Data privacy protection based on homomorphic encryption

Yarong Lv1,*

1Information Engineering College, Xi’an Mingde Institute of Technology, Xi’an, 710000, Shaaxi, China

*Corresponding author e-mail: fannaiqiang@nwpu.edu.cn

Abstract. Data security and privacy protection are increasingly becoming a bottleneck in the development of cloud computing. To improve the existing homomorphic encryption technology, ECC homomorphic encryption algorithm is proposed, and ECC homomorphic encryption algorithm is designed and implemented. Based on this, the aggregation model of data privacy protection is constructed, and the effect of ECC homomorphic encryption data security operation is analyzed. Compared with the traditional RSA encryption algorithm, ECC homomorphic encryption algorithm shows a higher level of security. When the key length increases gradually, the security of ECC with more than 300 bits is much greater than that of RSA with more than 2000 bits. At the same time, ECC homomorphic encryption algorithm is better than RSA in terms of encryption time and cipher size. ECC homomorphic encryption algorithm has the advantages of low computational complexity and good security performance.

Keywords: Homomorphic Encryption, Ecc Encryption, Cloud Computing, Privacy Protection

1. Introduction

In recent years, information technology has achieved rapid development, cloud computing has been more and more applications, homomorphic encryption technology is also widely used in cloud computing and cloud storage [1]. However, with the development of cloud computing, the problem of data privacy protection has become increasingly prominent, which has become an important security problem to be solved. On the other hand, the calculation process of homomorphic encryption technology based on public key cryptosystem is more complex, unable to cope with the situation of large amount of calculation and numerous users, and there is no cloud center that can be trusted [2-3]. Therefore, ECC homomorphic encryption method is used to protect the privacy of data, which effectively protects the privacy of users during data aggregation. What cloud center obtains is the final result of aggregation, and the privacy data of each user is unknown, which ensures the data security of users. Compared with RSA encryption method, ECC homomorphic encryption algorithm has some advantages in data privacy protection and computing performance, which shows that ECC homomorphic encryption algorithm can effectively protect the data security of users.

2. ECC homomorphic encryption data security operation

As a kind of encryption function, homomorphic encryption refers to the encryption function with homomorphic property, which mainly carries out the operation of addition and multiplication on the
plaintext ring, and carries out the corresponding operation on the cipher after encryption, so as to obtain the equivalent result [4]. Specifically, selecting any two plaintexts \( m_1 \) and \( m_2 \), can satisfy the encryption function \( \text{Dec}(\text{En}(m_1) \otimes \text{En}(m_2)) = m_1 \oplus m_2 \). where \( \text{En} \) represents encryption operation, \( \text{Dec} \) represents decryption operation, \( \otimes \) and \( \oplus \) correspond to the operation on ciphertext and plaintext one by one respectively. When \( \otimes \) and \( \oplus \) denote addition, the encryption function refers to additive homomorphic encryption; when \( \otimes \) and \( \oplus \) denote multiplication, it is multiplicative homomorphic encryption [5-6].

As a traditional homomorphic encryption scheme, RSA encryption is a typical multiplicative homomorphic encryption scheme, which mainly includes five parts: public key, private key, plaintext, encryption and decryption. Because this encryption scheme can only satisfy multiplicative homomorphism, but not additive homomorphism, the operation based on this homomorphic encryption scheme becomes semi-homomorphic encryption scheme [7]. On the other hand, the traditional homomorphic encryption technology is mainly based on public key cryptosystem in practical application, which has high complexity and obviously can not cope with the situation of large amount of calculation and large number of users [8]. After improving the existing homomorphic encryption technology, based on the homomorphic method of Elliptic Curve Cryptography (ECC), a homomorphic encryption data privacy protection scheme is designed.

Elliptic curve encryption mechanism usually encodes the plaintext points on the elliptic curve during encryption, and decodes the points on the elliptic curve into plaintext information during decryption [9-10]. The following two public key system based on elliptic curve encryption method is introduced. Method 1: firstly, a random number \( k \) is selected, the order of base point \( G \) is \( n \), satisfying \( k \leq n \). after calculation, \( K = kG \) can be obtained. As public key, \( m \) in plaintext coding is transferred to a point \( P_m \) on elliptic curve \( E \). then, a random number \( r \) is arbitrarily selected, and formulas (1) and (2) can be obtained by calculation.

\[
\begin{align*}
C_1 &= rG \\
C_2 &= rK + P_m
\end{align*}
\]

\( (C_1, C_2) \) is the cipher. When decrypting, \( (C_1, C_2) \) needs to be encrypted with the private key \( k \). formula (3) can be obtained by calculation.

\[
C = kC_1
\]

\( kG_1 = k(rG) = rK \) is obtained from the above formula (1). After further calculation, \( B \) is obtained:

\[
C_2 - C = rK + P_m - rK = P_m
\]

After decoding \( P_m \), we can get the plaintext message \( m \).

