International Journal of Engineering & Technology, 7 (4.6) (2018) 100-102

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Homomorphic encryption security - a review

J. Phani Prasad[1]*, B. Seetha Ramulu[2] , E.Amarnatha Reddy[3]

[1]Dept of CSE,Vardhaman College of Engineering,Hyderabad
[2]Dept of CSE,Vardhaman College of Engineering,Hyderabad
[3]Department of Computer Science and Engineering MLR Institute of Technology, Hyderabad
**Corresponding author E-mail:phanimtechcse@gmail.com*

Abstract:

As the Data and Technology is increasing exponentially the security and privacy issues are becoming a major concern in the present scenario, lot of security mechanisms are in to real time and practice, but even there are some bottlenecks with the existing security techniques. In this work we are giving the essence and importance of an encryption scheme called homomorphic encryption and its related issues**.**

Keywords: *Cloud; Decryption; Encryption; Homomorphic encryption; RSA Security*

1. Introduction

Security of the Data or information is a primary concern of the day to day transactional process. There are various domains in which security can be provided as a part of protecting the things and systems, some of the security areas or domains are: home automation and security, providing the security to data pertaining to different organizations. The nominal and quite common security providing methods is usage of passwords in the form of text to our data and having some sort of security to it.
Another form of password protection other than text is providing captcha, or one time passwords now a days. in search engines like Google we can keep multiple authentication schemes like each time one enters in to his Gmail account one can put two factor authentication.
Apart from the above methods we can encrypt our data while storing it/some times whenever we send our information to someone , lot of encryption algorithms and methods are available like DES, AES, RSA, Blow fish and many more, and at the other side the receiver will use a method called decryption to receive the actual data from sender. Cryptography is the method of converting plain text to cipher text. by this method one can just read our data and use it for his purpose without altering that data.

In section 2 the literature review section 3 describes about the Homomorphic encryption and also encryption and decryption schemes, partial Homomorphic encryption Section 4 tells us about Implementation of PHE scheme on cloud by taking a small example also the comparison of various encryption schemes, Section 5 gives us the conclusion.

2. Literature review

Rivest et.al.(1978) introduced for the first time the concept of Homomorphic encryption. Taher (1985) introduced an algorithm based on multiplicative property.

Shahzadi et al (2012) did the study on the three homomorphic encryption algorithms. Naser and Bin (2013) surveyed on specific security issues and use of cryptography in cloud computing.
Carlos *et al*. (2013) discussed about the recent advances in homomorphic encryption techniques. They have done survey on recent advances in Somewhat Homomorphic Encryption (SWHE) and Fully Homomorphic Encryption (FHE) algorithms.
Liu (2012) has introduced some cloud computing system and also analyzes cloud computing security problem. He suggested that single security technique cannot be used to solve the cloud security problem therefore, many traditional and some new strategies are required to use together to provide the total security in cloud.
Ustimenko and Wroblewska (2013) proposed an idea for homomorphic encryption and multivariate key for cloud security. They have given detailed discussion on Key Dependent Message (KDM) encryption scheme can be used for cloud security.

Ramgovind *et al.* (2010) highlighted key security considerations currently faced by industry.

Aderemi and Oluwaseyi (2011) discussed about the security issues in cloud computing and the potentials of homomorphic encryption, and proposed an encryption layer on top of the encrypted data on the cloud.

3. Encryption, decryption and homomorphic encryption

Encryption is a process of converting the actual data in the form of an unknown text called as “cipher”, this cipher cannot be understandable to unauthorized people who wish to use it.
In encryption we have so many types as symmetric encryption and asymmetric encryption. In symmetric encryption the same key is used for both encrypting and for decrypting of the data .
In asymmetric encryption mechanism the sender of the system has one key to encrypt the data first, and the receiver after reception of the data from sender he has to use another key to decrypt the information.

Encryption process is involved with a factor called Key factor, in which the sender of the system includes this key whenever is is

using or applying encryption in his algorithm, along with the plain text to obtain ciphertext.

Decryption is a process of changing the encoded text in to the original format which is understandable to user or the computer system.

Like encryption Decryption also has a concept of Decryption key in which the receiver of the system who is at the other end of the communication applies this key to decode the received cipher text and to obtain the original message.

