FULLY HOMOMORPHIC SYMMETRIC KEY ENCRYPTION WITH SMITH NORMAL FORM FOR PRIVACY PRESERVING CLOUD PROCESSING

C.N.Umadevi
Research and Development Centre
Bharathiar University
Coimbatore
Tamilnadu
India
cnumadevi.s@gmail.com

N.P.Gopalan
Professor
Department of Computer Applications
National Institute of Technology
Tiruchirapalli
Tamilnadu
India
npgopalan@nitt.edu

ABSTRACT

Demand for security mechanisms increases with the increase of Cloud computing applications. Data holders can keep their private data securely on a cloud and clients can access them on demand. A number of encryption mechanisms ensure the security of private data but none of them supports to operate on the cipher. Homomorphic encryption is a solution as the existing schemes are complex and impractical. The present paper addresses the issue of ensuring security of private data, stored in the Cloud by Symmetric Fully Homomorphic encryption scheme. The data is converted into a square matrix and a private key pair is used to generate Q^n_P matrix called Golden matrix. The Homomorphic property of the scheme is ensured by the plaintext square matrix and the Q^n_P matrices, since they are in Smith Normal form. The clients may access the cipher for computations with no knowledge of the original data in the Cloud. The encryption scheme assuages the owners of the cloud data to secure their private data from perilous clients.

Key words: Fully Homomorphic encryption; Smith Normal form; Linear Congruence; Uni-modular matrices; Golden matrices.
.

ICIA-16, August 25-26, 2016, Pondicherry, India
© 2016 ACM. ISBN 978-1-4503-4756-3/16/08…$15.00
DOI: http://dx.doi.org/10.1145/2980258.2980462

1. INTRODUCTION

Cloud computing is a tool which facilitates today's data-centric world, through its scalable, economic and required computing infrastructure, with a ethical of service levels. Cloud computing also enables the consumers to store their valuable information and to analyze it by using shared computing resources regardless of its volume and complexity. Cloud computing faces many security threats and the conventional encryption techniques can overcome them. Ensuring security and confidentiality of stored data are the goals of encryption. Though cloud supports the availability of data, it must provide confidentiality and integrity of the stored data. The computationally weaker client depends on the Cloud Service Provider (CSP) and utilizes the resources of cloud. Thus the Computation-As-A-Service (CAAS) is the most promising services of the cloud [1]. The cloud owner outsources the private data and to maintain confidentiality, the data is encrypted. So for the user of the cloud is allowed to perform computations over the encrypted data by decrypting it and thus the secrecy disperses.

Homomorphic encryption (HE) is an encryption technique which supports us to perform operations on the encrypted data and when a HE scheme allows all the possible computations, it is known as Fully Homomorphic Encryption (FHE). Let *M be* a data and *C(M)* is its cipher. By homomorphic encryption a function *F* on *C (M)* is equal to M as shown in equation (1):

$$F(C(M)) = C(F(M)) \quad \text{-------} \quad (1)$$

The FHE scheme allows the cloud users to perform computations on the data without loss of security. To have privacy the FHE scheme must be a Symmetric Homomorphic Encryption [1, 2].

1.1 Literature Survey

Privacy homomorphism exists from 1970's. Later Partial Homomorphic encryption systems were proposed [11] and addition or multiplication operations (but not both) are only supported by them. A recent development in 2009 by Craig

Gentry [5] was a FHE scheme. So for several improvements and variations of Craig's method have been developed [4,6, 7,8, 9, 11, 12, 15] which are based on Ideal Lattices and needs to be bootstrapped and are quite complex. An approach in [16] is a FHE scheme, based on integers rather than Ideal Lattices. FHE based on approximate GCD [16], CRT based FHE [18] and large integer factorization with arbitrary matrix as key [19] is some of the ideas found in the literature with integers in background. In this paper, we propose a new Symmetric FHE scheme, based on Smith Normal form with Q_P^n matrix as secret key.

1.2 Our Contribution

Most of the FHE cryptosystems are public-key based and we introduce a new Symmetric FHE scheme. The technique found in the work of Gupta and Sharma [1, 2] was a Symmetric key FHE scheme and it has a drawback of using two keys like public key system , a secret key for encryption and another is a refresh key. An alternate to this scheme is using an integer key pair *(n1, n2) mod N* which is used to generate Q_P^n matrices which are being used for encryption .Thus the proposed technique do not use the Symmetric key directly for encrypting the data. In addition, computations are bounded by the Ring Z_N through which the homomorphic property was made light weight. Where N is obtained by the multiplication of 2*m* numbers and N is non prime.