Method 2: in the same method 1, the calculation method of \( C_1 \) is used to obtain \( C_2 = rK \) and \( C_3 = mC_2 \), where \( (C_1, C_3) \) is the cipher, and the private key \( k \) is used to decrypt \( (C_1, C_3) \). the calculation results are shown in formula (5).

\[
\begin{align*}
kC_1 &= k(rG) = rK = C_2 \\
C_2C_2^{-1} &= m
\end{align*}
\]

So get the plaintext message \( m \).

ECC public key encryption system can realize homomorphic addition and homomorphic multiplication. First of all, we need to add and multiply \( m_1, m_2, \ldots m_n \) of \( n \) data. In the aspect of homomorphic addition, select point \( P_{m_i} \) on elliptic curve \( E \) to encode plaintext \( m_i \), select a
random number \( r_i \) from it, and calculate \( C_i \) and \( C_j \) through formula (1) and formula (2), so as to obtain encrypted data \( \left( C_1, C_2 \right) \). After adding each item \( \left( C_1, C_2 \right) \cdots \left( C_n, C_n \right) \), we get

\[
\left( \sum_{i=1}^{n} C_i, \sum_{i=1}^{n} C_i \right).
\]

By using the private key \( k \), \( C = k \sum_{i=1}^{n} C_i \) is obtained from formula (3) and further proved that:

\[
k \sum_{i=1}^{n} C_i = kG \sum_{i=1}^{n} r_i = k \sum_{i=1}^{n} r_i
\]  

(7)

It is calculated by formula (4).

\[
\sum_{i=1}^{n} C_2 + C = K \sum_{i=1}^{n} r_i + \sum_{i=1}^{n} P_{m_i} - K \sum_{i=1}^{n} r_i = \sum_{i=1}^{n} P_{m_i}
\]  

(8)

So get the sum \( \sum_{i=1}^{n} P_{m_i} \), and after decoding again, get \( \sum_{i=1}^{n} m_i \).

In the aspect of realization of homomorphic multiplication, first select plaintext information \( m_i \), get \( C_1, C_2, C_3 \) through calculation, then get all encrypted data, \( \left( C_1, C_2 \right) \cdots \left( C_n, C_n \right) \). Multiply the data, then get \( \left( C_1 \cdot C_1 \cdots C_{i_n}, C_2 \cdot C_2 \cdots C_{i_n} \right) \). Combining with the private key \( k \),

\[
k^G \cdot C_1 \cdot C_1 \cdots C_n \text{ is calculated from formula (5), which further proves that:}
\]

\[
k^G \cdot C_1 \cdot C_1 \cdots C_n = k^G r_1 \cdot r_2 \cdots r_n = k^G r_1 \cdot r_2 \cdots r_n = C_2 \cdot C_2 \cdots C_n
\]

\[
C_3 \cdot C_3 \cdots C_n, C_4 \cdot C_4 \cdots C_n \text{ is calculated by formula (6), and}
\]

\[
m_1 \cdot m_2 \cdots m_n \text{ is obtained.}
\]

Based on ECC homomorphic encryption technology, the aggregation model of data security is constructed. As shown in Figure 1, the data are respectively \( A_1, A_2, \ldots, A_n \), as the operation center B, mainly as the data receiving end and aggregation point. Through the private key decryption, B obtains the final aggregation result as \( \sum_{i=1}^{n} m_i \), but the private key decryption cannot obtain the plaintext data \( m_i \) sent by each user \( A_i \). Therefore, when B is an untrusted node, on the one hand, it can ensure the smooth operation of aggregation, on the other hand, it can also fully guarantee the data privacy security of each user \( A_i \).
3. ECC homomorphic encryption data security operation effect analysis

The data privacy protection of this algorithm is mainly reflected in the security of cipher processing and ECC algorithm. The security of ECC algorithm is the basis of other security. Compared with RSA and other public key systems, in terms of security, ECC algorithm has significant advantages in terms of security. At the same time, ECC algorithm does not rely on trusted cloud center, so it can achieve secure cloud storage, cloud computing and data privacy protection. As shown in Figure 2, it shows the key lengths of RSA and ECC under different security levels. A, B, C, D and E represent five different security levels. The security level of RSA at 1024 bits is consistent with that of ECC at more than 100 bits. When the key length increases gradually, the security gap between RSA and ECC increases gradually. When the ECC algorithm has a key length of more than 300 bits, its security is far greater than that of RSA with more than 2000 bits. Compared with RSA, ECC encryption algorithm has higher security, and can effectively ensure the data privacy of users.

