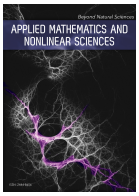
sciendo Applied Mathematics and Nonlinear Sciences 5(1) (2020) 479–484



Applied Mathematics and Nonlinear Sciences

https://www.sciendo.com

Simulation of a Homomorphic Encryption System

Hanife Çağıl Bozduman, Erkan Afacan

Department of Electrical and Electronics Engineering, Gazi University, Ankara, Turkey,
E-mail: bozduman2425@gmail.com, e.afacan@gazi.edu.tr

Submission Info

Communicated by Hacı Mehmet Baskonus.
Received June 15th 2019
Accepted March 2nd 2020
Available online April 10th 2020

Abstract

Cryptology is defined as the science of making communication incomprehensible to third parties who have no right to read and understand the data or messages. Cryptology consists of two parts, namely, cryptography and cryptanalysis. Cryptography analyzes methods of encrypting messages, and cryptanalysis analyzes methods of decrypting encrypted messages. Encryption is the process of translating plaintext data into something that appears to be random and meaningless. Decryption is the process of converting this random text into plaintext. Cloud computing is the legal transfer of computing services over the Internet. Cloud services let individuals and businesses to use software and hardware resources at remote locations. Widespread use of cloud computing raises the question of whether it is possible to delegate the processing of data without giving access to it. However, homomorphic encryption allows performing computations on encrypted data without decryption. In homomorphic encryption, only the encrypted version of the data is given to the untrusted computer to process. The computer will perform the computation on this encrypted data, without knowing anything on its real value. Finally, it will send back the result, and whoever has the proper deciphering key can decrypt the cryptogram correctly. The decrypted result will be equal to the intended computed value. In this paper, homomorphic encryption and their types are reviewed. Also, a simulation of somewhat homomorphic encryption is examined.

Keywords: homomorphic encryption, cryptology, cloud computing
AMS 2010 codes: 94A60

1 Introduction

Cryptology is defined as the science of making communication incomprehensible to third parties who have no right to read and understand the data or messages. Cryptology is divided into cryptography, which is the science of securing data, and cryptanalysis, which is the science of analyzing and breaking secure communication. The main terms in cryptology are given in Table 1.

1.1 Cryptography

Cryptography is the practice and study of hiding information. A cryptographic algorithm works in combination with a key—a word, number, or phrase—to encrypt the plaintext. The security of encrypted data is entirely

ISSN 2444-8656 doi:10.2478/AMNS.2020.1.00046

Table 1 Main terms in cryptology

Keywords	Explanation
Plaintext	Data that are wanted to be protected. Let us call it P
Ciphertext	Encrypted message
Encryption	Method of hiding message. If E refers to encryption function Ek (P) = C
Decryption	Recovering encrypted message. If D refers to decryption function Dk (C) = P
Key	A numeric value to cipher data to protect it

dependent on two things: the strength of the cryptographic algorithm and the secrecy of the key. There are two types of cryptography: symmetric and asymmetric. If the same key is used for encryption and decryption, then the mechanism is called symmetric key cryptography or classical cryptography. This also implies to share a different key with everyone that is wanted to communicate with. Nevertheless, symmetric schemes present the advantage of being fast and should be used as often as possible.

However, if two different keys are used for encryption and decryption, then the mechanism is called asymmetric key cryptography or modern cryptography. The encryption key is public, as the decryption key remains private in this type. Asymmetric schemes are more functional than symmetric ones since there is no need for the sender and the receiver to agree on anything before the transaction. Asymmetric schemes, however, have a big drawback. They are often based on nontrivial mathematical computations and much slower than the symmetric ones. The two most prominent examples are RSA and ElGamal.

The right scheme is the one that fits your constraints in the best way. By constraints, we may understand constraints in terms of time, memory, security, and so forth.

1.2 Cryptanalysis

It is the reverse process of cryptography. The objective of cryptanalyst is to decrypt the ciphertext.

2 Homomorphic Encryption

The goal of encryption is to ensure confidentiality of data in communication and storage processes.

