ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 5, Issue 2, February 2018

# Homomorphic Decryption Technique in Cloud Computing for Privacy Preserving

[1] G.Ashmitha, [2] K.Jayashree, [3] Syed Abdul Moeed
[1][2] M.Tech-CSE, [3] M.Tech(Ph.D)-CSE
[1][2]BITS, Narsampet, [3]Mewar University

***Abstract* - The rapid development of cloud computing technology makes the cloud service model has a vast application space, this model allows users to have the incomparable computing power and storage space and other advantages. In the cloud service mode, the privacy and security of users is the primary problem in their promotion and application. How to ensure the privacy of data while ensuring the availability of data is a major challenge in the process of computing data. Homomorphic encryption As a key measure to solve this problem, it has become a hot issue in recent years both at home and abroad. This paper introduces the research progress of cloud computing privacy security and homomorphic encryption, the classification of homomorphic encryption algorithms, the theory of security, the realization technology of homomorphic decryption scheme and the application of homomorphic encryption technology in cloud computing privacy protection. The advantages and disadvantages of homomorphic encryption schemes are introduced and analyzed, and the future research directions are put forward.**

***Keywords:* --- Cloud service, Homomorphic Decryption, Cipher-text computation, Privacy security**

## 1. INTRODUCTION

The rapid development of cloud computing technology makes all types of derivative services widely used.People are free from space and their own terminals through cloud serviceAccording to the processing capabilities to complete a series of huge limitationsdataAnalysis. DealStorage and other worksEnterprises will have to rely on large computer centers to finish mission of outsourcing in the AmazonCloud computing centers such as Google and other reduction operators30% above hardware costNot only has it been significantly reducedTerminal above the head simultaneously Provide unmatched computer power to the usersAlmost unlimited storage space and great economic potential2014Global Cloud ComputingBusiness revenue has been reached1480One hundred million US dollars2016The global cloud computing market has reached a quarter-quarter increase30%To2020Global City. The value will be achieved2700 One hundred million US dollar.However, the structural features of a computing environment also pose greater security risksRaise funds to reduce costsConsiderations of useNodes involved in computing are mostly heterogeneous and different in the typeRare physical distributionService providers can hardly check all nodescontrolUsers often rely on less reliable network communications and semi-confidential storage servers for data transfer and storageThere must be violenceDanger of sensitivity to breadIt is difficult to ensure the confidentiality of the dataIntegrityAndavailability.

1. This article makes a current analysis of cloud computing services Privacy and security issues, The Privacy of the user's privacy on key technologies - Development of Homomorphic Encryption Technology, The not enough, The next step must be studied And possible cases, The With the anticipation of cloud services in the future in the privacy of scientific research users and cloud computing industries Discover.

2 In general, the Division introduces the current status of privacy and security of cloud computing , The Privacy cloud threats and the advantages and disadvantages of current privacy protection methods analysis , The At the same time, the research progress on homomorphic encryption technology is summarized

3. This article introduces the theory and concept of homomorphic decryption algorithm

4. Homomorphic decryption Algorithm Classification and Relationship Technical researchAnalysis was carried out.

5.The Potential Application of Homomorphic decryption Technology in the Future in Protection of Cloud Protection of Privacy.

## 2. THE KEY TECHNOLOGY: HOMOMORPHIC DECRYPTION

Homomorphic decryption refers to performing a decryption function in cipher text space. The input is an encrypted decryption key and an encrypted cipher text.

The output is a fresh cipher text and the fresh cipher text has lower noise. Sounds a little bit awkward, in order to understand this process, think of plaintext space, if the deciphering function is executed in plaintext space, the input should be the decryption key and cipher text, and the output is plaintext. Because plaintext space and cipher text space are homomorphic, so mapping to cipher text space, input and output are plaintext space content encryption. One of the questions that follow is whether decryption functions can be performed in cipher text space. The reason for this problem is that the decryption function also has the problem of calculating the depth. Once the depth of the decryption function exceeds the tolerance range of the SWHE, the decryption function cannot be executed in the cipher text space. If the decryption function cannot be executed, the noise cannot be reduced and the noise cannot be reduced any depth calculation, that is, cannot achieve complete homomorphism. Unfortunately, the decryption function, constructed according to Gentry's logic, proved unsuitable for SWHE calculations. The solution to Gentry is to squeeze the decryption function into an acceptable set of functions for the SWHE. The basic idea of the compression and decryption function is "preprocessing", that is, adding a part of the private key information to the public key, preprocessing the cipher text at the time of encryption, and adding some auxiliary information to the cipher text in advance to reduce the decryption function Burden, reduce the depth of the decryption function. Incidentally, this preprocessing encryption method is widely used in the server-assisted encryption scheme, the computing power of weak client will be partially decrypted key information leaked to the server, with the server's powerful computing power to partially decrypt, leaving only A small amount of work to run the client. As the homomorphic decryption overhead is very large, so later someone invented the key exchange technology, die exchange technology to reduce cipher text noise.

