ISSN 2616-5775 Vol. 8, Issue 1: 13-18, DOI: 10.25236/AJCIS.2025.080103

Research on Secure Cloud Storage of Mobile Office Data Based on Homomorphic Encryption

Xiaoqian Zhang

Computer Simulation Magazine, Beijing, 100048, China

Abstract: *The dynamic nature of the mobile office environment and the high sensitivity of data make it impossible to effectively protect data privacy when encrypting data. In order to ensure the business continuity of the enterprise and at the same time improve the effect of data privacy protection, the study of secure cloud storage of mobile office data based on homomorphic encryption is proposed. Combined with the monitoring of the average information characteristics of the corresponding sequence, the discrete key transmission sequence of mobile office data is constructed using chaotic logistic inherent modal expansion sequence to analyze the data storage characteristics. Combined with homomorphic encryption algorithm for the public key encoding configuration of mobile office data, combined with the method of binary discrete random coding, it can realize the data encryption processing, and finally the layered architecture is used to build the cloud storage mechanism. Comparative experimental results show that the algorithm throughput is higher when the proposed method is used for data security storage, and it has a more ideal storage effect.*

Keywords: *homomorphic encryption; mobile office data; cloud storage; throughput*

1. Introduction

The convenience of mobile office greatly improves work efficiency, but it also poses unprecedented challenges to data security. Cloud storage, as the core infrastructure of mobile office, provides flexible data access services, but the security issues during data transmission and storage in the cloud are becoming more and more prominent. In particular, leakage and illegal access of sensitive data may not only lead to economic loss of enterprises, but also seriously damage user privacy[1]. Therefore, how to ensure the security of mobile office data has become an important issue to be solved in the field of information security at present. Traditional encryption techniques play an important role in protecting data security, but they have limitations in mobile office and cloud storage environments. Traditional encryption methods require data to be decrypted before use, a process that increases the risk of data leakage. Especially in cloud computing environments, untrusted cloud service providers have access to decrypted data, which in turn leads to data security issues. The emergence of homomorphic encryption provides new ideas to solve this problem. Homomorphic encryption allows computations to be performed directly on encrypted data without decryption, thus supporting various manipulations and analyses of data while protecting data privacy[2]. Homomorphic encryption has made significant progress over the decades since it was first proposed in 1978. Especially in recent years, with the rise of cloud computing and big data technology, homomorphic encryption technology has received extensive attention and research. At present, homomorphic encryption techniques are mainly categorized into three types: partial homomorphic encryption (PHE), homomorphic-like encryption (SWHE) and full homomorphic encryption (FHE). Among them, full homomorphic encryption technology can support unlimited addition and multiplication operations, which is the hot and difficult point of current research. In recent years, researchers have proposed a variety of fully homomorphic encryption schemes based on different mathematical puzzles, such as those based on the problems of ideal lattice, LWE (Learning With Errors) and RLWE (Ring-LWE). These schemes have made important breakthroughs in terms of algorithmic efficiency, security and practicality[3]. Aiming at the problems of low computational efficiency and high complexity of existing homomorphic encryption algorithms, this paper proposes an optimization algorithm to improve the execution efficiency of the algorithms by reducing the key size and lowering the computational complexity.

ISSN 2616-5775 Vol. 8, Issue 1: 13-18, DOI: 10.25236/AJCIS.2025.080103

2. Characterization of mobile office data storage

In order to realize the design of mobile office data security storage based on homomorphic encryption, the features of data storage are firstly analyzed. Combined with the average information feature monitoring of the corresponding sequence, the discrete key transmission sequence of mobile office data is constructed by using chaotic logistic intrinsic modal expansion sequence, and the stochastic linear feature detection method is used to establish the feature decomposition model of mobile office data storage, as shown in Fig. 1.