Homomorphic encryption is a form of encryption that allows specific types of computations like addition or multiplication to be carried out on ciphertext. The encrypted result will be the same when decryption is done. Widespread use of cloud computing raises the question of whether it is possible to delegate the processing of data without giving access to it. Encrypting one's data with a conventional encryption scheme to protect one's privacy seems to undermine the benefits of cloud computing since it is impossible to process the data without the decryption key [1]. However, in homomorphic encryption; only the encrypted version of the data is given to the untrusted computer to process. The computer will perform the computation on this encrypted data, without knowing anything on its real value. Finally, it will send back the result, and whoever has the proper deciphering key can decrypt the cryptogram correctly. For coherence, the decrypted result will be equal to the intended computed value.

Homomorphic encryption schemes are methods that allow the transformation of ciphertexts C(M) of message M, to ciphertexts C(f(M)) of a computation/function of message M, without disclosing the message. Generally, an encryption scheme contains a three-step algorithm. They are

1. Key Generation—creates two keys, i.e. the secret key sk and the public key pk.
2. Encryption—encrypts the plaintext m with the public key pk to yield ciphertext c.
3. Decryption—decrypts the ciphertext c with the secret key sk to retrieve the plaintext m [2].

Homomorphic encryption schemes can be classified into three main categories namely: partially homomorphic encryption (PHE), somewhat homomorphic encryption (SWHE), and fully homomorphic encryption (FHE) [6].

2.1 PHE Scheme

The most popular PHE methods available are the RSA, ElGamal, and Paillier methods.

1. *RSA method*: RSA was the first homomorphic encryption scheme developed by Ronald Rivest, Leonard Adleman, and Michael Dertouzos in 1978. It is a public-key crypto technique and is multiplicative homomorphic.
2. *Paillier method*: This is additive homomorphic and is developed by Pascal Paillier in 1999 [2].

2.2 SWHE Scheme

The most popular SWHE method is Boneh-Goh-Nissim (BGN) method. This method allows any number of additions but only one multiplication to be performed on data.

2.3 FHE Scheme

The most popular FHE schemes are algebra homomorphic encryption scheme based on updated ElGamal proposed by Chen Liang and Gao Changmin in 2008 and enhanced homomorphic encryption scheme (EHES) proposed by Gorti VNKV and Subba Rao in 2013. In 2009, Gentry proposed the first not yet broken FHE scheme [3]. FHE refers to cryptosystems that can process both additions and multiplications in the encrypted domain. Any polynomial function over encrypted data can be computed.

3 Simulation of a Homomorphic System

In homomorphic encryption, the encryption of the product of two numbers is equal to the product of the encryptions of the numbers:

$$E(a.b) = E(a).E(b)$$

E is an encryption algorithm or it is also known as a *scheme.* Schemes can be thought as *Somewhat Homomorphic* and *Fully Homomorphic. Fully Homomorphic:* Any kind of operations on ciphertexts can be done. This idea came from Rivest in 1978 but Craig Gentry was the first one whose scheme works on arbitrary functions. *Somewhat Homomorphic:* In this type, two ciphertexts can be added or a ciphertext and a plaintext can be multiplied [6].

In this paper, SWHE is examined. The length of the plaintext and encryption time comparison has been made. To do that, Gentry's encryption scheme is used [7]. The parameters will be

$$\rho = \lambda, \rho' = 2\lambda, \eta = \tilde{O}(\lambda^2), \lambda = \tilde{O}(\lambda^5)$$

λ is a safety parameter. Security parameters associated with the security of the scheme, usually take dozens to hundreds of bits.

Key generation: We choose η bit odd prime numbers p and θ bits odd prime numbers q randomly and order N=pq. Then choose two random integers $l \in [0,2\gamma/p]$ $h \in [-2p, 2p]$, and calculate x=*pl+2h*.

Set public key *pk=(N,x)*=, private key *sk=p*.

