International Journal for Housing Science and Its Applications
Publish August 4, 2025. Volume 46, Issue 3 Pages 2191-2201

https://doi.org/10.70517/ijhsa463174

Research on Data Encryption Algorithm Optimization and Information Security Protection Strategy for Cloud Computing Environment

Pengfei Wang[1,*], Yanan Lv[1] and Xiaoyun Zheng[1]
[1] Information Center, Hebei Baisha Tobacco Co., Ltd., Shijiazhuang, Hebei, 050000, China
Corresponding authors: (e-mail: 15610985892@163.com).

Abstract Traditional encryption and decryption algorithms have high performance overheads in cloud computing environments, and it is difficult to truly balance security and operational efficiency. This paper takes optimizing data encryption and decryption performance and strengthening data information security as the research objectives. After completing the definition of security space text similarity connection and related issues, the cloud computing system model is proposed. Then, combined with the cloud computing environment, the lattice core encryption and decryption algorithm consisting of key generation algorithm, core encryption algorithm and core decryption algorithm is designed. The algorithm ensures that users can perform complex and correct encryption and decryption calculations under limited resources by using fog nodes to bear most of the computational overhead for users. After combining the lattice core encryption and decryption algorithm with the cloud computing model to obtain the data encryption algorithm, a security model is set up to ensure the security of private data such as data information, query information, and result information. The data encryption algorithm in this paper takes only 3.75s and 4.13s to encrypt and decrypt compared to similar algorithms, which is the most efficient encryption and decryption.

Index Terms lattice core encryption and decryption algorithm, security model, cloud computing system model, fog node

I. Introduction

Cloud computing is an emerging technology that provides computing services, storage services and other services to users over a network, where users do not need to purchase, configure and maintain physical devices, but only need to purchase and access computing resources, storage resources and other services on-demand over the Internet. Cloud computing has three major advantages which are scalability cost effectiveness, flexibility and easy management and maintenance [1]-[3]. As the use of cloud computing becomes more and more widespread, the trustworthiness and security of the data on the cloud is the focus of users' attention when utilizing cloud computing for computation and storage services, and trustworthy and secure cloud computing services are one of the current obstacles restricting the development of cloud computing [4]-[6]. In the traditional model, the computation and storage of data on the cloud are in the hands of the cloud service provider. The ownership of the user's data exists in name only, and all the user's data and operations depend on the centralized platform of cloud servers, which means that some attackers can illegally access and dominate the data on the centralized platform by attacking it [7]-[10]. To summarize, data on the cloud brings certain security problems due to management issues, and users face risks such as data privacy, integrity and security.

Cloud computing allows data owners to transfer shared data and applications to third-party servers for storage, which can cause the data owner to completely lose ownership of the data and physical control over the resources, which can be a huge security risk [11]-[14]. On the one hand, data owners store their private information in the cloud, which can lead to the leakage of user information in case of hacker's attack on the security holes existing in the cloud server [15]. In 2018 the media reported that two hackers had successfully compromised some Chromecast end devices connected to the Internet, exposing smart devices such as Chromecast and Google Homes to the public interconnected network. On the other hand, if the third-party servers go through users' private information to obtain improper benefits, such as the trafficking of users' private information and the projecting of users' consumption habits through machine learning algorithms, so as to carry out purposeful commercial advertisement pushing [16], [17]. 2021 On June 3, 2021, by the Shangqiu Court in China to make public a criminal verdict, a criminal suspect through the software he developed illegally Obtained about 1.18 billion Taobao users' account information, and packaged and sent the user information therein to others for marketing as commercial information, and ultimately made a profit of 340,000 yuan.

In recent years, the successive impact of cloud servers is relatively large data leakage caused by information security events, but also inevitably let a lot of people have a crisis of confidence in cloud computing, but also the same many enterprises and individuals sounded the alarm of data security in the cloud, the optimization of encryption technology is imperative [18]. When dealing with large-scale datasets, conventional data encryption techniques may encounter performance limitations that result in encryption activities taking too much time for use environments with stringent demands for immediate response. The security of some encryption techniques is also challenged with the possibility of decryption as computational power increases and cryptography research advances [19], [20]. In addition, due to the single source of traditional data, the amount of data to be encrypted, stored, managed and analyzed is relatively small, and can be handled by a single machine or an encryption mechanism of low complexity. Compared with the parallelism of computing to improve the speed of data processing, traditional encryption technology pursues a high degree of integrity and confidentiality, and it is difficult to ensure the efficiency and real-time performance of data encryption [21]-[23]. In order to meet these challenges, it is necessary to continuously research and improve data encryption technology to strengthen information security.

In this paper, we first analyze in detail the meanings of secure space text data objects, relevance, similarity, and similarity connection in cloud computing environment. It also gives the corresponding cloud computing system model and briefly analyzes the interaction process of different components in the system model. Subsequently, the core encryption and decryption algorithm is introduced as the data encryption and decryption algorithm, and the key generation algorithm, core encryption algorithm, and core decryption algorithm in the framework are elaborated. At the same time, a security model is established to ensure the security of encrypted and decrypted data information in different scenarios. Finally, the effectiveness of the encryption and decryption algorithm and security model designed in this paper is verified in the form of comparison and simulation experiments.

