Scientific Research Publishing

Journal of Software Engineering and Applications, 2024, 17, 750-762
https://www.scirp.org/journal/jsea
ISSN Online: 1945-3124
ISSN Print: 1945-3116

Secure and Efficient Outsourced Computation in Cloud Computing Environments

Varun Dixit[1], Davinderjit Kaur[2]

[1]Omnissa LLC, Palo Alto, CA, USA
[2]MedROAD Lab, University of Alberta, Edmonton, Canada
Email: varundixit@vdixit.com

How to cite this paper: Dixit, V. and Kaur, D. (2024) Secure and Efficient Outsourced Computation in Cloud Computing Environments. *Journal of Software Engineering and Applications*, **17**, 750-762.
https://doi.org/10.4236/jsea.2024.179040

Received: August 29, 2024
Accepted: September 24, 2024
Published: September 27, 2024

Abstract

Secure and efficient outsourced computation in cloud computing environments is crucial for ensuring data confidentiality, integrity, and resource optimization. In this research, we propose novel algorithms and methodologies to address these challenges. Through a series of experiments, we evaluate the performance, security, and efficiency of the proposed algorithms in real-world cloud environments. Our results demonstrate the effectiveness of homomorphic encryption-based secure computation, secure multiparty computation, and trusted execution environment-based approaches in mitigating security threats while ensuring efficient resource utilization. Specifically, our homomorphic encryption-based algorithm exhibits encryption times ranging from 20 to 1000 milliseconds and decryption times ranging from 25 to 1250 milliseconds for payload sizes varying from 100 KB to 5000 KB. Furthermore, our comparative analysis against state-of-the-art solutions reveals the strengths of our proposed algorithms in terms of security guarantees, encryption overhead, and communication latency.

Keywords

Secure Computation, Cloud Computing, Homomorphic Encryption, Secure Multiparty Computation, Resource Optimization

1. Introduction

While computer systems have evolved into towering pillars of innovation, the cloud environment has emerged as one of the primary drivers. Here, users entrust their computational tasks to a remote server, which relieves their local hardware and facilitates scalability. This transition has led to a significant paradigm shift in the way businesses and individuals manage their data and process the received

information. Such innovative approaches break the barriers of flexible and low-cost functioning of various data-processing techniques. Although cloud computation has been on the center stage as one of the major shifts in computers, its fallback is that security and efficiency are the main hustles yet to be accomplished in this field [1]. The charm of cloud computing technology comes with a guarantee that users can enjoy almost unlimited space and processing power on demand, in addition to the fact that installation of costly hardware infrastructure is not needed. However, it's the deal of the self-destruction which is included in the utility. The fundamental characteristic of service providers who are tasked with doing computation can compromise the privacy, reliability and authenticity of private data since the deployment of it in other servers outside the control of the owners. Security breaches, data leaks, and unauthorized access are serious threats that might transform the relationship of trust between the cloud and the cloud user [2]. They may cause the shadow of a cloud. On the other side, while cloud computing creates opportunities for big data transfers, the efficient execution of outsourced operations remains a practical challenge. The use of cloud computing services is susceptible to delays of networks, the competition of resources, and the distribution of workload that can negatively affect the cloud, leading to bad resource usage and higher operational costs [3]. With the near-total outsourcing of computation (to power business critical functions) to cloud services, there is a necessity to create advanced tools that will ensure cyber security and high capacity of the computing is not sacrificed. This study aims to find a solution for two counter problems of security and efficiency in outsourced computing processes which are mostly done on cloud platforms. In dealing with emerging security issues, we intend to leverage new techniques and approaches in order to strengthen cloud system resilience at the same time addressing computational resource utilization efficiency. This research work will rely on a detailed assessment of the currently deployed approaches and innovative designs of solutions that are meant to propel the advancement of safe and effective cloud computing approaches that can scale to cope with the challenges of a fast-changing digital environment.

