Research and Analysis on Comparison Scheme of IoT Encrypted Data in Cloud Computing Environment

Rong Xu[1(✉)] and Fu-yong Bian[2]

[1] School of Information Engineering,
Anhui Radio and TV University, Hefei 230022, China
xurong528528@163.com
[2] Chuxiong Medical College, Chuxiong 675005, China

Abstract. Conventional cloud computing iot encrypted data comparative analysis method to simple encryption scheme comparison, the accuracy of the scheme comparison is not high, for more complex data encryption scheme comparison, comparative stability lower deficiencies, therefore put forward under the cloud computing environment, the Internet of things encrypted data comparison analysis research. Introducing the sliding window technology, build the Internet of things to encrypt data security evaluation mechanism, determine the iot encrypted data comparison analysis algorithm, constructing iot encrypted data comparison analysis model is presented. Run the analysis model, analyze the iot encrypted data comparison scheme, and implement the iot encrypted data comparison scheme analysis. The experimental data show that the proposed scheme is more routine than the conventional scheme analysis, and the stability of the method is maintained at 70%–90%, which is suitable for the comparison scheme analysis of the network encryption data with different difficulty coefficients. The proposed method for comparative analysis of IoT encrypted data is highly effective.

Keywords: Cloud computing environment · The Internet of Things · Encrypt data · Comparative analysis

1 Introduction

The conventional cloud computing IoT encrypted data comparison analysis method can compare simply encrypted data schemes. When comparing with more complex encrypted data schemes, due to analysis technology limitations, there is a relatively low stability [1]. Thus, the research and analysis on comparison scheme of IoT encrypted data in cloud computing environment is proposed. The sliding window comparison technology is introduced, the security evaluation mechanism for IoT encrypted data is built, the analysis algorithm of IoT encrypted data comparison scheme is determined, and the analysis model of IoT encrypted data comparison scheme is constructed; using IoT encrypted data comparison scheme analysis model algorithm for parameter generation Gen, data partition Par, label generation Der, ciphertext generation Enc, and ciphertext comparison cmp calculations to determine the relationship between num,

S. Liu and G. Yang (Eds.): ADHIP 2018, LNICST 279, pp. 195–203, 2019.
https://doi.org/10.1007/978-3-030-19086-6_22

num*, to achieve the IoT encrypted data comparison scheme analysis. Thus, the research and analysis on comparison scheme of IoT encrypted data in cloud computing environment is completed. To ensure the effectiveness of designed IoT encrypted data comparison and analysis method, the IoT encrypted data test environment is simulated, by using two different methods of comparative analysis of IoT encrypted data, comparative stability simulation test is performed, the experimental results show that the proposed method for comparative analysis of IoT encrypted data is highly effective.

2 System Objective and Analysis

Research and analysis on comparison scheme of IoT encrypted data in cloud computing environment mainly includes:

(1) Solve the problems existing in the comparative analysis methods of conventional cloud computing IoT encrypted data, optimize the model building process, scientifically and reasonably combine modern computer technologies to build a security evaluation mechanism for IoT encrypted data.
(2) Optimize IoT encrypted data comparison scheme analysis algorithm design, analyze IoT encrypted data variable parameters.
(3) Optimize parameter generation Gen, data partition Par, label generation Der, ciphertext generation Enc, and ciphertext comparison cmp calculations to determine the relationship between num, num*, to achieve the IoT encrypted data comparison scheme analysis.

3 Construction of IoT Encrypted Data Comparison Scheme Analysis Model

3.1 Introduction of Sliding Window Comparison Technology

The sliding window comparison method is generally applied to power exponent operations. In general, the integer e is divided into fixed-length blocks, and then multiplication of non-zero block times is performed [2]. If the used block lengths are different, non-zero blocks can be reduced to reduce the total number of multiplications. This method of segmentation is called sliding window comparison. In practical applications, the sliding window comparison method will be optimized because in the binary representation of numbers, both zero and non-zero bits are significant. Therefore, no distinction is made between zero-window comparison and non-zero window comparison. Instead, the binary form of numbers is uniformly windowed so that each window is equal in size. This technique of improving efficiency by reducing the amount of calculation has attracted wide attention from various industries [3].