![Figure 1. Security data aggregation](image)

![Figure 2. Security comparison of three encryption algorithms](image)

As an important index of homomorphic encryption algorithm, the computational performance can further measure the pros and cons of cryptographic system, and test the computational performance of RSA and ECC. As shown in Figure 3, with the increase of plaintext, the encryption time and cipher size of ECC and RSA are on the rise, and the operation efficiency of ECC is much higher than that of RSA. As can be seen from Figure 3 (a), when comparing the encryption time of ECC and RSA with the same key length, except for the encryption time of 1 byte plaintext, the overall ECC encryption time is significantly less than that of RSA. Specifically, when the size of RSA and ECC cipher is 32...
bytes, the encryption time is 0.3s and 0.1s respectively, and the cipher size of RSA increases rapidly after 256 bytes. When the size of ECC and ECC cipher is 512 bytes, the encryption time is 3.0s and 0.4s respectively. On the whole, the encryption time of ECC is much lower than that of RSA, and the running time is shorter and more efficient. As can be seen from Figure 3 (b), comparing the cipher size of ECC and RSA with the same key length, the cipher size of ECC is obviously smaller than that of RSA.

**Figure 3.** Comparison of RSA and ECC encryption time and cipher size

As shown in Figure (4), the influence of calculation parameters on encryption and decryption performance. As can be seen from Figure 4 (a), when the key length is 40 bits, the encryption time and decryption time are 0.22s and 0.14s respectively, and when the key length is 90 bits, the encryption time and decryption time are 0.42s and 0.23s respectively. Therefore, with the increase of the key length, the decryption time and encryption time show a steady upward trend. When the key length is the same, the encryption time is much longer than the decryption time. When the key length is 100 bits, the difference between the two is the largest. As can be seen from Figure 4 (b), when the length of the finite field is 100 bits, the encryption time and decryption time are 0.18s and 0.14s respectively. When the length of the finite field is 200 bits, the encryption time and decryption time are 0.29s and 0.18s respectively. With the increase of the length of the finite field, although it decreases in a certain length of the finite field, it is in a steady upward trend, which takes more time to encrypt than to secret. Through the comparison of the same plaintext size, we can see that the key length and finite field length can have a certain impact on the encryption and decryption time.

**Figure 4.** Influence of Key length and Limited domain on encryption and decryption time

4. **Conclusion**

The large-scale use of cloud computing not only facilitates people's life, but also brings some security threats to users' privacy. This paper studies and analyzes the existing homomorphic encryption methods, proposes ECC homomorphic encryption algorithm, implements homomorphic addition and homomorphic multiplication in ECC public key encryption system, and constructs an aggregation model of data security, which is used for privacy protection in data operation. Compared with the traditional RSA encryption algorithm, ECC homomorphic encryption algorithm has a higher security level. With the gradual increase of the key length, the security gap between RSA and ECC gradually increases. When ECC algorithm has a key length of more than 300 bits, its security is far greater than
that of RSA with more than 2000 bits, which further shows that ECC homomorphic encryption algorithm can effectively protect the data security of users. At the same time, ECC encryption time and cipher size are significantly smaller than RSA, which has good computing performance. Although the design of ECC homomorphic encryption algorithm has achieved some results in cloud computing, there are still some shortcomings. It is expected to further verify the security of data storage on the basis of this research and better optimize the algorithm, which is also a direction of future research.

References
[1] B J L A , C X K , A S L , et al. Privacy preservation for machine learning training and classification based on homomorphic encryption schemes[J]. Information Sciences, 2020, 526:166-179.
[2] Jing L. Design of Privacy Protection Electronic Voting Scheme Based on ElGamal Homomorphic Encryption[J]. Computer Science and Application, 2019, 09(5):941-946.
[3] Lu Y , Zhou T , Tian Y , et al. Web-Based Privacy-Preserving Multicenter Medical Data Analysis Tools Via Threshold Homomorphic Encryption: Design and Development Study[J]. Journal of Medical Internet Research, 2020, 22(12):e22555.
[4] Fenner P , Pyzer-Knapp E O . Privacy-Preserving Gaussian Process Regression -- A Modular Approach to the Application of Homomorphic Encryption[J]. Proceedings of the AAAI Conference on Artificial Intelligence, 2020, 34(4):3866-3873.
[5] Mohammed M A , Abed F S . Cloud Storage Protection Scheme Based on Fully Homomorphic Encryption[J]. ARO-The Scientific Journal of Koya University, 2020, 8(2):40-47.
[6] Kiran A , Puri M , Srinivasa S . Privacy Preserving Model using Homomorphic Encryption[J]. International Journal of Computer Applications, 2019, 182(38):12-18.
[7] Ren W , Tong X , Du J , et al. Privacy-preserving using homomorphic encryption in Mobile IoT systems[J]. Computer Communications, 2021, 165:105-111.
[8] Park J , Kim D S , Lim H . Privacy-Preserving Reinforcement Learning Using Homomorphic Encryption in Cloud Computing Infrastructures[J]. IEEE Access, 2020, 8:203564-203579.
[9] Zhang L , Zhang J , Hu Y H . A Privacy-Preserving Distributed Smart Metering Temporal and Spatial Aggregation Scheme[J]. IEEE Access, 2019, 7:28372-28382.
[10] El-Ansari A , Hssane A B , Saadi M . An Enhanced Privacy protection scheme for Profile-based personalized search[J]. International Journal of Advanced Trends in Computer Science and Engineering, 2020, 9(3):4128-4136.