**Frontiers in Computing and Intelligent Systems**
ISSN: 2832-6024 | Vol. 2, No. 1, 2022

# Homomorphic computing of encrypted data outsourcing in cloud data center

**Hao Bao**

College of Electronic and information, Southwest Minzu University Chengdu 610041, Sichuan, China

**Abstract:** In the era of data explosion, data contains massive information, such as health data, time and place, hydrological waves, etc. In order to process and calculate these data, local Wang networking devices will send data to the cloud data center for outsourcing processing due to their limited storage and computing capabilities. However, our data contains a large amount of private data, so we need to protect the privacy of our outsourced data before outsourcing, so as to protect our personal privacy. At the same time, cloud data centers have strong advantages in data storage and computing capabilities, so cloud data centers are increasingly used.

**Keywords:** Cloud data center; Data encryption; Data outsourcing; Homomorphic computing.

## 1. Introduction

With the progress of Internet of Things (IoT) technology, more and more IoT devices are deployed in an application, such as smart grid, intelligent transportation and smart city. According to the prediction released by IDC, there will be 41.6 billion IoT devices in 2025, which can cooperate to provide us with better life and work experience. For example, in an intelligent transportation environment, networked vehicles can be driven more safely and comfortably. The smart city project, which combines information technology with Internet connection equipment, has extensively strengthened all aspects of municipal management worldwide, from waste management to smart grid. Intelligent electronic medical technology mainly integrates wearable medical devices, the Internet of Things and mobile Internet into traditional medical services, bringing great convenience to patients' intelligent disease diagnosis and medical monitoring.

However, because the cloud data center is not reliable and may leak the data we outsourced to the cloud data center, we need to encrypt our private data before data outsourcing. The encryption schemes that can be applied in the cloud data center are: FHE, SWHE and PHE. The main purpose is to realize that our plaintext calculation results in the ciphertext form are no different from the plaintext direct calculation results.

## 2. System model

The system model mainly includes four aspects: authorities, users, cloud data centers, and query users.

Authority: Users, cloud data centers and query users will send requests to the authority and get authorization from the authority. Only users can obtain the granted key, which is used to encrypt their private data, and can send the private data to the cloud data center for further processing and calculation. The cloud data center can obtain the key sent by the authority after being authorized by the authority, and can decrypt and compare the calculation results of the ciphertext. The query user is authorized by the authority to obtain a key to encrypt his query and send it to the cloud data center for query.

Users: Local users can encrypt their private data after obtaining the encrypted data from the authority. At the same time, we assume that all values in the dataset are integers. If they were not originally integers, we could easily convert them to integers. Because the data owner has limited computing power and storage space, it outsources the data set to the cloud for subsequent calculations.

Cloud data center: The cloud data center can generally be composed of 1-2 cloud servers, which can cooperate to save and process data from users. Generally, all encrypted data are calculated by an ECS, and random numbers are added to keep the real calculation results confidential. The other server is responsible for decrypting and comparing the calculation results.

Query user: After being authorized, the query user can encrypt his/her query request and send the encrypted query request to the ECS, which will return the calculated results to the query user.

## 3. Security Model

In our security system, the user is trusted because he is the initiator of the whole system, and all calculations are performed on this basis. For query users, they will honestly send query requests to the ECS and will not collude with the ECS. The two ECS servers are considered to be semi honest and semi trusted. They will not only keep their user data confidential, but also be curious about this information. This is feasible in real life, because each ECS provider will keep users' data confidential in order to maintain goodwill. Because different cloud service providers may have conflicts of interest, there will be no linkage between the two ECS providers. Therefore, ECS will faithfully store encrypted data for data owners and provide similarity query services for query users.

## 4. Homomorphic calculation

### 4.1. Paillier encryption system

The Paillier cryptosystem is a broadly used probabilistic public-key cryptographic scheme that supports homomorphic operations. It has the following three parts.

Key Generation: Given a security parameter $K$ ( $Z^+$, large prime numbers p and q of length K are randomly and independently selected. Let $n = pq$ and $\lambda = \mathrm{lcm}(p-$

$1, q-1)$. Choose a random g ( $Z_n$ such that $\mu = L(g^{\lambda} \bmod n^2)^{-1}$ exists, where function $L(x) = (x-1)/n$. At the end of the algorithm, it yields the public key $pk = (n, g)$ and the private key $sk = (\lambda, \mu)$.