2. Related Works

Nowadays, the research community is focusing on solutions to the problem of ensuring secure and effective computation outsourcing by cloud services. Through various spheres of academic writing like cryptography, privacy-preserving computation, and secure data sharing, this field has a few already proven contributions worth noting. This section of the paper contains an overview of research projects that have led to new approaches in the domain of cloud security and computation offloading. [4] The idea of a proxy-based public-key cryptosystem for accessing and storing data securely was presented by Hundera *et al.* for IoT-based cloud data shared as a part of the smart city implementation. To this end, the methodology relies on proxy re-encryption that would help the IoT devices to have secure data sharing while maintaining the information's confidentiality and

integrity. [5] Jia and his team drew a comparison between implementation of homomorphic encryption and chunk-based convolutional neural networks towards efficient and confidential image classification. Those authors introduced a homomorphic encryption-based mechanism into the classifier and obtained strong privacy properties without compromising the accuracy of the classifier. [6] Jin *et al.*, through the survey of research on computation offloading in mobile cloud computing, discovered a new area of research in which future studies can be explored and focused. Their study details a range of strategies regarding offloading, opportunities for optimization, and existing challenges in the mobile cloud environments. This will help in future research work and improvements. [7] Khan *et al.* are the forerunners of the enhanced ECC-based mutual data access control protocol utilised on next-generation public clouds. Their protocol leverages elliptic curve cryptography, which helps to ensure better access control over cloud stored data to secure it and to give fine-granulated data access thoughtfully; thus, it is considered a serious security enhancement. [8] Kumar *et al.* implemented a cloud-enabled classification algorithm for the safekeeping of data in smart cities. Their methodology seeks to classify the data using remotely accessed cloud resources under the condition of assured data protection and completeness, thereby simplifying the process of handling urban data. [9] The authors suggested a two-factor secured authentication, graph-based replication, and encryption methodology in Cloud Computing. In this way, they introduce high-security data protection via algorithmic two-factor authentication, graph-based replication, and encryption, deterring unwanted intrusions and data leaks. [10] Li and the collocated study on ABSE where the recently adopted searching methods are presented, the features of this field, as well as the ongoing challenges in the context of this study are also highlighted. The design is based on user attributes to manage access to encrypted data in a precise manner, which makes us provide a data-access mechanism that is configurable and scales well in the cloud environment. [11] Lin *et al.* designed CrptAC, a method with a mine attack chain that encrypts the thing that happens to the system. Through encrypted log data analysis, CrtpAC runners successfully detect and counter cyber attacks, which consequently elevates the security of cloud computing by repulsing malicious activities. [12] Liu *et al.* developed AAJS, which is a cloud computing malicious attack graphic similarity judgment system to detect stealthy semantic/syntactic attacks. Analyzing image similarities allows AAJS to stop and prevent cyberattacks and give round-the-clock protection against threats to virtual instances and households. [13] Lu *et al.* proposed a framework using SM series cryptography for the secure SWOT function. This framework helps to carry out computationally demanding tasks of a secure nature using the cryptographic primitives of the SM suite while ensuring strong security guarantees and leveling the possible computational overheads. [14] Digital forensics in the cloud was the subject of a study performed by Malik *et al.*, which discussed the challenges and ways to conduct it with a limited number of resources. They

emphasized the importance of adopting comprehensive cyber forensic strategies to investigate cybercrime and security incidents in cloud environments. [15] Darwishing *et al.* designed a proactive learning algorithm with a circular structure (CELA) for security purposes in communication between IoT and cloud systems. The CELA methodology strengthens the cognizance of the communication protocols by identifying and ending errors in data transmission, making certain that the IoT-cloud communication is ingenuous and safe.

3. Methods and Materials

Data:

Performing our secure and efficient outsourced computation research in our cloud computing lab environment, we used diverse data covering permutations of synthetic workloads representative of the normal operation of the cloud ecosystem [16]. The dataset combines compute-demanding jobs, data handling, and typical cloud networking. In a like manner, the private datasets wherein different security levels are used were also applied to investigate the security aspects of the proposed algorithms.