II. Design of data encryption and decryption algorithms and security models

II. A. Definition of the problem

Definition 1 (Spatial textual data object): given a data object $D=(D.s, D.t)$ containing both spatial and textual attributes, $D.s=(x, y)$, where (x, y) is the longitude and latitude of D, respectively, and $D.t=\left(D.t_1, D.t_2, \ldots, D.t_{t_i}\right)$, where $\left(D.t_1, D.t_2, \ldots, D.t_{t_i}\right)$ is the set of keywords for D. For example $D=\left((123.38, 41.8),(\text{Shopping, golf})\right)$.

Definition 2 (spatial proximity): for two spatial text data objects D and Q, the spatial proximity is negatively correlated with their distances, calculated as in equation (1).

$$Sim_s(D,Q)=\max\left(0,1-\frac{dist(D.s,Q.s)^2}{dist_{\max}^2}\right) \tag{1}$$

where $dist(D.s,Q.s)^2$ is the square of the Euclidean distance between D and Q, and $dist_{\max}$ is a normalized coefficient, which can be specified as the maximum Euclidean distance in a particular space.

Definition 3 (Textual relevance): in this paper, we use Jaccard similarity to represent textual relevance, for two spatial textual data objects D and Q, their textual relevance is calculated as in equation (2).

$$Sim_t(D,Q)=\frac{|D.t\cap Q.t|}{|D.t\cup Q.t|} \tag{2}$$

where $|D.t\cap Q.t|$ is the base of the intersection of D and Q keyword sets, and $|D.t\cup Q.t|$ is the base of the concatenation of D and Q keyword sets.

Definition 4 (spatial textual similarity): spatial textual similarity is defined as the result of the combined computation of spatial proximity and textual relevance of two data objects, which is calculated as in Equation (3).

$$Sim_{s,t}(D,Q)=\alpha Sim_s(D,Q)+(1-\alpha)Sim_t(D,Q) \tag{3}$$

where α is a balance parameter between spatial similarity and textual relevance, which can be provided as a preference by the user.

Definition 5 (Spatial Text Similarity Connection): given two spatial text datasets $\mathrm{D}=\{D_1, D_2, \ldots, D_m\}$ and □ $=\{Q_1, Q_2, \ldots, Q_n\}$, notate the spatial text similarity join results of D and □ as a set R that contains all spatial text similarities in the full join of D and $\square$ that are greater than a threshold τ, i.e. $\mathrm{R} = \{\langle D,Q\rangle \mid \langle D,Q\rangle \in \mathrm{D}\times\square, Sim_{s,t}(D,Q)\geq\tau\}$, where τ is the threshold value set by the user.

For example, $\mathrm{D}=\{D_1,D_2,D_3\}$ and $\square=\{Q_1,Q_2\}$ are two spatial textual datasets, where each data object contains the location (spatial information) and hobbies (textual information) of a social media user. Assuming $\tau=0.7$, $dist_{\max}^2=9$, and $\alpha=0.5$, the similarity of all data pairs is computed using Eq. (3). Filtering by threshold τ, we can get $\langle D_3,Q_1\rangle$ as the result of spatial text similarity connection is shown in Table 1.

Table 1: An example of spatial text similarity calculation

Data pair	Distance	Text relevance	Spatial text relevance
$\langle D_1,Q_1\rangle$	2.2	0.34	0.38
$\langle D_1,Q_2\rangle$	3.2	0	0
$\langle D_2,Q_1\rangle$	3.6	0	0
$\langle D_2,Q_2\rangle$	3.2	1	0.5
$\langle D_3,Q_1\rangle$	1.4	0.66	0.71
$\langle D_3,Q_1\rangle$	2.2	0	0.21

Definition 6 (Secure Space Text Similarity Joining): Encrypting the datasets D and $\square$ gives $E_{pk}(\mathrm{D})$ and $E_{pk}(\square)$ in the secure space text similarity connection at $E_{pk}(\mathrm{D})$ and $E_{pk}(\square)$ All data pairs with similarity greater than the threshold τ, where $E_{pk}(\cdot)$ represents the encryption method of the spatial text dataset.

II. B.System modeling

In this paper, we adopt a system model that has been widely used in many works in recent years, which consists of three parts: the data owner (DO), the authorized user (U), and the cloud servers (C_1 and C_2), and the specific framework of the system model is shown in Figure 1. In order to minimize the interaction between DO and U and to ensure the privacy of the data and the data access pattern, the model uses two non-colluding cloud servers C_1 and C_2 as the platform for cloud computing. The data stored on C_1 are ciphertexts, and computations are performed between ciphertexts. C_2 has both a public key (pk) and a private key (sk).

The specific division of labor among the entities in each part of the model is as follows:

(1) Data owner

DO generates the key pair (pk,sk), encrypts the dataset and the constructed index structure, and uploads it to C_1. Distribute pk to C_1, C_2 and U, sk to C_2.

(2) User

U encrypts the connection request using pk distributed by DO and initiates a connection request to C_1.

(3) Cloud server

After receiving the encrypted dataset, encrypted index structure and encrypted connection request from DO and U, C_1 performs the secure spatial text similarity connection method with C_2 and returns the connection result to the user.