## 3. THE PROGRAM DESCRIPTION

***(1) KeyGen algorithm (key generation)***

The algorithm generates an encrypted public key and decrypts the private key, and may also generate a cipher text-calculated public key for use in Evaluate cipher text calculations.

***(2) Enc algorithm (encryption)***

The algorithm generates a cipher text. In addition, the cipher text generated by Enc has the lowest noise and the cipher text noise after the homomorphism is gradually enhanced.

***(3) Dec algorithm (decryption)***

When the cipher text noise is within the threshold, the decryption is correct, and beyond the threshold, the decryption is not reliable.

***(4) Evaluate algorithm (cipher text calculation)***

This algorithm is the most important part of completely homomorphic encryption. With Evaluate, any function can be calculated and the input is cipher text. It is especially important that the algorithm can calculate the decryption function, which is the key to forming a complete homomorphic encryption scheme.

## 4. DEVELOPMENT

(1) In 1978, Rivest et al. proposed the concept of homomorphic encryption and introduced RSA, the earliest public-key cryptosystem, which satisfies multiplicative homomorphism.
(2) In 2009, Gentry proposed the first complete homomorphic encryption scheme [1]. The scheme is based on the ideal lattice structure. The difficulty is based on two assumptions: the standard and lattice-intensive subsets and problems.
(3) In 2009, Dijk and Gentry et al. proposed a complete homomorphic encryption scheme on integers [2]. The difficulty is based on the approximate GCD problem. The main contribution of this scheme is to replace the original SWHE based on "ideal lattice" with a very simple integer description SWHE.The concept was greatly simplified, but still using Gentry's homomorphic decryption technology to convert SWHE to FHE. So efficiency has not improved.
(4) The completely homomorphic encryption scheme proposed by Brakerski, Gentry and Vaikuntanathan in 2011 [3], referred to as BGV scheme, can be regarded as the second-generation FHE. The difficulty is based on LWE (Error Correction Learning). The use of key exchange technology, mode switching technology to reduce the cipher text dimension and reduce noise, making the efficiency greatly improved.

## 5. CASE ANALYSIS

Choose a relatively easy to understand program analysis here, in 2009 Dijk and Gentry and other integer based on the modulo arithmetic of the complete homomorphic encryption.

The first is a symmetric partial homomorphic encryption scheme:

***Key Generation:***
Generate Key p (Odd), Public Parameter q (Large Integer)

***Encryption:***
$c = m + 2r + pq$

***Decryption:*** $m = c \bmod p \bmod 2 = (c- c / p) \bmod 2$ = LSB (c) XOR LSB ("c / p")

***Cipher text Calculation:***
Basic Addition and Multiplication Calculation. Where r is a random small integer; plaintext $m \in \{0,1\}$; cipher text c; "": rounding; LSB: least significant bit.
Suppose a mod $p \in (-p / 2, p / 2)$, $m + 2r$ as "noise", plaintext space is {0,1} and cipher text space is an integer field.

***Proof of correctness:*** If the guarantee noise $m + 2r < p / 2$, can be correctly decrypted, $(m + 2r + pq) \bmod 2 = (m + 2r) \bmod 2 = m$, the program parameters can be chosen to ensure fresh The noise of the text $m + 2r$ must be less than $p / 2$, so it can be decrypted successfully.

***Homomorphism verification:*** Suppose $c1 = m1 + 2r1 + pq1$, $c2 = m2 + 2r2 + pq2$, among them c1 is the cipher text m1 encrypts, c2 cipher text m2 cipher.
$c1 + c2 = (m1 + m2) +2 (r1 + r2) + p (q1 + q2)$ If noise of $c1 + c2$ is $2 (r1 + r2) + m1 + m2 < p / 2$
Then $((c1 + c2) \bmod p) \bmod 2 = ((m1 + m2) +2 (r1 + r2)) \bmod 2 = m1 + m2$,
$c1 * c2 = (m1 + 2r1) (m2 + 2r2) + p (pq1q2 + m1q2 + m2q1 + 2r1q2 + 2r2q1)$
Then $(c1 * c2 \bmod p) \bmod 2 = (m1 + 2r1) (m2 + 2r2) \bmod 2 = m1 * m2$, ie multiplicative homomorphism;
It can also be seen from the above equation that the sum of the ciphertexts is the sum of the cipher text noise; and the cipher text product noise is the product of the noise.
The solution described above is symmetric encryption, into asymmetric encryption (public key encryption) is also very convenient.