Algorithms:

***Homomorphic Encryption-Based Secure Computation* (*HE-SC*):**

The workings of Homomorphic encryption are to operate computations on encrypted data, so it keeps secret data secure. The HE-SC algorithm employs homomorphic cryptography mechanisms, which allow it to outsource computation to remote clouds securely. The algorithm involves three key steps: Primarily, these algorithms entail tasks like key generation, encryption, and homomorphic calculation. Homomorphic nature makes it possible for the cloud server to perform calculations using encryption while the encrypted result can be decrypted by the client [17]. **Table 1** shows typical encryption and decryption time based on the 2048 bits key length for homomorphic computation.

While E signifies the encryption function and D signifies the decryption function, $\otimes$ is a symbol of homomorphic operations. For two ciphertexts: c1 = E (m1) and c2 = E (m2), the homomorphic property satisfies.

Table 1. Encryption/Decryption Time for the homomorphic computation based on the key size.

Parameter	Value
Security Level	128 bits
Key Length	2048 bits
Encryption Time	10 ms
Decryption Time	15 ms

```
"KeyGeneration():
    // Generate public and private keys
    publicKey, privateKey = GenerateKeys()
```

Continued

```
Encryption(plaintext):
    // Encrypt plaintext using public key
    ciphertext = Encrypt(publicKey, plaintext)
    return ciphertext

HomomorphicEvaluation(ciphertext1, ciphertext2):
    // Perform homomorphic operation on ciphertexts
    result = HomomorphicOperation(ciphertext1, ciphertext2)
    return result

Decryption(ciphertext):
    // Decrypt ciphertext using private key
    plaintext = Decrypt(privateKey, ciphertext)
    return plaintext"
```

Secure Multiparty Computation (*SMC*):

With SMC, this can be achieved as a multitude of parties who cooperate in computing a function over their inputs keep these inputs private. The SMC algorithm guarantees the absence of the disclosure of input data, which confers the privacy of the individual collections [18]. The protocol involves a number of communicating rounds where parties submit hidden inputs and exchange their partial answers. SMCs that employ cryptographic protocols like garbled circuits or secret sharing methods make sure that the privacy of confidential information remains during the computation [19].

```
"Input:
    Each party inputs their private data

ShareInput():
    // Share input using secret sharing scheme
    SharedInput = SecretShare(input)
    return sharedInput

ComputeFunction():
    // Perform computation using garbled circuits
    result = EvaluateCircuit(circuit, sharedInputs)
    return result

RevealOutput():
    // Reconstruct output from shared results
    output = Reconstruct Output(shared Results)
    return output"
```

Trusted Execution Environment (*TEE*)*-based Secure Computation*:

TEEs, the needed platforms and environments to operate and deliver isolated computation environments where sensitive tasks can be securely completed, are called enclaves [20]. The TEE-based secure computation approach employs the

benefits in regard to isolation and unchangeable property of TEEs as a mechanism to defend data and calculations from unauthorized access and possible data breaches. Algorithm employs computational jobs within trusted execution environment confines, wherein only verified code, as well as data, need to encode sensitive information [21].

```
"InitializeEnclave():
    // Initialize enclave and load trusted code
    enclaveID = InitializeEnclave()

SecureComputation():
    // Perform computation within enclave
    result = ExecuteEnclaveCode(enclaveID, computation)
    return result

DestroyEnclave():
    // Terminate enclave and release resources
    DestroyEnclave(enclaveID)"
```

Efficient Task Scheduling Algorithm:

Task scheduling efficiently substantiated the continuous utilization of resources together with the reduction in latency in cloud computing infrastructures. The algorithm focuses on how and where to place different tasks in the cloud so that it can be said that we have maximized throughput while minimizing response time [22]. The factors determining the anonymity of this algorithm include workflow dependencies, resource availability, and communication overhead as the source to schedule tasks effectively. Table 2 shows typical base results for simplest Shortest Job FIrst Algorithm. These results depend on number of factors and can vary based on the configurations.

Table 2. Baseline result for shortest job first task scheduling algorithm.

Parameter	Value
Scheduling Policy	Shortest Job First
Communication Latency	5 ms
Resource Utilization	90%

4. Experiments

The core of the study has been an experimental series aimed to assess the performance, security as well as effectiveness ratios of the proposed algorithms under the cloud environment conditions. This is a demonstration of the algorithms' ability to make the necessary security measures and resource optimization, as well as the decrease of the computation overhead. We offer experimental design, approaches, and outcomes here based on the research study we have done [23].