Moreover the data to be encrypted is transformed into a diagonal matrix, such that the primitive operations on matrices are simple. Thus the computational complexity is significantly reduced. The security of this scheme is ensured by large integer factorization technique, a key concept in RSA algorithm[21].

This paper is organized as follows: preliminary concepts are introduced in Section 2. Section 3 introduces the key generation, encryption and decryption process. Section 4 is about the application of the proposed scheme and Section 5 is the conclusion.

2. PRELIMINARIES

2.1 Definitions and Notations

The proposed scheme translates all the integers in a Ring Z_N to operations on matrix M_4 (Z_N), where *N* is non prime and it is product of 2m numbers. The scheme is made CPA secure by a security parameter denoted by λ and to withstand η number of plain text attacks, m and λ are selected such that $\eta = m \ ln \ poly(\lambda)$ where $poly(\lambda)$ is a fixed polynomial in λ. This scheme involves *2m* odd, mutual prime numbers $p1_i$ and $p2_i$, $1 \leq i \geq m$, let $f_i = p1_i \, p2_i$ and $N = \prod_{i=1}^{m} f_i$ where *N* is a RSA type modulus of unknown factorization. Primality testing of a number takes only polynomial time, thus choosing *2m* odd mutual prime numbers, which involves a time complexity of *O(m)* that is *O(poly(λ))*. By making the large integer factorization infeasible, factoring *N* in polynomial time is also infeasible.

2.2 Fibonacci Numbers and Golden Matrices

2.2.1 Fibonacci Q-Matrix

Theory of Fibonacci numbers supplemented a new theory called Fibonacci Q-Matrix a 2×2 matrix of determinant ±1[13]. The following theorem can be used to generate a Q-Matrix:

Theorem 1: For a given integer *n*, Q^n matrix is given by the following equation (2).

$$Q^n = \begin{pmatrix} Fib(n+1) & Fib(n) \\ Fib(n) & Fib(n-1) \end{pmatrix} \text{ -------- (2)}$$

Where *Fib(n-1), Fib(n)* and Fib(n+1) are Fibonacci numbers and the matrix Q^n is represented by the following recursive equations (3) and (4) :

$$Q^{n-2} = Q^n + Q^{n-1} \text{ ---------- } (3)$$

$$Q^{n-3} = Q^{n-1} + Q^{n-2} \text{ --------} (4)$$

These two recursive relations are used to obtain the explicit form of the matrices Q^n and Q^{n-1} as in Table 1.

Table1. Explicit form of Q^n and Q^{n-1}

n	0	1	2	3
Q^n	$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$	$\begin{pmatrix} 2 & 1 \\ 1 & 1 \end{pmatrix}$	$\begin{pmatrix} 3 & 2 \\ 2 & 1 \end{pmatrix}$
Q^{-n}	$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 1 \\ 0 & -1 \end{pmatrix}$	$\begin{pmatrix} 1 & -1 \\ -1 & 2 \end{pmatrix}$	$\begin{pmatrix} -1 & 2 \\ 2 & -3 \end{pmatrix}$

2.2.2 Generalized Fibonacci Q_P Matrices

Alexey Stakhov generated the concept of Q-matrix called Fibonacci Q_P matrix - a (P+1)(P+1) square matrix [13, 14] and its generalized form is given below:

$$Q_P = \begin{pmatrix} 1 & 1 & 0 & 0 & . & . & . & 0 & 0 \\ 0 & 0 & 1 & 0 & . & . & . & 0 & 0 \\ 0 & 0 & 0 & 1 & . & . & . & 0 & 0 \\ . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . \\ 0 & 0 & 0 & 0 & . & . & . & 1 & 0 \\ 0 & 0 & 0 & 0 & . & . & . & 0 & 1 \\ 1 & 0 & 0 & 0 & . & . & . & 0 & 0 \end{pmatrix}$$

Q_P matrix with values of P=1, 2 and 3 are given below:

$$Q_1 = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$$

$$Q_2 = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}$$

$$Q_3=\begin{pmatrix}1 & 1 & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\\ 1 & 0 & 0 & 0\end{pmatrix}$$