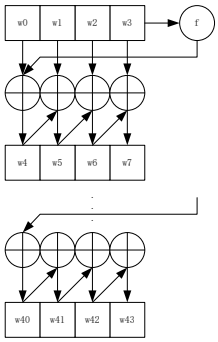

Figure1: Feature decomposition model for mobile office data storage

Using the method of balanced access control under the optimal policy, the key statistic of mobile office data encryption is constructed at $P-value \geq 0.01$, and through the parameter feature decomposition and vector parsing control, the subspace parameter of mobile office data encryption is obtained to satisfy the condition of $KS \in \{0,1\}$, which is expressed as a 1-bit coding mapping[4]. Given the original hypothesis H_0 (assuming that the mobile office data sequence to be tested is random) and the other alternative hypothesis H_1 (the mobile office data sequence is not random), the ciphertext characteristic encoding generalized function expression of mobile office data is obtained as follows.

$$f^{-1}(I)=\begin{cases} p*I, s=0 \\ 1-(1-p)*I, s=1 \end{cases} \tag{1}$$

Among them, p is the pseudo-random number generation key, I represents the private key of the sender of the mobile office data, s is the random sequence code element, the initial value of the encryption key of the mobile office data is set $I=[0,1]$, the successive feature bits in the sequence of bits are disambiguation detected, and the correlation feature vector of the mobile office data is input v [5]. The encryption standard algorithm is used to calculate the public key pk of the subspace distribution of the mobile office data transmission, and under the control of the private key sk , the final coding interval expression of the office data is obtained as shown below.

$$f(x)=\begin{cases} x/P_1, x\in I_1 \\ (x-P_1)/P_2, x\in I_2 \\ \ldots \\ \left(x-\sum_{i=1}^{n-1}P_i\right)/P_n, x\in I_n \end{cases} \tag{2}$$

where P_i denotes the similarity probability of the ciphertext distribution of the wireless sensor network data, and the probability of the occurrence of each symbol in s can be expressed as[6]. Thus, the arithmetic coding and quantized feature resolution of mobile office data can be realized by encryption key space structure reset, so as to initially plan the spatial structure of data storage and facilitate data management.

ISSN 2616-5775 Vol. 8, Issue 1: 13-18, DOI: 10.25236/AJCIS.2025.080103

3. Homomorphic encryption processing of mobile office data

A mobile office data network is a network of multiple nodes that are typically distributed in a flexible environment. These nodes are able to automatically sense the physical and chemical quantities in the environment and transmit this pooled information to a data processing center for processing. Since data can be easily stolen or tampered with during transmission, it needs to be encrypted. Homomorphic encryption is an encryption technique that can perform operations while keeping the data encrypted[7]. Homomorphic encryption enables arithmetic operations such as addition and multiplication without decryption. Homomorphic encryption of data means that after encrypting the data, the encrypted data can be added, subtracted, multiplied, and other arithmetic operations can be carried out under the premise of guaranteeing the security of the data, and the final result returned is also in the encrypted state[8]. This allows calculations to be performed without revealing the data, thus protecting the privacy and security of the data.

In the application of mobile office data networks, homomorphic encryption can play an important role in the process of data collection and analysis. In order to protect the data collected by the sensing nodes from being tampered with or leaked by third parties, it is necessary to encrypt these data[9]. After the encrypted data is transmitted to the data processing center, homomorphic encryption technology can be used to achieve multi-party data co-processing, thus realizing efficient and secure processing and analysis of data. In general, mobile office data homomorphic encryption technology can realize multi-party data cooperative processing under the premise of ensuring the security of data encryption, which improves the efficiency and security of data processing. Using homomorphic encryption methods, calculations can be performed without decrypting no data, primary encryption is performed, and the public key is configured for secondary encryption to improve the security of data storage[10]. The coding scheme of homomorphic encryption is designed to carry out the public key coding configuration of mobile office data, combined with the method of binary discrete random coding, to obtain the data encryption plaintext block monitoring function as $t_i = H_1(ID, upk_i)$, from which the correlation eigenquantity expression of the plaintext sequence and the circular shift key can be obtained as shown below.