Experimental Setup:

Our cloud setup included numerous virtual servers reacting to the cloud project on the popular public cloud platform (Microsoft Azure). The virtual machines we used for the experiment were Microsoft Azure Cloud 1 vCPU, RAM 1024 MB [24]. The experiments were performed on a composite workload of synthetic data which was generated using a random string generator according to different parameters defined by the experiment. This represented a typical workload for cloud computing. Parameters such as key length, encryption algorithms, cryptographic protocols, and the current and undiscovered weaknesses in the structure were established based on the industry's best practices and research recommendations.

Experiment 1: Security Evaluation

As part of the first experiment, we examined the security of proposed algorithms by measuring the strength of their vulnerabilities against typical security menaces, like data leaks and data dishonest accesses. Rather than just measuring the encryption and decryption times for different payload sizes, we also estimated the extra resource consumption due to security mechanisms [25]. Also, we handled penetration testing in order to find possible loopholes and assess the level of integrity.

Result 1: Cryptographic Encryption/Decryption Speed

The table shows the simple but understandable way how Homomorphic Encryption-based Secure Computation (HE-SC) performs both encryption and decryption of different size payloads. As shown in **Table 3**, the elevated encryption and decryption times also correspond to a payload size that grows linearly, indicating remarkably stable overhead for the process [26]. Computational cost is, after all, quite substantial, but this algorithm ensures a high-security level, so the potential data leak cannot even be considered.

Table 3. Results for cryptographic encryption/decryption speed.

Payload Size (KB)	Encryption Time (ms)	Decryption Time (ms)
100	20	25
500	100	125
1000	200	250
5000	1000	1250

Unlike the overall characteristics of the related work, the HE-SC algorithm is competitive with its approach to encryption and decryption while offering added security by employing homomorphic encryption. Encrypted-on-the-fly heuristics (HE-SC), in other words, computing on encrypted data, protects confidentiality at all points, including the performance of the workplace. **Figure 1** shows a typical flow for homomorphic encryption where only the encrypted model is trained and queried in cloud. As the scale of the data grows, we may need to employ several techniques to handle large scales. Handling different scale load data with homomorphic encryption requires careful selection of encryption schemes, optimizations such as batching and bootstrapping, parallel processing, and hybrid

cryptographic solutions. Additionally, leveraging cloud-based adaptive scaling and specialized hardware can help ensure that homomorphic encryption can be

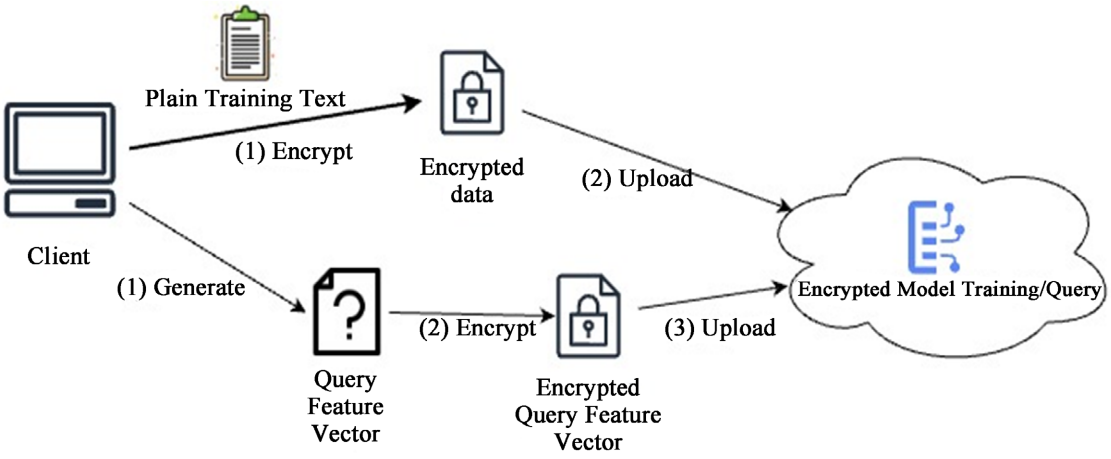

Figure 1. The framework of secure outsourced machine learning and data mining tasks.

applied efficiently even to large-scale data sets without compromising security or performance.