3.2 Construction of the Security Evaluation Mechanism of IoT Encrypted Data

Firstly, the weak distinguishable is defined. Data can be divided into strong data types and weak data types. Strongly typed languages is a language that always compels type definitions. Java and Python are strongly defined. If there is an integer, you can't treat it as a string if you don't convert it. Weakly typed definition language, a type of language that can be ignored, contrary to strongly typed definitions. VBScript is weakly defined [4]. In VBScript, the string '12' can be concatenated with the integer 3 to get the string '123', then it can be taken as an integer 123 without the need to display the conversion. Weakly typed data is called weak distinguishability. Then the security evaluation program under the weak characteristics of the IoT encrypted data scheme in cloud computing environment is introduced. This part is to prove that the IoT encrypted data security evaluation plan satisfies the weak distinguishability under the standard model [5].

It's assuming that challenger C and opponent A ask for weakly distinguishable competition. Firstly, Challenger C receives the security parameter k ∈ N and the range parameter n ∈ N. The parameter generation algorithm Gen is then executed, i.e., Gen(k, n) = (param,mkey), and the generated public parameter param is returned to the opponent A. Opponent A inquiries Challenger C. During this process, Challenger C responds to the inquiry as follows:

First of all, Challenger C receives any inquiry number 0 < num < 2, then executes the label generation algorithm Der and returns the generated label token = Der(param, mkey,num) [6].

Then, Challenger C receives any inquiry number 0 < num < 2n, and then executes the encryption algorithm Enc and returns the ciphertext ciph = Enc(param,mkey,num).

Finally, Challenger C receives a set of numbers 0 < num * 0 < num * 1 < 2n that need to be interrogated. The challenger randomly selects b ∈ {0, 1} and a ciphertext ciph* = Enc(param,mkey,numb) is generated [7]. In this process, Opponent A does not allow the following questions:

$$num = \sum_{i=0}^{n-1} \alpha_i 2^i; num_0^* = \sum_{i=0}^{n-1} \beta_i 2^i; num_0^* = \sum_{i=0}^{n-1} \gamma_i 2^i, \alpha_i, \beta_i, \gamma_i \in \{0,1\} \quad (1)$$

In the formula: n is the number of times of inquiry and α_i is the probability that the enemy A can inquire the probability; and β_i is the probability that the enemy B can inquire, γ_i is the probability that the enemy C can inquire about.

On the end of process, the results that Opponent A send $b' \in \{0,1\}$ to C is $Exp_{C,A}^k = \begin{cases} 1, & b = b' \\ 0, & b \neq b' \end{cases}$, defined as in any polynomial time, after Opponent A asks, $Adv_{C,A}^k := \left| \Pr(Exp_{C,A}^k = 0) - \Pr(Exp_{C,A}^k = 1) \right|$ is ignorable for k in weak distinguishability, thus, the establishment of security evaluation mechanism for IoT encrypted data is performed.

3.3 Design of IoT Encrypted Data Comparison Scheme Analysis Algorithm

There are five steps for IoT encrypted data comparison scheme analysis algorithm, which are respectively parameter generation Gen, data partition Par, label generation Der, ciphertext generation Enc, and ciphertext comparison cmp.

Generation Gen needs to give the security parameter k ∈ N and the range parameter n ∈ N. The algorithm outputs the public parameter param and the master key mkey, i.e., Gen(k,n) = (param,mkey). The data partition Par: num = (b0, b1, …, bn − 1); bi ∈ {0, 1} is the binary representation of the given number, the output value is num = $(B_0, B_1, \ldots, B_{m-1})$; $\frac{n}{m} = t$. Label generation Der needs to give public parameter param, master key mkey, and number num. Algorithm output label token, namely, token = Der(param,mkey,num). Ciphertext generation Enc needs to give public parameter param, master key mkey, and number num [8]. Algorithm output ciphertext ciph, namely, ciph = Enc(param,mkey,num). Ciphertext comparison cmp needs to give public parameter param, ciphertext ciph, ciph* and the corresponding label token of one ciphertext. Algorithm output result is $Cmp = \begin{cases} -1, & num > num^* \\ 0 & num = num^* \\ 1 & num < num^* \end{cases}$.

Figure 1 shows the schematic diagram of analysis model of the IoT encrypted data comparison solution. The system model includes three entities, namely data owners, cloud tenants, and cloud servers. The data owner needs to encrypt the shared data before uploading it to the cloud server [9]; the semi-trusted cloud server is responsible for data storage and retrieval operations; the cloud tenant is responsible for submitting query requests to get the size relationship of the data. Based on the sliding window comparison technology, the security evaluation mechanism for IoT encrypted data and design of IoT encrypted data comparison program analysis algorithm is used to achieve the construction of IoT encrypted data comparison program analysis model.