Encryption: Given a message m ( $Z_n$, the ciphertext can be computed with the public key pk as $c = E_{pk}(m) = g^m \cdot r^n \bmod n^2$, where r is a random number in $Z_n$.

Decryption: With the private key sk, a ciphertext $c = E_{pk}(m)$ can be decrypted by computing $m = PDec(c) = L(c^{\lambda} \bmod n^2) \cdot \mu \bmod n$.

The Paillier password system has the following attributes:

Homomorphic addition: Given two ciphertext $E(m_1), E(m_2)$ ( $Z_n, m_1 + m_2 < n$, we have $E(m_1 + m_2) = E(m_1) + E(m_2)$.

Homomorphic multiplication: Given a ciphertext $E(m_1)$ and a plaintext $m_2$ ( $Z_n, m_1 \cdot m_2 < n$, we have $E(m_1 \cdot m_2) = E(m_1)^{m_2}$.

Self-blindness: Given a ciphertext $E(m_1)$, By calculating $E(m_1) \cdot r^n \bmod n^2$, we can change it to another ciphertext with the same content, random number r ( $Z_n$ .

### 4.2. Symmetric Homomorphic Encryption

SHE is an effective homomorphic encryption, which uses the same key in both encryption and decryption. Generally, SHE includes three algorithms, namely: 1) Key generation KeyGen(); 2) Encrypt Enc(); 3) Decrypt Dec() as follows.

KeyGen( $k_1, k_2, k_3$ ): Given security parameters $\{k_1, k_2, k_3\}$ satisfying $k_1 < k_2 < k_3$, the key generation algorithm first chooses two large prime numbers $p, q$ with $|p| = |q| = k_0$, and sets $N = pq$. Then, it selects a random number L with $|L| = k_2$. Finally, the algorithm outputs the public key pk = ($k_0$, $k_1$, $k_2$, N ), the secret key $sk = (p, L)$, and the message space $M = \{m | m ( (-2^{k_1}, 2^{k_1})\}$.

$Enc(sk, m)$: Taking a secret key sk and a message m ( M as inputs, the encryption algorithm outputs the ciphertext $E(m) = (rL + m)(1 + r'p) \bmod N$, where r ( $\{0,1\}^{k_2}$ and r' ( $\{0,1\}^{k_0}$ are random numbers.

$Dec(sk, E(m))$: On inputting a secret key sk and a ciphertext E(m), the decryption algorithm recovers a message $m' = (E(m) \bmod p) \bmod L = (m + rL) \bmod L$. If $m' < (L/2)$, it indicates $m \geq 0$ and $m = m'$. Otherwise, $m < 0$ and $m = m - L$.

SHE technology satisfies the properties of homomorphic addition and multiplication as follows.

Homomorphic addition-I:Given two ciphertext $E(m_1), E(m_2), E(m_1) + E(m_2) = E(m_1 + m_2)$ is satisfied.

Homomorphic addition-II:Given a ciphertext $E(m_1)$ and a plaintext $m_2, E(m_1) + m_2 = E(m_1 + m_2)$ is satisfied.

Homomorphic multiplication-I:Given two ciphertext $E(m_1)$ and $E(m_2), E(m_1) * E(m_2) = E(m_1 * m_2)$ is satisfied.

Homomorphic multiplication-II:Given a ciphertext $E(m_1)$ and a plaintext $m_2, E(m_1) * m_2 = E(m_1 * m_2)$ is satisfied.

## 5. Conclusion

In this paper, we mainly introduce two homomorphic computing schemes, namely pailliar encryption and SHE encryption, which are used for data processing and computing in outsourcing to the cloud data center.

## References


[1] Y. Guan, R. Lu, Y. Zheng, S. Zhang, J. Shao, and G. Wei, "Toward Privacy-Preserving Cybertwin-Based Spatio-Temporal Keyword Query for ITS in 6G Era", IEEE Internet of Things Journal, Vol. 8, No. 22, pp. 16243-16255, 2021.

[2] J. Jiao, S. Wu, R. Lu, and Q. Zhang, "Massive Access in Space-based Internet of Things: Challenges, Opportunities, and Future Directions," IEEE Wireless Communications, Vol. 28, No. 5, pp. 118-125, 2021.