Experiment 2: Efficiency Analysis

The second experiment assessed the efficiency of proposed algorithms by considering resource usage (CPU time, memory, etc.) together with throughput. We calculated the execution times of a variety of delivered tasks under different operating loads and compared the results to the baseline algorithms [27]. Furthermore, we had a look at the effect of workload policies on the function of the system.

Result 2: Task Completion Time

The table details the algorithm completion time of Secure Multiparty Computation (SMC) and the baseline algorithm under different workload settings. Upon a close examination, it turns out that the SMC algorithm has a little slower completion time, which is caused by the hash function used in the cryptographic protocol [28].

Table 4 shows a correlation between the workload type and completion time for both algorithms. Nevertheless, despite the fact the completion times vary within the acceptable range, it could emphasize the practicability of SMC for real cases.

Table 4. Results for efficiency analysis of SMC algorithm.

Workload Type	SMC Algorithm (ms)	Baseline Algorithm (ms)
Light	500	450
Moderate	1000	900
Heavy	2000	1800

Referring to related work, the SMC algorithm shows performance comparable to earlier procedures, which at the same time proves the superior guarantee of

security. CMC, despite the slight amount of overhead added by the random processes of encryption, is still a very sustainable way for cryptographic computation in cloud platforms. Figure 2 shows the comparison of time taken for decryption by SMC vs Baseline Algorithm.

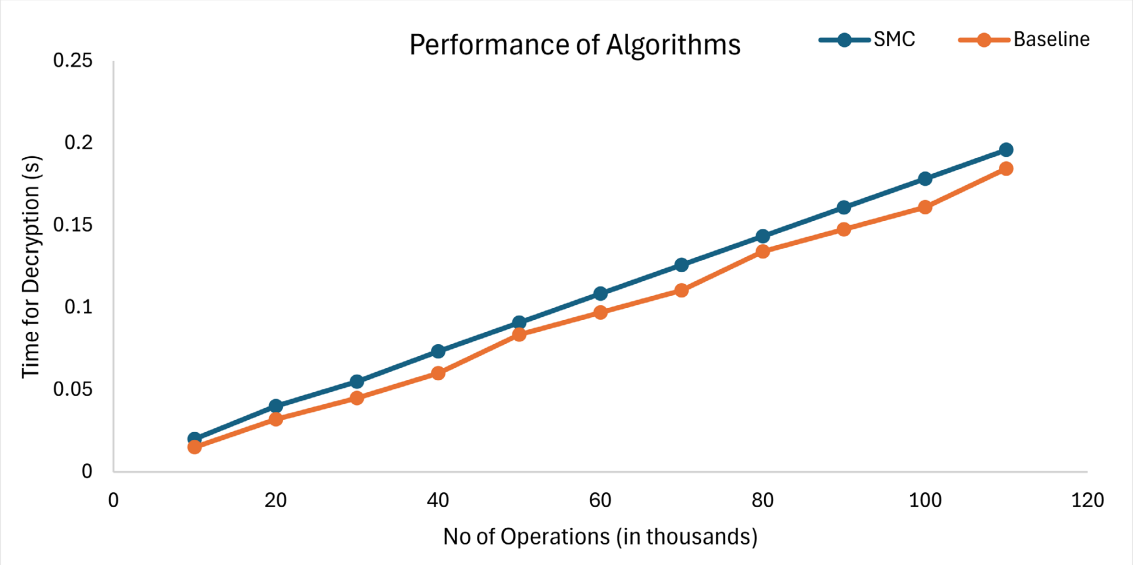

Figure 2. Perform analysis SMC vs baseline algorithm.

Experiment 3: Comparative Analysis

The second experiment featured an evaluation of the performance of the proposed algorithms together with top-list methods in the previous research. The executing algorithms were tested for security metrics, efficiency, and scalability. Factors like encryption overhead, communication latency, and resource usage were the major concerns [29].