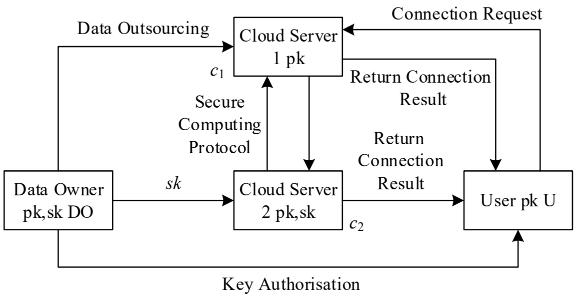

Figure 1: System model

II. C.Grid Core Encryption and Decryption Algorithm Construction

In this section, the core encryption and decryption algorithms are proposed based on secure space text similarity connection and related issues, including key generation algorithm, core encryption algorithm and core decryption algorithm.

II. C. 1) Key Generation Algorithm

The key generation algorithm will generate the public key (PK_0, PK_1) and the private key SK based on the secure-space text-similarity connection puzzle.

First the public parameters in the system need to be determined. Let n be a positive integer, p be a large prime number, and p satisfy $p \equiv 1 \bmod 2n$. $q \in Z_p$ is a positive integer and satisfies $q \ll p$, $q > 1$, and $\gcd(p,q) = 1$. Thus the ring can be determined as in Eqs. (4)-(5):

$$R_p = Z_p[x] / x^n + 1 \tag{4}$$

$$R = Z[x] / x^n + 1 \tag{5}$$

After that, the algorithm selects the error distribution $\chi \subset R_p$, which can be expressed as equation (6):

$$\chi = \Phi_\alpha(y) = \sum_{-\infty}^{+\infty} \frac{1}{\alpha} \exp\left(-\pi\left(\frac{y-k}{\alpha}\right)^2\right), y \in [0,1) \tag{6}$$

The public parameters in the system can be represented as $\{n, p, q, \chi\}$.

The algorithm will then generate the public key (PK_0, PK_1) as follows. The algorithm first randomly selects $PK_0 \in R_p$. Next, the algorithm picks $MK \in R_p$ and $r \in \chi$ and keeps them secret. After that, the algorithm generates the public key as in equation (7):

$$PK_1 = PK_0 \cdot MK + qr \tag{7}$$

where $PK_1 \in R_p$.

Finally, the algorithm will generate the private key SK as follows. The algorithm randomly selects $r' \in \chi$ and generates the private key as in equation (8):

$$SK = MK + qr' \tag{8}$$

$SK \in R_p$ of them.

II. C. 2) Core encryption algorithms

The core encryption algorithm will encrypt the plaintext M into the ciphertext C using the public key (PK_0, PK_1) based on the secure-space text-similarity joining problem in the following way.

The algorithm selects a secret value $s \in R_p$, and an error term $e_1 \in \chi$, employs a public key PK_1 and a public parameter of the system q, and encrypts the plaintext M into a ciphertext component C_1 as in equation (9):

$$C_1 = PK_1 \cdot s + qe_1 + M \tag{9}$$

After that, the algorithm selects another error term $e_2 \in \chi$ and uses the public key PK_0 and the system public parameter q to generate the ciphertext component C_2 as in equation (10):

$$C_2 = PK_0 \cdot s + qe_2 \tag{10}$$

Finally, the core encryption algorithm will output the encrypted ciphertext C as in equation (11):

$$C = \{C_1 = PK_1 \cdot s + qe_1 + M, C_2 = PK_0 \cdot s + qe_2\} \tag{11}$$

II. C. 3) Core decryption algorithm

The core decryption algorithm is based on the secure space text similarity connection problem, using the key SK, decrypting the ciphertext C, and finally obtaining the plaintext M, the process is as follows.

First, it is necessary to obtain the ciphertext as in equation (12):

$$C=\left\{C_1=PK_1\cdot s+qe_1+M,C_2=PK_0\cdot s+qe_2\right\} \tag{12}$$

Next, the decryption operation is performed using key $SK=MK+qr'$ as in equation (13):

$$\begin{aligned}M'&=C_1-SK\cdot C_2=M\\&+q\left(r\cdot s+e_1-e_2\cdot MK-r'\cdot PK_0\cdot s-qr'\cdot e_2\right)\end{aligned} \tag{13}$$

Finally, the plaintext can be obtained as in equation (14):

$$M=M'\bmod q \tag{14}$$

II. D.Security model

Here, the indistinguishability (IND-CPA) security model of the APIB-BPRE scheme is considered separately for the original ciphertext and the re-encrypted ciphertext under choice of plaintext. The indistinguishability (IND) game of the APIB-BPRE scheme is modeled by the interaction between the adversary A and the challenger C. The challenger C simulates the environment to execute and reply to queries from the adversary A. The exact execution is shown below.

CPA-APIB-BPRE-Or game. In the CPA-APIB-BPRE-Or game, consider the original ciphertext of the APIB-BPRE scheme under the indistinguishability under choice of plaintext (IND-CPA) security model. The adversary A and the challenger C interact with each other to execute the following game. The specific execution process is shown below.