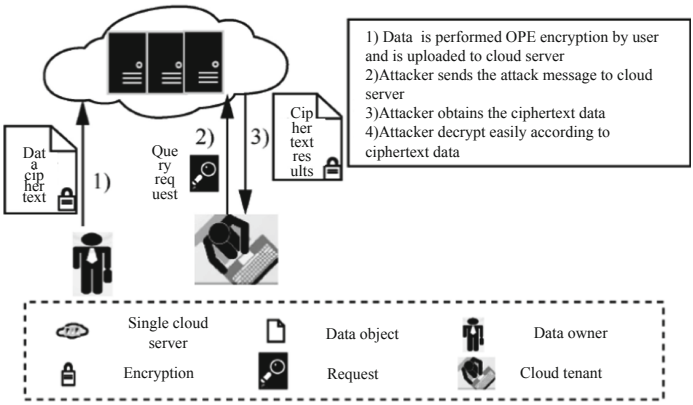

Fig. 1. IoT encrypted data comparison program analysis model

4 Analysis of IoT Encrypted Data Comparison Program

4.1 Operation Process of IoT Encrypted Data Comparison Program Analysis Model

It's assuming that the binary length of the number is nbit, each window contains tbit information, where n is a multiple of t. In fact, n can be of any length. If n cannot be divisible by t, it can be zero-padded until it is a multiple of t. For calculation convenience, the following representative meaning of formula is given: H_1 represents the function Hasla, H_2 represents the function $Hash_2$, H_3 represents the function $Hash_3$, H represents the hash function, mkey represents the master key, I represents the number Ls degree, n represents the number of hash function operations, m represents the number of windowing, and param represents the output parameter [10].

The parameter generation Gen, data partition Par, label generation Der, ciphertext generation Enc, and ciphertext comparison cmp calculation are performed by IoT encrypted data comparison scheme analysis model.

Generation Gen needs to give the security parameter $k \in N$ and the range parameter $n \in N$. H1, H2, H3 are randomly selected to satisfy the condition $\{0, 1\}^k \times \{0, 1\}^* \rightarrow \{0, 1\}^k$ algorithm outputs the public parameter param and the master key mkey. Among them, param = (n, H_1, H_2, H_3).

Data partition Par needs to give the binary representation of the number num = $(b_0, b_1, \ldots, b_{n-1})$; $b_i \in \{0, 1\}$, the output of the algorithm is the packet data with windowing value t. After windowing, data is expressed as $num = (B_0, \ldots, B_{m-1}) = \sum_{i=0}^{m-1} B_i (2^t)^i; \frac{n}{m} = t$, where $B_0 = (b_0, \ldots, b_{t-1}), \ldots, B_{m-1} = (b_{n-t}, \ldots, b_{n-1})$.

Label generation Der needs to give the public parameter param, master key mkey, and number num after windowing. The above formula generates the label d_i. The label output by the Der algorithm is token = $(d_1, d_2 \ldots, d_m)$. Among them, there are $d_i = H_1(\text{mkry}, B_m, B_{m-1}, \ldots, B_i)$, $i = 1, 2, \ldots, m$.

Ciphertext generation En needs to give the public parameter param, master key mkey, and number num after windowing. The Enc algorithm randomly generates $I \in \{0, 1\}^k$ and label token = $(d_1, d_2 \ldots, d_m)$, generates fi, and outputs ciphertext ciph = (I, $(f_0, f_1 \ldots, f_m)$). In order to make the length of the ciphertext shorter, $(f_0, f_1 \ldots, f_{m-1})$ is converted into an integer $F = \sum_{i=0}^{m-1} f_i (2^{t+1} - 1)^i$ to be stored, where $f_1 = H_1(d_{i+1}, I) + H_2(\text{mkey}, d_{i+1}) + B_i \bmod \left(2^{(t+1)} - 1\right)$.

Ciphertext comparison cmp calculation needs to give the public parameter param, ciphertext ciph = $(I, (f_0, f_1 \ldots, f_{m-1}))$, $\text{ciph}'^* = (I', (f'_0, f'_1 \ldots, f'_{m-1}))$ and the corresponding label token of one ciphertext. The first different window $c_j = f_j - f'_j - H_3(d_{j+1}, I) \bmod \left(2^{(t+1)} - 1\right)$ is obtained by comparison, where $j = m - 1, \ldots, l$. The output results of Cmp algorithm is

$$Cmp = \begin{cases} -1(num > num^*) & 1 \le c_j \le 2^t - 1 \\ 0(num = num^*) & c_j = 0 \\ 1(num < num^*) & 2^t \le c_j \le 2^{(t+1)} - 2 \end{cases} \quad (2)$$

The parameter generation algorithm is mainly used to generate the public parameter param and the master key mkey used in the following steps. The label generation algorithm is mainly used to generate label token related to the number num, and token* generated by the number num*. Similar to this process, the ciphertext generation algorithm is mainly used to generate the ciphertext ciph related to the number num, and the ciphertext ciph* generated by the number num*. The ciphertext comparison algorithm mainly uses the previously generated ciphertext data and the label associated with one of the numbers to perform a comparison operation, and ultimately determines the difference relationship between first different windows of a pair of ciphertexts ciph and ciph*.