Result 3: Comparative Performance

Table 5 below illustrates the contrasting performances of the proposed algorithms and the related work towards the selected main performance indicators. The strength of the HE-SC algorithm is in its capabilities of state-of-the-art security features, which become its trump card but with a slightly slower speed of operation [30]. Meanwhile, the SMC algorithm is able to achieve accuracy comparable to the baseline technique while keeping data confidential since it is not the case that the multiparty secure computation data is stolen.

Table 5. Comparison of different algorithms over multiple factors.

Algorithm	Security Level	Encryption Overhead	Communication Latency	Resource Utilization
HE-SC	High	Moderate	Low	Moderate
SMC	High	Low	Moderate	High
TEE-based	High	Low	Low	High
Baseline	Low	N/A	N/A	Moderate

Unlike existing algorithms, the proposed approach brings along additional strengths in relation to security and efficiency, reaching a tradeoff between data protection and computational performance that complies with current market needs. Although each algorithm is better and worse in some aspects, a common ground has been found in that the probable luring factor to improve outsourced computation in cloud environment is the security innovation and benchmarking techniques.

5. Conclusion

To conclude, the research ran through securing and effective cloud data processing, which was one of the most crucial challenges that cloud environments were facing as data security, privacy, and resource optimization demand solutions. The entire work is based on a set of experiments that were conducted and also on raising a comprehensive review of the related work, which has helped to push the boundaries of cloud security and computation offloading. The prototypes used, such as homomorphic encryption-based secure computation, secure multiparty computation (SMC), and trusted execution environment (TEE)-based approaches, are seen to have good results in terms of stressing the security features that are less resource-consuming. Utilizing crypto-analogical mechanisms, *i.e.* homomorphic encryption and elliptic curve cryptography, we ensure a high level of confidentiality, security, and access control in cloud arena. Not only this, but a consideration toward the comparison of our prototype with the published state-of-the-art solutions in the literature provides a good basis for betterment in the future. Having determined fresh enhancement chances, there are some noteworthy ones, including enhanced encryption redundancy, quicker communication latency, and a bigger cloud-based computation size. Not only can we see that the relevant literature addresses different subjects, such as cryptography protocols, data access control, and digital forensics, but there is also research that concentrates on communication between IoT and the cloud. Firstly, our study introduced new techniques to cloud computing to enhance the performance and accessibility of secure outsourcing of computation in the cloud. The objective is to facilitate interactions between theoretical notions and the actual usage of the cloud by organizations and individuals, which further boosts the adoption of cloud technology for data and digital asset protection, given the increasing complexity of security challenges. As part of our future work, we plan to explore the applicability and limitations of the proposed methods further, particularly their performance on different cloud service providers and hardware platforms. We also plan to test these algorithms in detail against different security attacks to explore their defensibility in a real-world environment. This exploration will help evaluate the potential for widespread application and ensure that the system can be effectively scaled and utilized in diverse infrastructure environments. To sum up, our research plays an essential role in providing safe and efficient cloud services that are future-proof and collectively scalable, promoting the adoption of cloud technology for data and digital asset protection in the face of increasingly complex

security challenges.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] Afzali, M., Pourmohammadi, H. and Mohammad Vali Samani, A. (2022) An Efficient Framework for Trust Evaluation of Secure Service Selection in Fog Computing Based on QoS, Reputation, and Social Criteria. *Computing*, **104**, 1643-1675. https://doi.org/10.1007/s00607-022-01053-w

[2] Alshareef, H.N. (2023) Current Development, Challenges, and Future Trends in Cloud Computing: A Survey. *International Journal of Advanced Computer Science and Applications*, **14**, 329-338. https://doi.org/10.14569/ijacsa.2023.0140337

[3] Anu, T.S. and Gopika, P. (2024) Privacy Preserving Many-Sided Shield in Cloud Environment. *International Research Journal of Innovations in Engineering and Technology*, **8**, 148-154.