Initialization: the adversary A selects the identity id^* as the challenge identity. Meanwhile, the challenger C performs the setup algorithm to generate the master public key mpk and master private key msk and sends mpk to the adversary A .

Query Phase 1: The adversary A initiates a key generation query. It enters the identity id and the master public key mpk, and if $id=id^*$, C outputs $\perp$. Otherwise, C runs the key extraction algorithm to generate the private key sk_{id} and returns sk_{id} to A .

Challenge: When two messages m_0, m_1 from the plaintext space M are received, C randomly selects $b\in\{0,1\}$ and sets the challenge ciphertext ct_0^* and returns ct_0^* back to the adversary A .

Query Phase 2: The adversary A continues the key extraction query and C gives the corresponding response in the same way as in Query Phase 1.

Guess: The adversary A outputs a guess b' about b . If $b'=b$, the adversary A wins the CPA-APIB-BPRE-Or game. Let $Adv_{CPA-APIB-BPRE-Or}$ define the advantage of the adversary A to win the CPA-APIB-BPRE-Or game, where there is equation (15):

$$Adv_{CPA-APIB-BPRE-Or}=\left|\Pr[b'=b]-1/2\right| \tag{15}$$

Definition 6: A LD-BPRE scheme is said to be CPA-APIB-BPRE-Or if there exists no probabilistic polynomial-time adversary A that wins the CPA-APIB-BPRE-Or game by a margin of $Adv_{CPA-APIB-BPRE-Or}$ at time t, and the LD-BPRE scheme is said to be CPA-secure under the IND-CPA security model regarding the original ciphertext.

CPA-APIB-BPRE-Re game. In the CPA-APIB-BPRE-Re game, consider that the re-encrypted ciphertext of the APIB-BPRE scheme is indistinguishable under the chosen plaintext (IND-CPA) security model. The adversary A and the challenger C interact with each other to execute the following game. The specific execution process is shown below.

Initialization: the adversary A outputs a set of challenge broadcast receivers $S_\mu^*=\left\{id_{\mu_1}^*,\ldots,id_{\mu_k}^*\right\}$, for any μ, where $1\le\mu\le m$, $k\le n$. Meanwhile, C runs the setup algorithm to generate the master public key mpk and the master private key msk, and returns mpk to the adversary A .

Query Phase 1: In this phase, A initiates the query and C replies to the query from A . The specific execution process is shown below.

(1) Key extraction query: the adversary A initiates a private key query for user id. It inputs the identity id and the master public key mpk, and if $id \in S_\mu^*$, C outputs $\perp$. Otherwise, C performs the key extraction algorithm to generate the private key sk_{id} and returns sk_{id} to the adversary A.

(2) Path establishment query: the adversary A initiates a path query about id. It inputs the master public key mpk and the identity id, and C executes the path building algorithm to generate the path $Pa = \left(id = S_0, S_1, \ldots, S_m\right)$ on id, and returns Pa back to the adversary A.

(3) Re-encryption key generation query. The adversary A initiates a re-encryption key query about id. It enters the master public key mpk, the identity id, the set of broadcast receivers $S_{\mu-1}$ and S_μ, where $\left(S_{\mu-1}, S_\mu\right) \in Pa$. C generates rk by running the re-encryption key generation algorithm and returns $rk_{\mu-1\to\mu}$ to the adversary A.

Challenge: Upon receiving two messages m_0, m_1 from the plaintext space M, C randomly selects $b \in \{0,1\}$, computes the challenge ciphertext ct_μ^*, and returns the ct_μ^* back to the adversary A.

Query Phase 2: The adversary A continues to initiate key extraction, path establishment and re-encryption key queries, and C gives the corresponding responses in the same manner as in Query Phase 1.

Guess: The adversary A outputs a guess b' about b. If $b' = b$, the adversary A wins the CPA-APIB-BPRE-Re game. Let $Adv_{CPA-APIB-BPRE-\mathrm{Re}}(\lambda)$ define the advantage of the adversary A in winning the CPA-APIB-BPRE-Re game, where Eq. (16) is available:

$$Adv_{CPA\text{-}APIB\text{-}BPRE\text{-}\mathrm{Re}} = \left|\Pr[b' = b] - 1/2\right| \tag{16}$$

Definition 7: A LD-DPRE scheme is said to be CPA-secure under the IND-CPA security model if there exists no probabilistic polynomial-time adversary A that can initiate up to q_{sk} key extraction queries, q_{cp} path creation queries, and q_{rk} re-encryption key queries at time t with the advantage $Adv_{CPA-APIB-BPRE-\mathrm{Re}}$ to win the CPA-APIB-BPRE-Re game, then the LD-DPRE scheme with respect to the re-encrypted ciphertext is said to be CPA-secure under the IND-CPA security model.

Definition 8: An APIB-BPRE scheme is said to be CPA-secure if the APIB-BPRE scheme is CPA-secure regarding the original ciphertext and the re-encrypted ciphertext under the IND-CPA security model, also called semantically secure.