The Q_P matrix can be raised to the nth power, resulting Q_P^n matrix, where P is the Fibonacci P-number, n=0, ±1, ±2, ----- and the determinant of Q_P^n matrix is -1. Thus Q_P^n is unimodular. A Q_P^n matrix with n=3 and P=3is given below:

$$Q_3^3=\begin{pmatrix}3 & 2 & 0 & 0\\ 2 & 1 & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{pmatrix}$$

2.3 Smith Normal Form

Smith normal form of matrices has many applications like solving systems of linear Diophantine equations, the determination of the invariant polynomials, linear programming of integers and so on. The definitions and notations on Smith Normal Form are discussed below:

Definition 1:

Let R_{mxn} denotes all the integer matrices of order $m \times n$. The unit *(or* unimodular) elements of $R_{m\times n}$ are those with determinant ± 1.

Definition 2:

The matrices *A, B of* $R_{m\times n}$ are equivalent, if $B = P\ A\ Q$ for some unimodular matrices $P \in R_{m\times m}$ *and* $Q \in R_{n\times n}$. Then *A, B* are in Smith Normal Form (SNF) and equivalence is written as B~A.

2.3.1 Matrices over Z_N

The symmetric key of this scheme is a square, uni-modular and invertible Q_P^n matrix with elements from the ring Z_N and all matrix operations are performed modulo *N*. The following definition describes the properties of matrices used in this paper:

Definition 3:

A matrix $S \in M_N(Z_N)$ is invertible if and only if its determinant $K \neq 0$ and GCD of (K,N) =0031 .

Definition 4:

A matrix $B \in M_N$ is said to be unitarily equivalent to a matrix $A \in M_N$ and *U* and *V* are unitary matrices such that $V \in M_N$ and $U \in M_N$ then, $B=UAV$ and $A=U^{-1}BV^{-1}$ where, U^{-1}, V^{-1} are the inverses of *U* and *V* respectively.

3. ENCRYPTION AND DECRYPTION

The following algorithm describes the process of key generation Keygen(n1,n2), encryption Enc(x) and decryption Dec(C) :

Keygen (n1, n2):

i. Select 2m mutual odd primes $p1_i$ and $p2_{i\ ,}$ 1<=i<=m.
ii. Let $f_i=p1_i p2_i$, $N=\prod_{i=1}^{m} f_i$.
iii. Choose two large integer key pairs *(n1, n2) mod N* and a value P not greater than 9 to fix the order of the square matrices.
iv. Generate Q_P^{n1} and Q_P^{n2} , compute their inverses.

Enc (x):

i. Choose an arbitrary integer r$\in Z_N$.
ii. For 1<=i<=m, set values a_i, b_i and c_i according to the following distribution[1,2]:

$$\Pr\begin{pmatrix}ai=x\\ bi=r\\ ci=r\end{pmatrix} = 1-(1/m+1)$$

$$\Pr\begin{pmatrix}ai=r\\ bi=x\\ ci=r\end{pmatrix} = (1/2(m+1)) \text{ and}$$

$$\Pr\begin{pmatrix}ai=r\\ bi=r\\ ci=x\end{pmatrix} = (1/2(m+1))$$

iii. Construct the following linear congruence using CRT, compute a, b and c [1,2].

$$a \equiv a_i \pmod{f_i}$$
$$b \equiv b_i \pmod{f_i}$$
$$c \equiv c_i \pmod{f_i}$$

Generate a diagonal matrix Diag (x, a, b, c).

iv. Output the cipher $C=(Q_P^{n1}.\ \text{Diag}\ (x,a,b,c)\ .\ Q_P^{n2}) \pmod N$

Dec (C):

Output: $((Q_P^{n1})^{-1}.\ C\ (Q_P^{n2})^{-1}) \pmod N$

Moreover the homomorphic addition, subtraction and multiplication can be computed as:

Add (c1, c2): output c = (c1+c2) mod N

Sub (c1, c2): output c = (c1-c2) mod N

Mul (c1, c2): output c = (c1×c2) mod N

The following example shows that the encryption and decryption techniques are correct:

Let m=2, f1=3×7 and f2=5×11, so that N=1155. Choose P=3, n1=4 and n2=5 thus the symmetric keys are Q_3^4 and Q_3^5 with inverse $(Q_3^4)^{-1}$ and $(Q_3^5)^{-1}$. To encrypt the integer data x=257$\in Z_{1155}$, a random number r=291$\in Z_{1155}$ is chosen. Let a1=m=257, b1=c1=r=291, a2= c2=r=91 and b2=m=257. Then CRT is used to solve the linear congruence a=291 mod 21, a=291 mod 55, b=257 mod 21, b=291 mod 55 and c=291 mod 21, c=257 mod 55, let the solutions be a=291, b=236 and c=312. Construct a

diagonal matrix Diag 257,291,236,312). Final step is to construct the cipher.

$$C = Q_3^4 \times Diag\,(257,291,236,312) \times Q_3^5$$

$$C = \begin{bmatrix} 1031 & 336 & 0 & 0 \\ 777 & 654 & 0 & 0 \\ 0 & 0 & 236 & 0 \\ 0 & 0 & 0 & 312 \end{bmatrix}$$

The cipher can be decrypted to get the original data x.

$$x = (Q_3^4)^{-1} \times C \times (Q_3^5)^{-1} \pmod{1155}$$

$$= \begin{bmatrix} 257 & 0 & 0 & 0 \\ 0 & 291 & 0 & 0 \\ 0 & 0 & 236 & 0 \\ 0 & 0 & 0 & 312 \end{bmatrix} = 257$$

4. SYMMETRIC FHE SCHEME AND THE CLOUD

Delivering hosted services over the internet is the primary goal of Cloud Computing. Cloud has many threats for providing successful services to its clients and one of them is securing the privacy of stored data. The existing cryptographic techniques provide different solutions but do not support computations over the encrypted data. The FHE scheme proposed in this paper could be deployed in such environment. Cloud comprises the following key components: Cloud owner, Cloud Service Provider (CPA), Third Party Auditing System and Remote users. The Cloud owner encrypts the data by generating the needed key and hosts the encrypted data in the Cloud. The Remote users are the users of the Cloud who are continuously monitored by the Third Party Authentication System, by issuing session key for each user and it ensures encryption enforced access control. The man in the middle attack can also be easily handled by our scheme because of the session key. Encrypted data, are stored in the data centers accessed through cloud servers and are administrated by the CPA. The CPA has no knowledge on the encryption algorithm and the key. This scenario is shown in Figure 1.

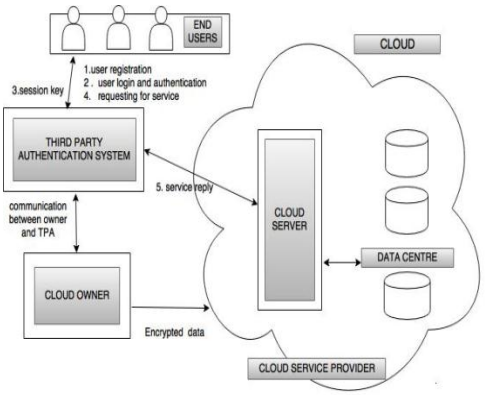

Figure1. FHE scheme and the Cloud

4.1 Comparison of key sizes of FHE schemes with our scheme

Comparing our scheme with Gupta and Sharma [1, 2], the time complexity of encryption and decryption algorithms are same except the key size that is $O(1)$. Rather than using arbitrary matrix as key, our scheme use integer key pairs. Also the time required to generate two Q_P^n matrices from the key pair *(n1, n2) mod N* is *O(log n)* each[20]. The following Table.2 shows the comparison of key size of our scheme with [1, 2, 7, 10].

Table.2 Comparison of our scheme with other FHE schemes

	BGV	**DGHV**	**Gupta and Sharma**	**Our scheme**
Key size	Same as plaintext	$O(\lambda^{10})$	$O(m\lambda)$	$O(1)$

5. CONCLUSION

Elasticity, pay-by-use, self-service and programmability are the key attributes of Cloud Computing. But Cloud is vulnerable to many problems related to privacy and security. In this paper, we have proposed a Symmetric Fully Homomorphic Encryption scheme using Smith Normal form and Q_P^n matrix. This scheme provides a new method to encrypt information without bootstrapping and requires no refreshing procedure. Precisely it is a light weight process in which encryption and decryption requires only two modular multiplications. More over all operations are performed in an algebraic domain of rings of

commutative matrices which we likely new to cryptographic applications. This technique works on integers rather than bits. The future work is to extend this process to protect categorical data including integers.