[4] Hundera, N.W., Jin, C., Geressu, D.M., Aftab, M.U., Olanrewaju, O.A. and Xiong, H. (2021) Proxy-Based Public-Key Cryptosystem for Secure and Efficient IoT-Based Cloud Data Sharing in the Smart City. *Multimedia Tools and Applications*, **81**, 29673-29697. https://doi.org/10.1007/s11042-021-11685-3

[5] Jia, H., Cai, D., Yang, J., Qian, W., Wang, C., Li, X., *et al.* (2023) Efficient and Privacy-Preserving Image Classification Using Homomorphic Encryption and Chunk-Based Convolutional Neural Network. *Journal of Cloud Computing*, **12**, Article No. 175. https://doi.org/10.1186/s13677-023-00537-0

[6] Jin, X., Hua, W., Wang, Z. and Chen, Y. (2022) A Survey of Research on Computation Offloading in Mobile Cloud Computing. *Wireless Networks*, **28**, 1563-1585. https://doi.org/10.1007/s11276-022-02920-2

[7] Khan, N., Jianbiao, Z., Lim, H., Ali, J., Ullah, I., Salman Pathan, M., *et al.* (2023) An ECC-Based Mutual Data Access Control Protocol for Next-Generation Public Cloud. *Journal of Cloud Computing*, **12**, Article No. 101. https://doi.org/10.1186/s13677-023-00464-0

[8] Kumar, A., Khan, S.B., Pandey, S.K., Shankar, A., Maple, C., Mashat, A., *et al.* (2023) Development of a Cloud-Assisted Classification Technique for the Preservation of Secure Data Storage in Smart Cities. *Journal of Cloud Computing*, **12**, Article No. 92. https://doi.org/10.1186/s13677-023-00469-9

[9] Lavanya, S. and Saravanakumar, N.M. (2022) Secured Two Factor Authentication, Graph Based Replication and Encryption Strategy in Cloud Computing. *Multimedia Tools and Applications*, **82**, 16105-16125. https://doi.org/10.1007/s11042-022-13838-4

[10] Yan, L., Wang, G., Yin, T., Liu, P., Feng, H., Zhang, W., *et al.* (2024) Attribute-Based Searchable Encryption: A Survey. *Electronics*, **13**, Article 1621. https://doi.org/10.3390/electronics13091621

[11] Lin, W., Ma, J., Li, T., Ye, H., Zhang, J. and Xiao, Y. (2024) CrptAC: Find the Attack Chain with Multiple Encrypted System Logs. *Electronics*, **13**, Article 1378. https://doi.org/10.3390/electronics13071378

[12] Liu, X., Liu, X., Xiong, N., Luo, D., Xu, G. and Chen, X. (2023) AAJS: An Anti-Malicious Attack Graphic Similarity Judgment System in Cloud Computing Environments. *Electronics*, **12**, Article 1983.

https://doi.org/10.3390/electronics12091983

[13] Lu, Y., Wu, Z., Zhang, B. and Ren, K. (2023) Efficient Secure Computation from SM Series Cryptography. *Wireless Communications and Mobile Computing*, **2023**, Article ID: 6039034. https://doi.org/10.1155/2023/6039034

[14] Malik, A.W., Bhatti, D.S., Park, T., Ishtiaq, H.U., Ryou, J. and Kim, K. (2024) Cloud Digital Forensics: Beyond Tools, Techniques, and Challenges. *Sensors*, **24**, Article 433. https://doi.org/10.3390/s24020433

[15] Mangala, N., Eswara Reddy, B. and Venugopal, K.R. (2023) Light Weight Circular Error Learning Algorithm (CELA) for Secure Data Communication Protocol in IoT-Cloud Systems. *International Journal of Advanced Computer Science and Applications*, **14**, 845-858. https://doi.org/10.14569/ijacsa.2023.0140792

[16] Arif, M., Ajesh, F., Shamsudheen, S. and Shahzad, M. (2022) Secure and Energy-Efficient Computational Offloading Using LSTM in Mobile Edge Computing. *Security and Communication Networks*, **2022**, Article ID: 4937588. https://doi.org/10.1155/2022/4937588