III. Examination and Evaluation of Data Encryption Algorithms and Security Models

III. A. Performance analysis of data encryption algorithms

III. A. 1) Encryption time comparison

The encryption time of different byte lengths is also different, in order to verify the connection between the two, this subsection selects four byte lengths of encrypted information streams, the information stream bytes are 64 bit, 128 bit, 256 bit, and 512 bit, and the encryption time is tested at the time when the plaintext data is 200kb, 300kb, and 400kb using the grid core encryption and decryption algorithm designed in this paper. The encryption time of the encrypted information stream with different byte lengths is shown in Figure 2.

The encrypted information stream of 64 bit has an encryption time of 6.40s when the plaintext data is 200 kb, and an encryption time of 15.32s when the plaintext data is 400 kb. The encrypted information stream of 128 bit has an encryption time of 13. 01s when the plaintext data is 400 kb. The encryption time for a 256-bit encrypted message stream with plaintext data of 200 kb, 300 kb, and 400 kb is 3. 11 s, 5. 94 s, and 9. 78 s, respectively. The encryption time for 512 bit encrypted message stream plaintext data is 2. 25s, 4.11s and 5.84s at 200kb, 300kb and 400kb respectively. It can be seen that the higher the number of bytes the shorter the encryption time, and the encrypted information flow for each byte length improves as the plaintext data increases.

III. A. 2) Comparison of encryption efficiency

In order to verify the effectiveness of (M5) algorithm in encryption and decryption efficiency, under the condition that the byte length is 512bit and the plaintext data size is 300kb, the encryption and decryption efficiency of the similar algorithms (M1) AES algorithm, (M2) ECC algorithm, (M3) SM2 algorithm, (M4) ZUC algorithm are compared respectively, and the results of the comparison of encryption and decryption efficiency of the five algorithms are shown in Fig. 3. The encryption and decryption time of (M3)SM2 algorithm is the longest, which is 8.37 s and 8.16 s respectively, followed by (M2)ECC algorithm, whose encryption and decryption time is 7.25 s and 7.3 s. In comparison, the encryption and decryption efficiency of the (M4)ZUC algorithm is better, and the encryption and decryption time is 5.48 s. The encryption and decryption time of the (M5)algorithm of the present paper is 3.75 s and 4.13 s, and the decryption and decryption efficiency of the five algorithms is shown in Figure 3. 4.13s, with the highest encryption and decryption efficiency.

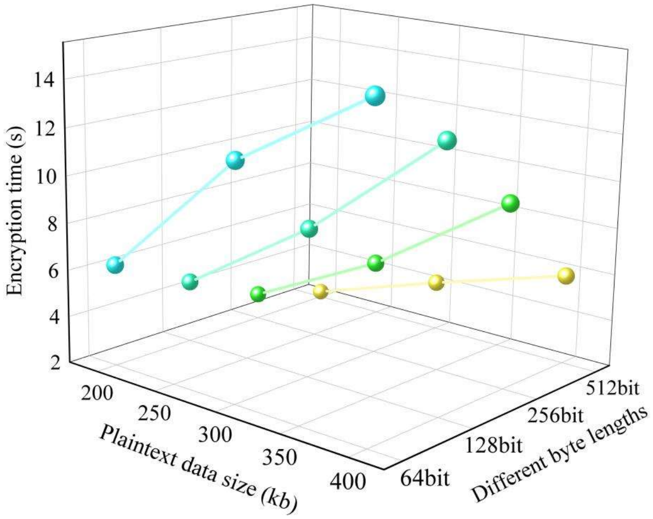

Figure 2: Encryption times of encrypted information flows with different byte lengths

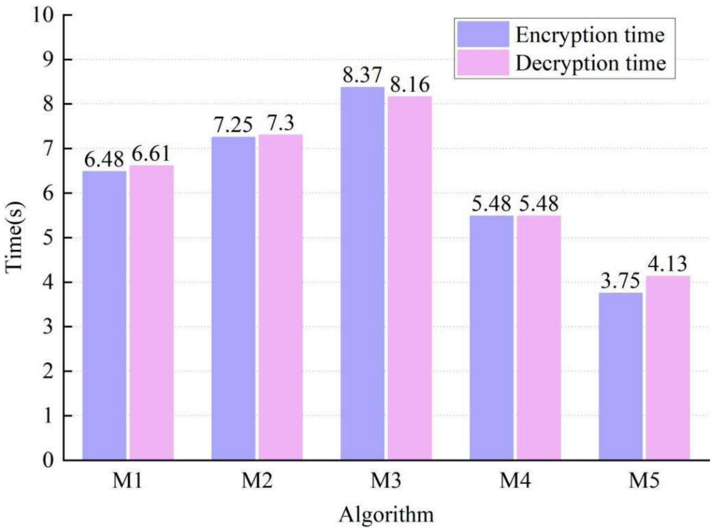

Figure 3: Comparison of encryption efficiency

III. A. 3) Analysis of encryption effect

In order to verify the practical effectiveness of the encryption method designed in this paper, the index of attacked index is introduced to measure the security of data encryption in computer network. The smaller value of the attacked index means the higher security of each data node in the computer network.