$$dsk_{ID_i} = \left(sk_{i1}, sk_{i2}\right) = \left(g_2^n \left(g_1^t \cdot h\right), g^{ui}\right) \tag{3}$$

$$rsk_{ID_i} = \left(sr_i = g_1^t\right) \tag{4}$$

Among them, sk_{i1} sk_{i2} represents the homomorphic eigencomponent of chaotic sequence encryption, g_2^n represents the key eigenvalue of encryption for each chunk encoding, g_1^t represents the first chunk ciphertext, h represents the encryption depth, and g^{ui} represents the random disorder parameter[11]. By updating the cyclic key, the private key agreement for homomorphic encryption of office data can be obtained, and the encryption of office data can be realized in this way.

4. Cloud Storage Mechanism Design

After completing the above data encryption processing, this paper realizes data security storage by building the cloud storage mechanism. The specific mechanism structure is shown in Figure 2.

This cloud storage mechanism adopts a layered architecture design, which mainly includes client layer, encryption processing layer, cloud storage service layer, data processing layer and security management layer[12]. Each layer interacts with each other through security protocols and interfaces to ensure secure data transmission and processing. Among them, the user interface of the client layer provides a friendly user interface to support users to upload, download, query and modify the encrypted data stored in the cloud[13]. The local key management module is used to securely store the private key on the user's device and provides the functions of key generation, storage, backup and recovery. It also uses homomorphic encryption algorithms to encrypt data that is about to be uploaded, ensuring that the data is encrypted before it leaves the user's device. The encryption processing layer mainly includes key distribution and data encapsulation modules. The key distribution module distributes the public key to the cloud storage service layer through a secure channel to ensure the secure transmission of the

ISSN 2616-5775 Vol. 8, Issue 1: 13-18, DOI: 10.25236/AJCIS.2025.080103

encryption key. The data encapsulation module encapsulates the encrypted data and adds necessary metadata (e.g., file type, size, encryption time, etc.) to facilitate processing and management by the cloud storage service layer[14]. The cloud storage service layer uses a distributed file system or object storage system to decentralize data storage on multiple nodes to improve data reliability and availability. Data access control implements fine-grained access control policies to ensure that only authorized users can access the encrypted data stored in the cloud. The data processing layer includes a homomorphic computation engine, which provides the ability to perform homomorphic computation on encrypted data and supports mathematical operations such as addition and multiplication, so that computation results can be obtained without decryption[15]. Based on user requests, computing resources are scheduled to process the encrypted data and return the results to the user. The security management layer conducts regular security audits of the cloud storage system, checking system configurations, access logs, etc., so as to detect and deal with potential security threats in a timely manner. It also deploys a real-time threat detection system to continuously monitor the cloud storage environment and take countermeasures as soon as abnormal behavior is detected.

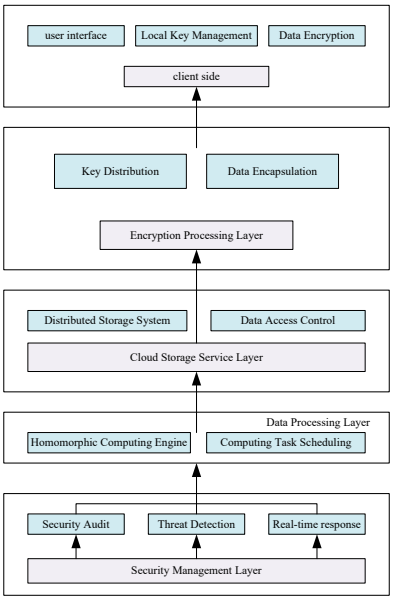

Figure2: Architecture of the cloud storage mechanism

5. Experimental component

5.1 Description of the experiment

Considering the authenticity and reliability of the final test results, a comparative approach is adopted to carry out the analysis, a mobile office data set is selected as the main target object of the test, and literature [1] method and literature [2] method are used as the control group, and the method of this paper is the experimental group. According to the actual data storage requirements and changes in standards, the test results are compared and finally obtained. Next, the synthesized fully homomorphic encryption algorithm is used to build the basic test environment.