***Key Generation:*** private key sk = p, public key pk = {x1, x2, ...,xt}, where $xi = pqi + 2ri$, that is, xi is the encryption of 0.

***Encryption:*** $c = m + 2r + \text{sum} (S)$, where S is a random subset of {x1, x2, ...,xt} and sum (S) is the sum of some 0 encodings.

***Decryption:*** $m = c \bmod p \bmod 2 = c - c \bmod 2$ = LSB (c) XOR LSB ("c / p")

***Cipher text Calculation:*** Basic Addition and Multiplication Calculation.
To add that the approximate GCD problem here is to give a partial $xi = pqi + 2ri$, it is hard to find p.
Improve the above program, making it a completely homomorphic encryption scheme.

***Key Generation:*** The private key sk '= <s1, s2,sm> is a random binary vector with the sparse subset S = {i: si = 1}, the public key pk' = <pk, y1, y2, ...,ym>, where the rational yi∈ [0,2) and sum (yi) ≈ (1 / p) mod 2, i∈S

***Encryption:*** c '= <c, z1, z2, ...zm>, where $zi = (c * yi) \bmod 2$

***Decryption***:m = (c '- sum (si * zi)) mod 2 = LSB (c') XOR LSB (sum (si * zi)

***Cipher text calculation:*** Calculated by the basic addition and multiplication. As can be seen from the above improvements, the key and cipher text volume are increased, at the expense of the decryption function is reduced, that is, the complex division operation c / p with sum (si * zi), of course To actually simplify the decryption function, Hamming Weight's technique is also used to simplify the binary addition calculation.

## 6. CONCLUSION:

Completely homomorphic can solve the problem of cloud computing security, entrusted to the cloud data are encrypted, the cloud can be decrypted without the user request operation. For example, a bank has many transactional data to analyze, but if its own data processing capability is weak, it can encrypt the transaction data and submit it to the cloud data processing center for analysis. The processing center analyzes the data, draws the result and returns. In this process, the data processing center is exposed to the cipher text, so that we can fully ensure the confidentiality of bank data. Another example, medical institutions can encrypt the patient's medical data stored to the cloud, the cloud some statistical analysis of the data, you can predict the condition and give recommendations for treatment, thus ensuring the patient's privacy and take full advantage of the powerful cloud The computing power. Homomorphic encryption can also be used for spam filtering. If you publish an encrypted public key, your friends can send you encrypted emails, but spammers can also use public key encryption ads and other spam to fill your mailbox. With full homomorphic encryption, spam filters filter out spam if it cannot be decrypted.

## 7. REFERENCES:

[1]. Fully Homomorphic Encryption Using IdealLattices

[2]. Fully Homomorphic Encryption over theIntegers

[3]. Fully Homomorphic Encryption without Boot strapping

[4]. Can Homomorphic Encryption be Practical

[5]Goldreich O, Kushilevitz E, Sudan M. Private information retrieval.Journal of the Acm,1998,45(6):965-981.

[6]Dan B,Kushilevitz E, Ostrovsky R, et al.Public Key Encryption That Allows PIR Queries.Advances in Cryptology CRYPTO 2007.2007:50-67.

[7]AvniH,DolevS,GilboaN,etal.SSSDB:Database with Private Information Search.Algorithmic Aspects of Cloud Computing.Springer International Publishing, 2016.

[8]Song D X, Wagner D, PerrigA.Practical Techniques for Searches on Encrypted Data.IEEE Symposium on Security and Privacy, 2000:44-55.

[9]Benaloh J, Chase M, Horvitz E, et al.Patient controlled encryption:ensuring privacy of electronic medical records.ACM Cloud Computing Security Workshop, Ccsw 2009, Chicago,Usa,November DBLP, 2009:103-114.

[1]G.Ashmitha, completed M.Tech from Kakatiya Institute of Technology and Science, having experience of one year as Asst.Prof in the Dept of CSE from BITS, Narsampet. Interested subjects are DBMS, CC, SE, and OOAD.

[2]K.Jayashree, completed M.Tech in CSE having working experience of 10+ years in teaching. Currently working ate BITS, Narsampet and Interested Subjects are CN, CC, C

[3]Syed Abdul Moeed, M.Tech (Ph.d) in CSE dept having a work experience of 11+ Years. Pursuing Ph.d from Mewar University in CSE Dept. His Interested Research Subjects are SE, CC, Image Processing, CG.