In the experimental process, combined with the performance of encryption and decryption efficiency of different algorithms in the previous subsection, the experimental group method: (M5) the algorithm in this paper, and the control group method: (M4) ZUC algorithm, (M1) AES algorithm are used to encrypt the computer network data respectively. After completing the encryption operation of each method, the attacked index of 10 key encryption nodes is calculated according to the above formula is shown in Fig. 4.

From the above figure, it is seen that (M5) this paper algorithm shows optimal encryption performance in the comparison of computer network privacy data encryption. After encryption using (M5) this paper's algorithm, the attack index of each node in the computer network reaches the lowest level and the difference in the attack index between nodes is also minimized. The average attack index of (M5) this paper's algorithm is only 0.057, which is 0.2596 and 0.4514 lower than the control (M4) ZUC algorithm and the control (M1) AES algorithm, respectively.

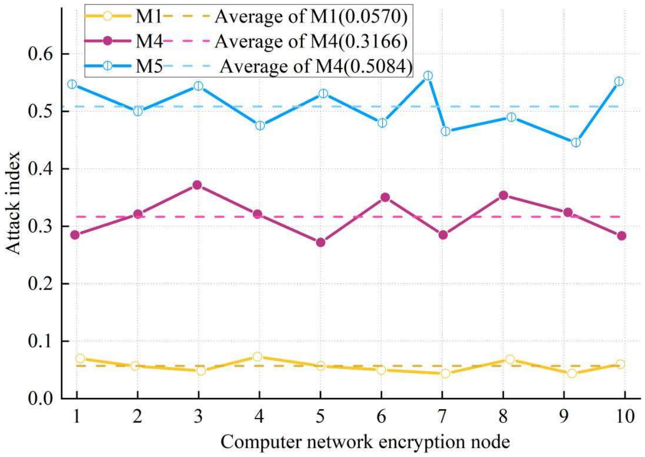

Figure 4: Comparison of privacy data encryption effects of different methods

III. B. Key Simulation Experiments

III. B. 1) Key generation phase

(M5) The comparison results of the algorithm in this paper with (M4) ZUC algorithm and (M1) AES algorithm in the key generation phase of the time required to generate the key are shown in Fig. 5. It can be seen that (M4) ZUC algorithm and (M1) AES algorithm are significantly higher than the algorithm in this paper in the key generation phase, which is mainly due to the fact that (M4) ZUC algorithm and (M1) AES algorithm have performed the modulo-power and modulo-multiplication operations several times, which cost more than the algorithm in this paper. The cost spent is higher. The computational complexity of (M5) algorithm in the key generation phase is relatively low, and users can quickly get the feature conversion key distributed by the key center.

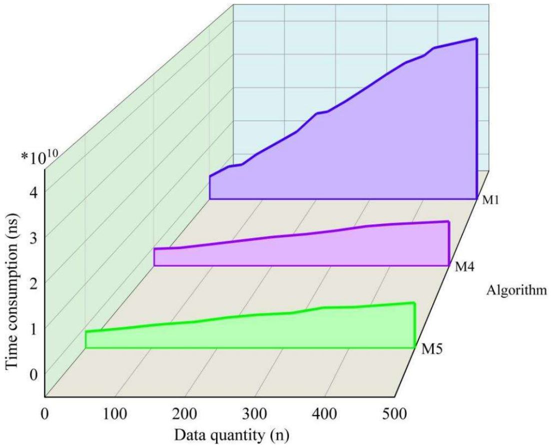

Figure 5: Comparison of key generation stages of different schemes

III. B. 2) Encryption phase

Comparison of the computational time-consuming results in the encryption phase between (M5) this paper's algorithm and (M4) ZUC algorithm and (M1) AES algorithm is shown in Fig. 6, and the results of the three algorithms are more similar. Although the encrypted data phase of the three schemes uses different computational methods, the final computational complexity is similar.

III. B. 3) Decryption phase

The computational overhead of (M5) this algorithm and (M4) ZUC algorithm and (M1) AES algorithm in the decryption stage is shown in Fig. 7.Since (M5) this algorithm and (M4) ZUC algorithm only need to carry out symmetric decryption once in the decryption stage, and the convergence key can be calculated to obtain the plaintext data, the decryption overheads of the two algorithms are similar. While (M1) AES algorithm needs to use power operation and bilinear pair operation in the decryption process, so the computational overhead is larger.