The smart mobile office behavior analysis dataset constructed by the experiment contains a total of 500,000 records, covering a variety of smart mobile office terminals, such as smartphones, tablets and laptops. In the design of the dataset structure, the experiment adopts a multi-table association approach to ensure the integrity and flexibility of the data. The core tables include the "User Information Table", which records the basic information of 10,000 employees in different departments, positions, and workplaces; the "Task Management Table", which records the creation, assignment, completion status, and deadline of over 300,000 tasks. The "Task Management Table" records more than 300,000 tasks' creation, assignment, completion status and deadline in detail, and each task is associated with a specific user, which simulates the real project management process; the "Document Operation Table" tracks about 500,000 uploads, downloads, edits and sharing of documents, and records the document type, time of operation, operator, and collaborative partners; the "Meeting Arrangement Table" records 20,000 meetings, and the "Meeting Schedule Table" records the basic information of employees from different departments, positions, and workplaces. The "Meeting Schedule" records details of 20,000

ISSN 2616-5775 Vol. 8, Issue 1: 13-18, DOI: 10.25236/AJCIS.2025.080103

meetings, including meeting topics, time, location, participants, and minutes to reflect the communication and collaboration patterns within the enterprise; the "Communication Record Sheet" collects a total of 200,000 instant messages and email exchanges between users, demonstrating the diverse ways of communication. The "communication log table" collects 200,000 instant messages and emails between users, which shows the diversified ways of communication. We simulate the encryption and secure storage of the data using the three methods to compare the actual performance of the different methods.

5.2 Experimental results

The experiment uses throughput of different methods as a comparative indicator to measure the actual storage performance of the algorithm. The specific experimental results are shown in Table 1.

Table1: Comparison results of throughput rate under different methods

Experimental conditions	Method Throughput (MB/s)	Method A Throughput (MB/s)	Method B Throughput (MB/s)
Benchmarking (no load)	1,200	800	900
Single large file read/write	1,100	750	850
Multi-file concurrent reads and writes (100 files)	900	600	700
High-load scenario (simulates highly concurrent access)	800	500	600
Distributed storage mode (4 nodes)	4,000	2,500	3,000
SSD solid state disk mode	6,000	4,000	5,000
Data redundancy and fault tolerance mechanism on	1,050	700	800

According to the analysis of the data in Table 1, under benchmark test conditions, the throughput of our method is 1200MB/s, the throughput of Method A is 800MB/s, and the throughput of Method B is 900MB/s; Under the condition of reading/writing a single large file, the throughput of our method is 1100MB/s, the throughput of method A is 750MB/s, and the throughput of method B is 850MB/s; Under the condition of concurrent read and write of multiple files, the throughput of our method is 900MB/s, the throughput of method A is 600MB/s, and the throughput of method B is 700MB/s; In high load scenarios, the throughput of our method is 800MB/s, method A has a throughput of 500MB/s, and method B has a throughput of 600MB/s; In distributed storage mode, the throughput of this method is 4000MB/s, the throughput of method A is 2500MB/s, and the throughput of method B is 3000MB/s; In SSD solid state drive mode, the throughput of this method is 6000MB/s, the throughput of method A is 4000MB/s, and the throughput of method B is 5000MB/s; The throughput of our method under data redundancy and fault tolerance mechanisms is 1050MB/s, method A has a throughput of 700MB/s, and method B has a throughput of 800MB/s; The above experimental results show that the method in this paper is still able to maintain high throughput in complex scenarios such as concurrent reading and writing of multiple files, high load scenarios, and distributed storage modes. This proves the stability and efficiency of this paper's method in handling complex computing and storage tasks. By optimizing the algorithm and architectural design, the method in this paper can effectively manage resources and reduce resource competition and waiting time, thus improving the overall performance of the system.