4.2 Analysis of IoT Encrypted Data Comparison Program

Running the analysis model of the IoT encrypted data comparison program, it can be obtained that H_1, H_2, H_3 are pseudo-random functions, and then $|Adc^k_{C,A} - Adc^k_{C_B,A}| < \varepsilon$ and $Adc^k_{C_B,A} = 0$, the IoT encrypted data analysis program satisfies the weak distinguishability. It's assuming that a pair of ciphertexts ciph and ciph* that need to be compared are generated by number $num = \sum_{i=0}^{m-1} b_i 2^i = num = \sum_{i=0}^{m-1} B_i (2^t)^i, \frac{n}{m} = t$ and number $num = \sum_{i=0}^{m-1} \beta_i 2^i = num = \sum_{i=0}^{m-1} B_i^{'} (2^t)^i$ known by Gen. Among them, t represents the window size of windowing, m represents the total number of windows.

It can be known by data partition Par that the label token generated by num and num* are expressed respectively token = $(d_1, d_2 \ldots, d_m)$ and token* = $(d'_0, d'_1 \ldots, d'_m)$. In addition, the ciphertext generated by num and num* are expressed respectively ciph = $(I, (f_0, f_1 \ldots, f_{m-1}))$ and ciph'* = $(I', (f'_0, f'_1 \ldots, f'_{m-1}))$, to make the ciphertext shorter, $(f_0, f_1 \ldots, f_{m-1}), (f'_0, f'_1 \ldots, f'_{m-1})$ are converted respectively as $F = \sum_{i=0}^{m-1} f_i (2^{t+1} - 1)^i$ and $F = \sum_{i=0}^{m-1} f'_i (2^{t+1} - 1)^i$ to be stored.

From these relationships it can be seen that the components d and d′ in the label token are only related to B_i, B_{i+1}, B'_i, B'_{i+1} and mkey. Assuming that l is the first different window of num and num*, then for $i = l+1, \ldots, m-1$, if $B_{i+1} = B'_{i+1}$, then $d_{i+1} = d'_{i+1}$.

If num = num*, $\forall i = 0, 1, \ldots, m-1$, then Cmp output is 0. If num $\neq$ num*, for the first different window, the following formula is given by:

$$C_j = f_j - f'_j - F_3(d_{j+1}, I) \bmod (2^{(t+1)} - 1) \quad (3)$$

Based on the analysis model of the IoT encrypted data comparison scheme analysis model, running the analysis model, and using the IoT encrypted data comparison scheme analysis model algorithm to perform parameter generation Gen, data partition

Par, label generation Der, ciphertext generation Enc, and ciphertext comparison cmp calculation, determine the relationship between num, num*, and to achieve the IoT encrypted data comparison program analysis.

5 Experimental Test and Analysis

To ensure the effectiveness of the analysis and comparison study of IoT encrypted data in cloud computing environment proposed in this paper, simulation experiments are conducted. In the test process, different IoT encrypted data were used as test objects to conduct comparative stability simulation tests. Different types of structures and difficulty coefficient of IoT encrypted data are simulated. In order to ensure the validity of the experiment, the conventional IoT encrypted data comparison and analysis method in cloud computing was used as a comparison object. The results of the two simulation experiments were compared and the test data was presented on the same data chart.

5.1 Preparation of Experimental Test

In order to ensure the accuracy of the simulation test process, the test parameters of the test are set. This article simulates the test process, uses different IoT encrypted data as the test object, uses two different methods of comparative analysis of IoT encrypted data, conducts a comparative stability simulation test, and analyzes the simulation test results. Because the analysis results obtained in different methods and the analysis methods are different, the test environment parameters must be consistent during the test. The test data set results in this paper are shown in Table 1.