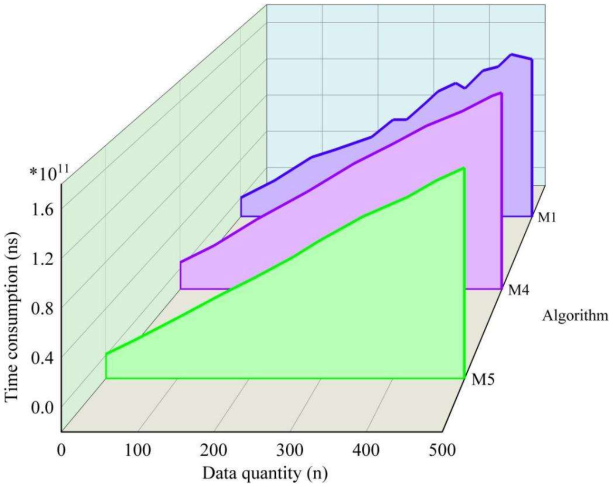

Figure 6: Comparison of encryption stages of different schemes

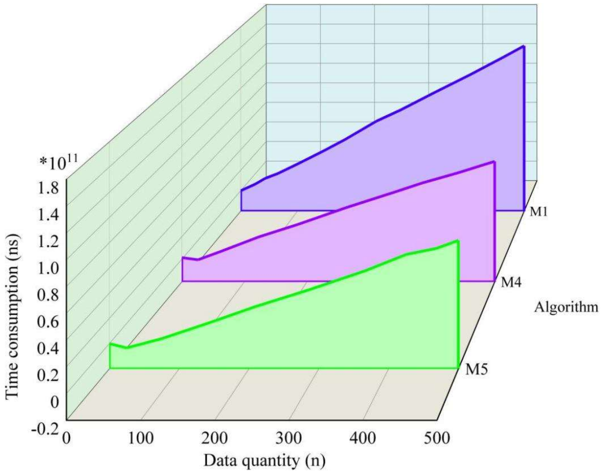

Figure 7: Comparison of decryption stages of different schemes

III. C. Performance testing of the safety model

III. C. 1) Analysis of the degree of privacy protection

The degree of privacy protection is measured using the probability of the target information privacy being recognized by the attacker, using P(D) to denote the value of the probability of the target information privacy being recognized by the attacker. Conventionally, the smaller the value of P(D), the higher the degree of privacy protection, and conversely, the larger the value of P(D), the lower the degree of privacy protection. The probability of target information privacy being recognized by the attacker is obtained through experiments, and the security modeling method designed in this paper is shown in Fig. 8 in comparison with the classical security modeling method.

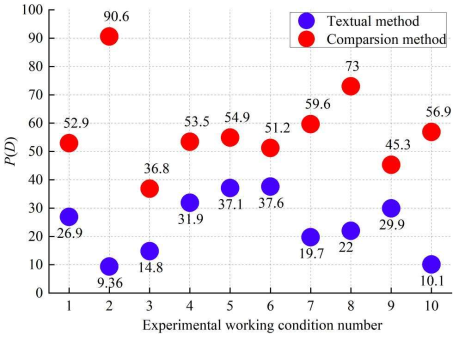

Figure 8: The probability of privacy being identified by attackers

The probabilities of the compared methods are generally higher than 35%, up to 90.6%. In contrast, the probability of target information privacy being recognized by the attacker obtained by the algorithm in this paper has a smaller value, which is distributed between 9.00% and 38.00%, and the minimum value reaches 10.1%, which indicates that the proposed method has a higher degree of protection for data information privacy.

III. C. 2) Information utility analysis

Information utility is measured by averaging the difference between the autocorrelation coefficients of the privacy information, using μ to denote the information utility measure and N to denote the number of privacy information. g_i and g_j denote the autocorrelation coefficients of the privacy information. Conventionally, the smaller the value of g_i, the lower the loss of privacy information, and the higher the availability of the information. On the contrary, the larger the value of g_i, the larger the loss of privacy information and the lower the information usability.

The average value of the difference between the autocorrelation coefficients of the privacy information of the two methods obtained through experiments is shown in Figure 9.

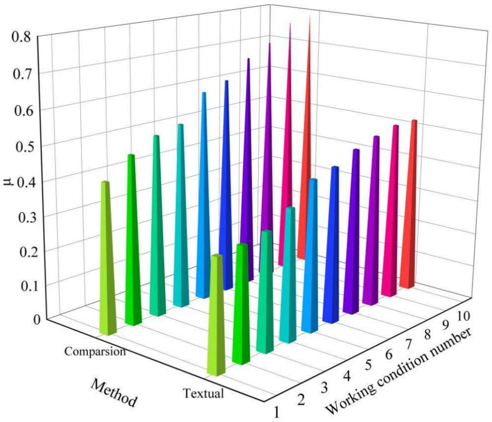

Figure 9: The average value of the differences in the autocorrelation coefficients

With the increase of the amount of private information, the average value of the difference of the autocorrelation coefficient of private information shows a gradual upward trend, but the average value of the difference of the autocorrelation coefficient of private information of the method based on the security model of this paper is generally lower than 0.55, while the comparison method reaches as high as 0.79, which indicates that the usability of information is higher after the processing of this paper's security modeling method.

The above experimental data show that the average value of the difference between the probability of the target information privacy being recognized by the attacker and the autocorrelation coefficient of the privacy information obtained by applying this paper's method is smaller compared to the comparison method, which indicates that the proposed method has a better degree of privacy protection and information utility, and the performance of privacy information security protection is better.

IV. Conclusion

Aiming at the demand for data encryption and decryption performance and security in the cloud computing environment, this paper consists of a grid core encryption algorithm through the key generation algorithm, the core encryption algorithm, and the core decryption algorithm. And the security model is constructed to ensure the security of data encryption and decryption under different switching conditions.