Homomorphic encryption is a technique that permits the calculation on encrypted data without prior decryption and after operation ,if the data is decrypted by user which is in encrypted form it gives actual result without knowing the actual plain text(yang et al. 2014).

3.1.1. Functions of homomorphic encryption

Homomorphic Encryption H is a set of four functions [11] as shown in Figure 1.

H = {Key Generation, Encryption, Decryption, Evaluation}

1. Key generation: client will generate pair of keys public key pk and secret key sk for encryption of plaintext.

2. Encryption: Using secret key sk client encrypt the plain text PT and generate Esk(PT) and along with public key pk this cipher text CT will be sent to the server.

3. Evaluation: Server has a function f for doing evaluation of cipher text CT and performed this as per the required function using pk.

4. Decryption: Generated Eval(f(PT)) will be decrypted by client using its sk and it gets the original result.

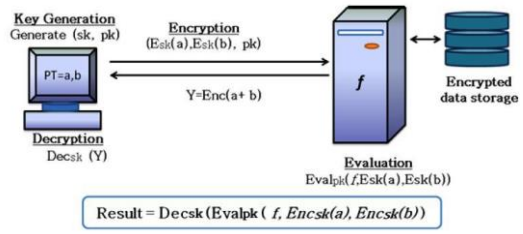

Fig 1: Functions of homomorphic encryption.

Suppose if plain text is M Operation (M) decrypt (Operation (encrypt (M))) (1)

In fig2 (below) Homomorphic encryption is applied on some set of integer values using some algorithm. For encryption algorithm is implemented as 7*2=14 where 2 is the encrypted element in the above operation, and for 5 ie 5*2=10. The reverse process is followed for decryption of data here after multiplication (14*10)/2 =70 it is divided by 2 because of homomorphic encryption property, after decryption of result we will get 7*5=35 which is actual result.

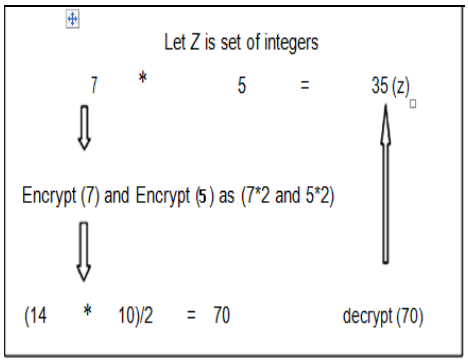

Fig 2: Homomorphic encryption on integers

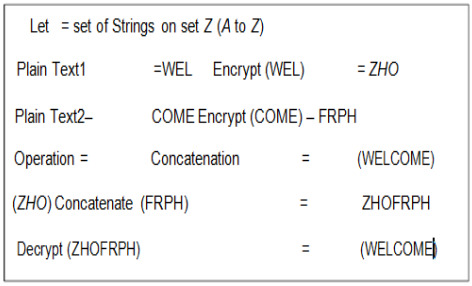

Fig 3: Homomorphic encryption on strings.

3.2.2 Types of homomorphic encryption

There are three types of Homomorphic encryption available:

1. **Partial homomorphic encryption(Phe)**

In this encryption technique it performs single operation on encrypted data i.e either multiplication or addition but not both.

2. **Some what homomorphic encryption(Swhe)**

In this technique it support limited number of addition and multiplication operations on encrypted data.

3. **Fully homomorphic encryption(Fhe)**

In this technique it support both multiplication and addition operations and also any other computations also possible on encrypted data.

4. Rsa and homomorphic encryption:

Rsa is a asymmetric algorithm used for encryption of the data , which was introduced by Ron Rivest, Shamir and ad leman , it is mainly used for encryption using public and private key concepts till now, but it can be combined with homomorphic encryption.

The properties of Homomorphic encryption are:
Encrypt (p1 ⊕ p2) = Encrypt (p1) ⊕ Encrypt (p2) (additive homomorphic property) (2)
Encrypt (p1⊗p2) = Encrypt (p1) ⊗ Encrypt (p2) (multiplicative homomorphic property (3)

4.1 Multiplication Homomorphism using Rsa:

Start
1. Select two large prime numbers r,s
2. Compute n=r*s
3. Pii(n)=(r-1)*(s-1)
4. Select e, where 1<e<pii(n) and e,n are co primes
5. Compute d as (d*e)mod(pii(n))=1
6. Public key{e,n},private key{d,n}
7. Encryption of plaintext $C=M^e \bmod n$
8. Decryption of cipher text $M=c^d \bmod n$

9. RSA follows homomorphic property as:
Encrypt (p1* p2) = Encrypt (p1) ⊗ Encrypt (p2) (multiplicative Homomorphic property).
End.