Table 1. Test data set

Simulation experiment parameters	Implementation range/parameters	Observation
Difficulty coefficient of IoT encrypted data	DC0.1–DC1.6	DC unit of difficulty coefficient, maximum 2.0
IoT encrypted data	.DCD/.DGC, encryption of logic data, data volume 0–1 GB	Design analysis with two different methods
Simulation system	DJX-2016-3.5	Windows platform

5.2 Analysis of Experimental Test Results

During the testing process, two different methods of comparative analysis of IoT encrypted data were used to work in a simulated environment, and the changes in the comparative stability were analyzed. At the same time, due to the use of two different methods for comparative analysis of IoT encrypted data, the analysis results cannot be compared directly. For this purpose, third-party analysis and recording software is used to record and analyze the test process and results, and the results are displayed in the

comparative curve of the test results. In the simulation test result curve, the third-party analysis and recording software function is used to eliminate the uncertainty caused by simulation laboratory personnel operation and computer simulation equipment, the comparative stability simulation test was conducted only for different IoT encrypted data, different methods for comparative analysis of IoT encrypted data. The test results are compared with the histogram shown in Fig. 2.

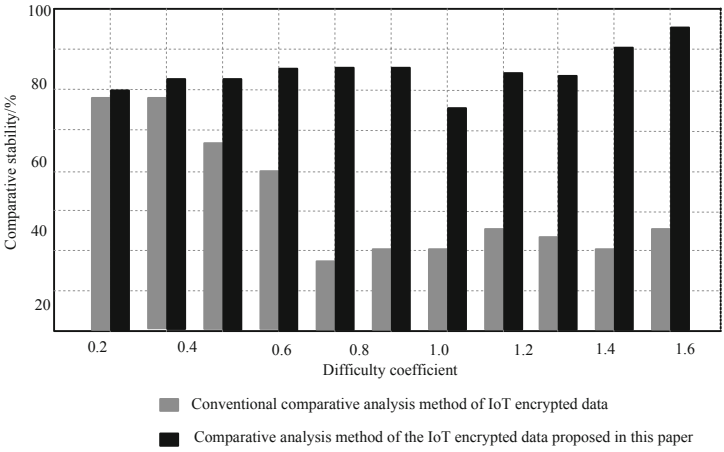

Fig. 2. Comparison of test results with histogram

Based on the results of the test histogram, using third-party analysis and recording software, the comparative stability of comparative analysis method of the IoT encrypted data proposed in this paper and the conventional comparative analysis method of IoT encrypted data in cloud computing are arithmetically weighted. The comparison and analysis method of the Internet of Things encryption data proposed in this paper is arithmetically weighted with the stability of the conventional cloud computing IoT encryption data comparison analysis method, and the comparison histogram is obtained. The stability of the method is maintained at 70%–90%, suitable for The analysis of networked encrypted data comparison schemes with different difficulty coefficients is more effective.

6 Conclusion

This paper presents the analysis and comparison of the IoT encrypted data in the cloud computing environment, based on the analysis model of the IoT encrypted data comparison program, the operation of analysis model and the analysis of IoT encrypted data comparison program is performed to achieve the study of this article. The experimental data shows that the method designed in this paper has extremely high effectiveness. It is hoped that the research in this paper will provide a theoretical basis for the comparative analysis of IoT encrypted data.

References

1. Meng, Q., Ma, J.F., Chen, K.F., et al.: IoT encrypted data comparison solution based on cloud computing platform. J. Commun. **1**(4), 34–37 (2018)
2. Cheng, Z.Q., Lian, H.P.: Research on encryption simulation of IoT communication characteristic data information. Comput. Simul. **33**(11), 324–327 (2016)
3. Cao, J.C.H., Li, C.: Study on IoT database storage system of maritime military mass data. Ship Sci. Technol. **33**(12), 175–177 (2016)
4. Ma, X.X., Yu, G.: Publicly accountable ciphertext-policy attribute-based encryption scheme. Comput. Sci. **44**(5), 160–165 (2017)
5. L, Y.: Realization of multi-card recognition in UHF radio frequency identification system in internet of things. Microelectron. Comput. **36**(11), 104–107 (2017)
6. Li, J.R., Li, X.Y., Gao, Y.L., et al.: Research on data forwarding model in internet of things. J. Softw. **22**(1), 196–224 (2018)
7. Qin, X.J.: Research on privacy protection encryption algorithms with smaller space in IoT environment. Bull. Sci. Technol. (2018)
8. Li, W., Ge, C.H.Y., Gu, D.W., et al.: Research on statistical fault analysis of LED lightweight password algorithm in internet of things environment. J. Comput. Res. Dev. **54** (10), 2205–2214 (2017)
9. Xu, J.G., Zhang, J.: IoT data parallel transmission path prediction simulation. Comput. Simul. **32**(1), 172–175 (2018)
10. Gu, W.J.: Research on data scheduling of shared resources in internet of things. Comput. Simul. **34**(1), 268–271 (2017)