Compared with similar algorithms, the encryption and decryption time of the designed lattice core encryption algorithm is 3.75s and 4.13s respectively, which has the highest encryption and decryption efficiency. The designed security model obtains a smaller probability of the target information privacy being recognized by the attacker, which can reach 10.1%, and the average difference of the autocorrelation coefficients of the privacy information is lower than 0.55, which indicates that in addition to the lattice kernel encryption/decryption algorithm being able to realize highly efficient and accurate decryption, the security model is also able to withstand the attack of quantum computation, and maintains the safety of the privacy information data.

References

[1] Antonova, A., & Bartkova, S. (2020). An overview of the advantages of cloud computing and online IDE. Automation of technological and business processes, 12(3), 50-54.

[2] Abdel-Basset, M., Mohamed, M., & Chang, V. (2018). NMCDA: A framework for evaluating cloud computing services. Future generation computer systems, 86, 12-29.

[3] Changchit, C., & Chuchuen, C. (2018). Cloud computing: An examination of factors impacting users' adoption. Journal of Computer Information Systems, 58(1), 1-9.

[4] Parast, F. K., Sindhav, C., Nikam, S., Yekta, H. I., Kent, K. B., & Hakak, S. (2022). Cloud computing security: A survey of service-based models. Computers & Security, 114, 102580.

[5] Goyal, M. P., & Deora, D. S. (2022). Reliability of Trust Management Systems in Cloud Computing. Indian Journal of Cryptography and Network Security, 2(1), 1-5.

[6] Venkatesh, A., & Eastaff, M. S. (2018). A study of data storage security issues in cloud computing. International Journal of Scientific Research in Computer Science, Engineering and Information Technology, 3(1), 1741-1745.

[7] Mansouri, Y., Toosi, A. N., & Buyya, R. (2017). Data storage management in cloud environments: Taxonomy, survey, and future directions. ACM Computing Surveys (CSUR), 50(6), 1-51.

[8] Mohamadi Bahram Abadi, R., Rahmani, A. M., & Alizadeh, S. H. (2018). Server consolidation techniques in virtualized data centers of cloud environments: a systematic literature review. Software: Practice and Experience, 48(9), 1688-1726.

[9] Moorthy, V., Venkataraman, R., & Rao, T. R. (2020). Security and privacy attacks during data communication in software defined mobile clouds. Computer Communications, 153, 515-526.

[10] Kaja, D. V. S., Fatima, Y., & Mailewa, A. B. (2022). Data integrity attacks in cloud computing: A review of identifying and protecting techniques. Journal homepage: www. ijrpr. com ISSN, 2582, 7421.

[11] Bhushan, K., & Gupta, B. B. (2017). Security challenges in cloud computing: state-of-art. International Journal of Big Data Intelligence, 4(2), 81-107.

[12] Huang, Q., Yang, Y., Yue, W., & He, Y. (2019). Secure data group sharing and conditional dissemination with multi-owner in cloud computing. IEEE Transactions on Cloud Computing, 9(4), 1607-1618.

[13] Park, D. B., Li, X., Shahhosseini, A. M., & Tsay, L. S. (2021). Data ownership in cloud: Legal issues. International Journal of Forensic Engineering and Management, 1(2), 125-148.

[14] Leontiou, N., Dechouniotis, D., Denazis, S., & Papavassiliou, S. (2018). A hierarchical control framework of load balancing and resource allocation of cloud computing services. Computers & Electrical Engineering, 67, 235-251.

[15] Devi, P. (2018). Attacks on Cloud Data: A Big Security Issue. International Journal of Scientific Research in Network Security and Communication, 6(2), 15-18.

[16] Talha, M., Sohail, M., & Hajji, H. (2020). Analysis of research on amazon AWS cloud computing seller data security. International Journal of Research in Engineering Innovation, 4(3), 131-136.

[17] Ong, A. K. S., Altes, G. C., & German, J. D. (2024). Actual usage assessment among cloud storage consumers in the Philippines using a machine learning ensemble approach. Scientific Reports, 14(1), 28955.

[18] Ahmadian, M., & Marinescu, D. C. (2018). Information leakage in cloud data warehouses. IEEE Transactions on Sustainable Computing, 5(2), 192-203.

[19] Seth, B., Dalal, S., Jaglan, V., Le, D. N., Mohan, S., & Srivastava, G. (2022). Integrating encryption techniques for secure data storage in the cloud. Transactions on Emerging Telecommunications Technologies, 33(4), e4108.

[20] Salman, D., & Sulaiman, N. (2024). A Review of Encryption Algorithms for Enhancing Data Security in Cloud Computing. AlKadhim Journal for Computer Science, 2(1), 53-71.

[21] Oladoyinbo, T. O., Oladoyinbo, O. B., & Akinkunmi, A. I. (2024). The Importance Of Data Encryption Algorithm In Data Security. Current Journal of International Organization of Scientific Research Journal of Mobile Computing & Application (IOSR-JMCA), 11(2), 10-16.

[22] Yazdeen, A. A., Zeebaree, S. R., Sadeeq, M. M., Kak, S. F., Ahmed, O. M., & Zebari, R. R. (2021). FPGA implementations for data encryption and decryption via concurrent and parallel computation: A review. Qubahan Academic Journal, 1(2), 8-16.

[23] Gudimetla, S. R. (2024). Data encryption in cloud storage. International Research Journal of Modernization in Engineering Technology and Science, 6, 2582-5208.