6. Conclusion

This study focuses on homomorphic encryption-based secure cloud storage of mobile office data, and discusses in depth how to effectively guarantee the security and privacy of mobile office data in the process of storage and transmission in the cloud under the background of the current informationization era. Through the systematic research and innovative application of homomorphic encryption, we have

ISSN 2616-5775 Vol. 8, Issue 1: 13-18, DOI: 10.25236/AJCIS.2025.080103

not only enriched the theoretical foundation of mobile office security, but also provided practical solutions to data security challenges in practice.

References

[1] Song J, Chang J .General construction of compressive integrity auditing protocol from strong homomorphic encryption scheme[J].Cluster Computing, 2024, 27(5):5663-5675.
[2] Wegner T, Lassnig M, Ueberholz P, et al. Simulation and Evaluation of Cloud Storage Caching for Data Intensive Science[J].Computing and Software for Big Science, 2022, 6(1):1-17.
[3] Arasan K K, Anandhakumar P .Hybrid COOT-Reverse Cognitive Fruit Fly Optimization-Based Big Data Services and Virtual Machine Allocation for Cloud Storage System[J].Journal of circuits, systems and computers, 2024, 33(1):1.1-1.31.
[4] Liu Y .A Stable Cloud Storage Algorithm for Online Interaction Effect Data based on HarmonyOS[J]. 2023 Third International Conference on Artificial Intelligence and Smart Energy (ICAIS), 2023:1117-1121.
[5] Pawar A, Ghumbre S, Jogdand R .Study and Analysis of Various Cloud Security, Authentication, and Data Storage Models: a Challenging Overview[J].Int. . J. Decis. Support Syst. Technol. 2023, 15:1-16.
[6] Ekwonwune E N, Chigozie U C, Ekekwe D A, et al. Analysis of Secured Cloud Data Storage Model for Information[J]. Applications, 2024, 17(5):297-320.
[7] Boomija M D, Raja S V K .Securing medical data by role-based user policy with partially homomorphic encryption in AWS cloud[J].Soft computing: a fusion of foundations, methodologies and applications, 2023, 27(1):559-568.
[8] Guo X, Wang B, Jiang Y, et al. Homomorphic encryption based privacy-aware intelligent forwarding mechanism for NDN-VANET[J].Comput. Sci. Inf. Syst. 2023, 20:1-24.
[9] Wu X, Yu F, Wang J, et al. Bpf-payment: fair payment for cloud computing with privacy based on blockchain and homomorphic encryption[J]. Networking and Applications, 2023, 16(5):2649-2666.
[10] Zhang Q Y, Wen Y W, Huang Y B, et al. Secure speech retrieval method using deep hashing and CKKS fully homomorphic encryption[J].Multimedia Tools and Applications, 2024, 83(26): 67469-67500.
[11] Petrean D E, Potolea R .Random forest evaluation using multi-key homomorphic encryption and lookup tables[J]. Information Security, 2024, 23(3):2023-2041.
[12] Zhou T, Liu W, Li N, et al. Secure Scheme for Locating Disease-Causing Genes Based on Multi-Key Homomorphic Encryption[J]. Technology, 2022, 27(2):333-343.
[13] Shi J, Zhao X .Anti-leakage method of network sensitive information data based on homomorphic encryption[J].Journal of Intelligent Systems, 2023, 32(1):2517-39.
[14] Salanitro M, Penzel T, Rosenblum L, et al. Hybrid homomorphic encryption: the future of privacy-preserving data analytics and machine learning in sleep medicine?(HARPOCRATES)[J].Sleep Medicine, 2024, 115:107-108.
[15] Lin C P, Wu Z, Liu C H .Privacy Protection Scheme for Personal Health Record System Using Blockchain Based on Homomorphic Encryption[J].2023 IEEE 6th Eurasian Conference on Educational Innovation (ECEI), 2023:212-215.