design secure cloud storage protocol using homomorphic encryption scheme," Comput. Networks, vol. 129, pp. 37–50, 2017, doi: https://doi.org/10.1016/j.comnet.2017.08.019.

[28] B. Seth, S. Dalal, and R. Kumar, "Hybrid Homomorphic Encryption Scheme for Secure Cloud Data Storage," in Recent Advances in Computational Intelligence, R. Kumar and U. K. Wiil, Eds. Cham: Springer International Publishing, 2019, pp. 71–92. doi: 10.1007/978-3-030-12500-4_5.