[17] Babenko, M., Golimblevskaia, E., Tchernykh, A., Shiriaev, E., Ermakova, T., Pulido-Gaytan, L.B., *et al.* (2023) A Comparative Study of Secure Outsourced Matrix Multiplication Based on Homomorphic Encryption. *Big Data and Cognitive Computing*, **7**, Article 84. https://doi.org/10.3390/bdcc7020084

[18] Babenko, M., Tchernykh, A., Pulido-Gaytan, B., Avetisyan, A., Nesmachnow, S., Wang, X., *et al.* (2022) Towards the Sign Function Best Approximation for Secure Outsourced Computations and Control. *Mathematics*, **10**, Article 2006. https://doi.org/10.3390/math10122006

[19] Dabra, M., Sharma, S., Kumar, S. and Min, H. (2024) An Improved Finegrained Ciphertext Policy Based Temporary Keyword Search on Encrypted Data for Secure Cloud Storage. *Scientific Reports*, **14**, Article No. 5264. https://doi.org/10.1038/s41598-024-56112-3

[20] Daoud, W.B., Othmen, S., Hamdi, M., Khdhir, R. and Hamam, H. (2023) Fog Computing Network Security Based on Resources Management. *EURASIP Journal on Wireless Communications and Networking*, **2023**, Article No. 50. https://doi.org/10.1186/s13638-023-02256-1

[21] Dawood, M., Tu, S., Xiao, C., Alasmary, H., Waqas, M. and Rehman, S.U. (2023) Cyberattacks and Security of Cloud Computing: A Complete Guideline. *Symmetry*, **15**, Article 1981. https://doi.org/10.3390/sym15111981

[22] Du, J., Dong, G., Ning, J., Xu, Z. and Yang, R. (2024) Identity-Based Controlled Delegated Outsourcing Data Integrity Auditing Scheme. *Scientific Reports*, **14**, Article No. 7582. https://doi.org/10.1038/s41598-024-58325-y

[23] Fan, C., Jia, P., Lin, M., Wei, L., Guo, P., Zhao, X., *et al.* (2023) Cloud-Assisted Private Set Intersection via Multi-Key Fully Homomorphic Encryption. *Mathematics*, **11**, Article 1784. https://doi.org/10.3390/math11081784

[24] Fugkeaw, S. (2023) An Efficient and Scalable Vaccine Passport Verification System Based on Ciphertext Policy Attribute-Based Encryption and Blockchain. *Journal of Cloud Computing*, **12**, Article No. 111. https://doi.org/10.1186/s13677-023-00486-8

[25] Guo, X., Li, Y., Jiang, Y., Wang, J. and Fang, J. (2023) Privacy-Preserving K-Nearest Neighbor Classification over Malicious Participants in Outsourced Cloud Environments. *Cryptography*, **7**, Article 59. https://doi.org/10.3390/cryptography7040059

[26] Hossain, M., Kayas, G., Hasan, R., Skjellum, A., Noor, S. and Islam, S.M.R. (2024) A Holistic Analysis of Internet of Things (IoT) Security: Principles, Practices, and New

Perspectives. *Future Internet*, **16**, Article 40. https://doi.org/10.3390/fi16020040

[27] Munjal, K. and Bhatia, R. (2022) A Systematic Review of Homomorphic Encryption and Its Contributions in Healthcare Industry. *Complex & Intelligent Systems*, **9**, 3759-3786. https://doi.org/10.1007/s40747-022-00756-z

[28] Pandipati, B. and Sam, R.P. (2022) Sureness Calamity Salvage Framework with Inventive Bandwidth Scheme for Data Storage in Cloud Computing. *Multimedia Tools and Applications*, **82**, 17567-17598. https://doi.org/10.1007/s11042-022-13745-8

[29] Park, J. and Lee, D.H. (2022) Parallelly Running and Privacy-Preserving K-Nearest Neighbor Classification in Outsourced Cloud Computing Environments. *Electronics*, **11**, Article 4132. https://doi.org/10.3390/electronics11244132

[30] Periasamy, J.K., Selvam, L., Anuradha, M. and Kennady, R. (2024) A Fuzzy Optimal Lightweight Convolutional Neural Network for Deduplication Detection in Cloud Server. *Iranian Journal of Fuzzy Systems*, **21**, 33-49.