Encryption (pk,m): Given a plaintext $m \in \{0,1\}^*$; we choose two random integers $r1 \in$ (-2p,2p) and $r2 \in$ (-2p,2p). According to the public key $pk=(N,x)$, we can calculate

$$c = m + 2r_1 + r_2 x \bmod N \,(c \text{istheresultofciphertext.})$$

Decryption (sk,c): According to the given ciphertext, make use of private key *sk* to calculate:

$$m' = (c \bmod p) \bmod 2$$

Evaluate (pk, C, c1,c2,ct): Given a Boolean circuit C with t inputs and t ciphertexts ci. Let us put T ciphertext into extended circuits to perform all its operations, then verify the result of the circuit's output in (1) to see if it conform (2) [7].

$$c* = \text{evaluate}(pk, C, c) \tag{1}$$

$$Dec(sk, c*) = C(m_1, m_2, \ldots, m_t) \tag{2}$$

In the experimental environment, safe parameter λ is set to the length of the 10 bits and the order of magnitude as 10^5. So the order of magnitude for $p = \lambda$ as 10^5. $p' = 2\lambda$ as 10^5, the order of magnitude for $\eta = O(\lambda^2)$ as 10^{10}, the order of magnitude for $\theta = O(\lambda^4)$ as 10^{20}, and the order of magnitude for $\gamma = \sim O(\lambda^5)$ as 10^{25}. Algorithm keygen of the data is determined by the above parameters. The relationship between the plaintext and ciphertext size is shown in Figure 1. The simulation time is calculated in seconds. It is observed that when plaintext size is increased, the ciphertext size is also increased. But the growthe rate is not high. Still, it is gradually increased.

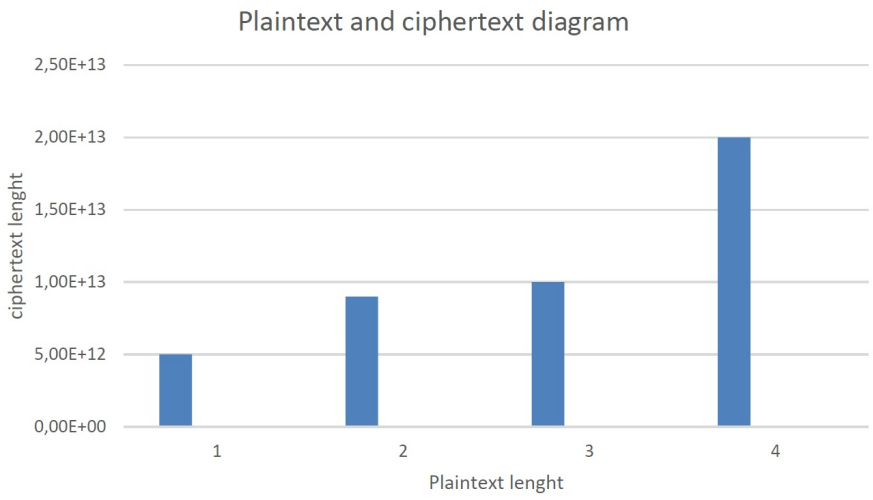


Fig. 1 Relationship between the plaintext and ciphertext size.

4 Conclusion

Cryptography is a powerful tool to protect information. In recent years, cryptography and cryptanalysis had been improved. Widespread use of cloud computing raises the question of whether it is possible to delegate the processing of data without giving access to it. Homomorphic encryption is a new way to protect private data. Because it allows making computation without decrypting data. It is a new field and research is going on.

References

[1] Ayushi Lecturer, “A Symmetric Key Cryptographic Algorithm”, Hindu College of Engineering H.No:438, sec-12, Sonipat, Haryana, ©2010 International Journal of Computer Applications (0975 - 8887) Volume 1 – No. 15.
[2] R. Cramer, R. Gennaro, and B. Schoenmakers, “A secure and optimally efficient multi-authority election scheme”, in Advances in Cryptology (EUROCRYPT ’97), vol. 1233 of Lecture Notes in Computer Science, pp. 103–118, Springer, New York, NY, USA, 1997.
[3] Y. Gahi, M. Guennoun, and K. El-Khatib. “A secure database system using homomorphic encryption schemes”. CoRR, abs/1512.03498, 2015.
[4] Maya Mohan, M K Kavitha, Jeevan Prakash, “Homomorphic Encryption-State of the Art”, International Conference on Intelligent Computing and Control (I2C2), 2017
[5] Keke Gai, Meikang Qiu “An Optimal Fully Homomorphic Encryption Scheme”, IEEE 3rd International Conference on Big Data Security on Cloud.
[6] Craig Gentry, “a fully homomorphic encryption scheme”, STANFORD UNIVERSITY, 2009.
[7] Jing Yang, Mingyu Fan, Guangwei Wang, Zhiyin Kong “Simulation Study Based on Somewhat Homomorphic Encryption”, China, 2014.