6. REFERENCES

[1] C.Gupta and I. Sharma, A fully homomorphic encryption scheme with symmetric keys with application to private data processing in clouds, Proceedings of Networks of the Future (NOF), 2013 Fourth International Conference. pp.1-4, 23-25 Oct. 2013.

[2] It Sharma, A Symmetric FHE Scheme Based on Linear Algebra, International Journal of Computer Science & Engineering Technology (IJCSET), vol. 05, pp. 558-562, 2014.

[3] Coron JS, Lepoint T, Tibouchi M., New multilinear maps over the integers, Annual Cryptology Conference 2015 Aug 16 (pp. 267-286), Springer Berlin Heidelberg.

[4] Cheon JH, Stehlé D, Fully homomophic encryption over the integers revisited, Annual International Conference on the Theory and Applications of Cryptographic Techniques 2015 Apr 26 (pp. 513-536), Springer Berlin Heidelberg.

[5] C. Gentry, A Fully Homomorphic Encryption scheme, Dissertation. Sep 2009. Available at https:// crypto. stanford.edu/craig/craig-thesis.

[6] C. Gentry and S. Halevi, Implementing Gentry's fully homomorphic encryption scheme, EURO-CRYPT 2011, LNCS, Springer, K. Paterson (Ed.),2011.

[7] M. van Dijk, C. Gentry, S. Halevi, and V. Vaikuntanathan, Fully homomorphic encryption over the integers, Proceedings of Eurocrypt-10, Lecture Notes in Computer Science, vol 6110,. Springer, pp 24-43, 2010.

[8] J.-S Coron, A.Mandal, D.Naccache, and M. Tibouchi. "Fully homomorphic encryption over the integers with shorter public-keys", Advances in Cryptology - Proc. CRYPTO 2011, vol. 6841 of Lecture Notes in Computer Science. Springer, 2011.

[9] Z..Brakerski and V.Vaikuntanathan, Efficient fully homomorphic encryption from (standard) LWE, Foundations of Computer Science, 2011. Available at Cryptology ePrint Archive, Report 2011/344.

[10] Z.Brakerski, C.Gentry, and V.Vaikuntanathan, Fully homomorphic encryption without bootstrapping, Cryptology ePrint Archive, Report 2011/277.

[11] R. Rivest, L. Adleman, and M. Dertouzos, On data banks and privacy homomorphism's, Foundations of Secure Computation, pp 169-180, 1978.

[12] N. P. Smart and F. Vercauteren, Fully homomorphic SIMD operations, Cryptology ePrint Archive, Report 2011/133.

[13] A.P.Stakhov, A Generalization of the Fibonacci Q-Matrix, Reports of the National Academy of sciences of Ukraine", vol.9, pp.46-49, 1999

[14] A. P. Stakhov,**"**A History, the Main Mathematical Results and Applications for the Mathematics of Harmony, Applied Mathematics, vol.5, pp. 363-386, 2014.

[15] Zhigang Chen, Jian Wang, ZengNian Zhang and Xinxia Song, A Fully Homomorphic Encryption Scheme with Better Key Size, Cryptology ePrint Archive, Report 2014/697.

[16] N.P.Smart1 and F.Vercauteren, Fully Homomorphic Encryption with Relatively Small Key and Ciphertext Sizes, Cryptology ePrint Archive, Report 2009/571.

[17] J.Coron, T.Lepoint and M.Tibouchi, Batch fully homomorphic encryption over the integers, 2012. Available at http://eprint.iacr.org/2013/36.

[18] J.Kim, M.S. Lee, A.Yun and J .H. Cheon., CRT-based fully homomorphic encryption over the integers, 2012. Available at http://eprint.iacr.org/2013/57.

[19] L.Xiao, O Bastani and I-L.Yen., An efficient homorphic encryption protocol for multiuser systems, 2012. Available at http://eprint.iacr.org/2012/193.

[20] Donald E. Knuth, The Art of Computer Programming, Volume 1, Fundamental Algorithms.

[21] R. Rivest, A. Shamir and L. Adleman, *A Method for Obtaining Digital Signatures and Public-Key Cryptosystems*. Communications of the ACM, 21 (2), pp. 120-126, February 1978.