[3] M. Zhou, Y. Zheng, Y. Guan, L. Peng, and R. Lu, "Efficient and Privacy-Preserving Range-Max Query In Fog-based Agriculture IoT", Peer-to-Peer Networking and Applications, Vol. 14, No. 4, pp. 2156-2170, 2021.

[4] J. Jiao, L. Xu, S. Wu, R. Lu, and Q. Zhang, "MSPA: Multi-Slot Pilot Allocation Random Access Protocol for mMTC-enabled IoT System", IEEE Internet of Things Journal, Vol. 8, No. 24, pp. 17403-17416, 2021.

[5] X. Yang, H. Zhu, F. Wang, S. Zhang, R. Lu, and H. Li, "MASK: Efficient and Privacy-Preserving M-tree Based Biometric Identification over Cloud", Peer-to-Peer Networking and Applications, Vol. 14, No. 4, pp. 2171-2186, 2021.

[6] X. Zhang, R. Lu, J. Shao, H. Zhu, and A. Ghorbani, "Continuous Probabilistic Skyline Query for Secure Worker Selection in Mobile Crowdsensing", IEEE Internet of Things Journal, Vol. 8, No. 14, pp. 11758-11772, 2021.

[7] P. Zeng, Z. Zhang, R. Lu, and K. Choo, "Efficient Policy-Hiding and Large Universe Attribute-Based Encryption with Public Traceability for Internet of Medical Things", IEEE Internet of Things Journal, Vol. 8, No. 13, pp. 10963-10972, 2021.

[8] X. Li, S. Liu, R. Lu, and X. Zhang, "On Security of An Identity-Based Dynamic Data Auditing Protocol for Big Data Storage", IEEE Transactions on Big Data, Vol. 7, No. 6, pp. 975-977, 2021.

[9] Q. Kong, R. Lu, Y. Feng, and S. Cui, "Privacy-Preserving Continuous Data Collection for Predictive Maintenance in Vehicular Fog-Cloud", IEEE Transactions on Intelligent Transportation Systems, Vol. 22, No. 8, pp. 5060-5070, 2021.

[10] K. Zhang, M. Wen, R. Lu, and K. Chen, "Multi-client Sub-Linear Boolean Keyword Searching for Encrypted Cloud Storage with Owner-Enforced Authorization", Transactions on Dependable and Secure Computing, Vol. 18, No. 6, pp. 2875-2887, 2021.

[11] X. Yang, R. Lu, J. Shao, X. Tang, and A. Ghorbani, "Achieving Efficient and Privacy-Preserving Multi-Domain Big Data Deduplication in Cloud", IEEE Transactions on Services Computing, Vol. 14, No. 5, pp. 1292-1305, 2021.

[12] Q. Kong, R. Lu, F. Yin, and S. Cui, "Blockchain-Based Privacy-Preserving Driver Monitoring for MaaS in the Vehicular IoT", IEEE Transactions on Vehicular Technology, Vol. 70, No. 4, pp. 3788-3799, 2021. [PDF]

[13] Y. Zhan, B. Wang, R. Lu, and Y. Yu, "DRBFT: Delegated Randomization Byzantine Fault Tolerance Consensus Protocol for Blockchains", Information Science, Vol. 559, pp. 8-21, 2021. [PDF]

[14] S. Zhang, S. Ray, R. Lu, Y. Zheng, and J. Shao, "Preserving Location Privacy for Outsourced Most-Frequent Item Query in Mobile Crowdsensing", IEEE Internet of Things Journal, Vol. 8, No. 11, pp. 9139-9150, 2021. [PDF]

[15] F. Wang, H. Zhu, R. Lu, Y. Zheng, and H. Li, "A Privacy-preserving and Non-interactive Federated Learning Scheme for Regression Training with Gradient Descent", Information Sciences, Vol. 552, pp. 183-200, 2021. [PDF]

[16] B. Hu, Z. Zhang, J. Liu, Y. Liu, J. Yin, R. Lu, and X. Lin, "A Comprehensive Survey on Smart Contract Construction and Execution: Paradigms, Tools and Systems", Patterns, Vol. 2, No. 2, pp. 100179, 2021. [PDF]

[17] J. Wang, S. Hao, R. Wen, B. Zhang, H. Hu, and R. Lu, "IoT-Praetor: Undesired Behaviors Detection for IoT Devices", IEEE Internet of Things Journal, Vol. 8, No. 2, pp. 927-940, 2021.