4.2 Partial homomorphic encryption

Partial Homomorphic Encryption (PHE) can be implemented in various domains like network security, cloud computing, Big data and many other fields where security of data and storage is the major concern.

For example if we take cloud to secure the data, so many encryption and decryption algorithms are in to practice, but among them the Partial Homomorphism is the technique which reduces the amount of computation when compared with other algorithm.

4.3. Implementation of rsa as phe in the cloud:

We are implementing a small case study by taking the length and breadth of 50 grounds and the number of persons visiting the ground for playing and other activities. Ground shape is assumed to be different i.e. (rectangle or square). The data is encrypted and it is stored in a cloud by the user, and whenever it is required user can compute the area of the ground.

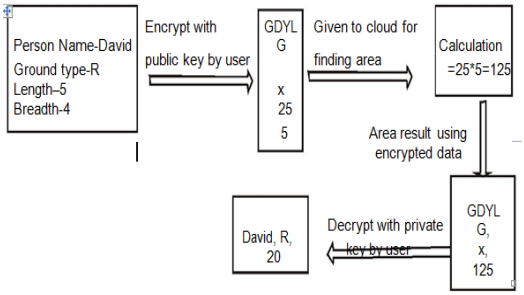

Fig 4: Computation of area in cloud.

Here in the above computation the length value 5 is encrypted as 25 and breadth value 4 is encrypted as 5 and the area is computed as 25*5 which is 125, and David is encrypted as GDYLG now, the user at the other end will compute the decryption on the encrypted data, and the actual result is obtained.

By using Rsa algorithm with the help of public key the data is encoded as {3,55} that is stored in cloud, and the area is computed with the help of encrypted data by the formula (encrypt(length)*encrypt(breadth)) and the computed result is sent to user. The user takes the encrypted form of data and decrypts it by using private key as {27, 33} to get the actual area of the ground.

5. Conclusion

In this work we have discussed about the homomorphic encryption technique as a method of providing security to the data in various fields, and mainly on cloud. We have also implemented RSA as a partial homomorphic technique on cloud by taking a case study. In future it may be extended and the research direction has to be driven towards fully homomorphic encryption technique.

6. References

[1] Taher El Gamal (1985), "A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms", in *Advances in Cryptology*, pp. 10-18, Springer Berlin Heidelberg.

[2] Shahzadi Farah *et al.* (2012), "An Experimental Study on Performance Evaluation of Asymmetric Encryption Algorithms", Recent Advances in Information Science, Proceeding of the 3rd European Conf. of Computer Science, (EECS-12).

[3] Rivest Ronald L, Adleman Leonard M and Dertouzos Michael (1978), "On Data Banks and Privacy Homomorphism", *Foundations of Secure Computation*, Vol. 4, No. 11, pp. 169-180.

[4] Carlos Aguilar Melchor, Simon Fau, Caroline Fontaine *et al.* (2013), "Recent Advances in Homomorphic Encryption", *IEEE Singal Processing Magazine*, March, pp. 108-117.

[5] Liu Wentao (2012), "Research on Cloud Computing Security Problem and Strategy", Proceedings of IEEE Conference.

[6] Ramgovind S, Eloff M and Smith E (2010), "The Management of Security in Cloud Computing", Proceedings of IEEE Conference

[7] Naser A W S and Bin Md Fadli (2013), "Use of Cryptography in Cloud Computing", pp. 179-184, Proceedings of IEEE International Confer ence on Control System, Malaysia

[8] Rastogi Garima and Sushil Rama (2015), "Cloud Computing Implementation: Key Issues and Solution", Proceedings of IEEE Conference INDIACOM, pp. 173-179.

[9] Gentry C (2009), "Fully Homomorphic Encryption Using Ideal Lattices", *ACM Symposium on Theory of Computing*, pp. 169-178.

[10] Garima Rastogi and Rama Sushil (2015) ,"cloud computing security and Homomorphic Encryption", pp 48-58.

[11] Payal V. Parmar (2014), "Survey of Various Homomorphic Encryption algorithms and Schemes", International Journal of Computer Applications, pp 26-32.