Homomorphic Encryption Security for Cloud Computing

By J. Phani Prasad & B Seetha Ramulu

Vardhaman College of Engineering

Abstract- As the Data and Technology is increasing day by day the security and privacy issues are becoming a major concern now a days, with the advent of different security mechanisms there are some bottlenecks. In this work we are giving the essence and importance of an encryption scheme called homomorphic encryption and its related issues.

Keywords: encryption, homomorphic encryption, RSA, security.

GJCST-B Classification : K.6.m

HOMOMORPHICENCRYPTIONSECURITYFORCLOUDCOMPUTING

Strictly as per the compliance and regulations of:

Homomorphic Encryption Security for Cloud Computing

J. Phani Prasad [α] & B Seetha Ramulu [σ]

Abstract- As the Data and Technology is increasing day by day the security and privacy issues are becoming a major concern now a days, with the advent of different security mechanisms there are some bottlenecks. In this work we are giving the essence and importance of an encryption scheme called homomorphic encryption and its related issues.

Keywords: encryption, homomorphic encryption, RSA, security.

I. 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/sometimes 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.

II. 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) done 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.

III. Homomorphic Encryption

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).

Suppose if plain text is M.

Operation(M) decrypt(Operation(encrypt(M))).

In figure1 (below) homomrphic 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.

Author α: *Assistant Professor, Vardhaman College of Engineering, Hyderabad. e-mail: phanimtechcse@gmail.com*
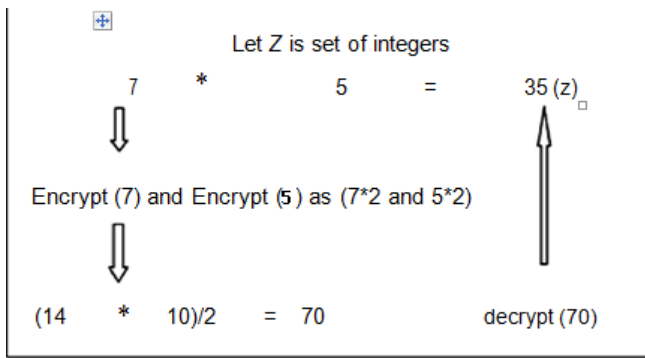Author σ: *Professor, Vardhaman College of Engineering, Hyderabad.*

Figure 1: Homomorphic Encryption on Integers

Figure 2: Homomorphic Encryption on Strings

IV. Homomorphic Encryption and Types

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.

a) 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).

Encrypt (p1* p2) = Encrypt(p1) X Encrypt(p2) (multiplicative homomorphic property.

b) 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 mod n
8. Decryption of cipher text M=c^d mod n
9. RSA follows homomorphic property as:

Encrypt(p1* p2) = Encrypt(p1) X Encrypt(p2) (multiplicative homomorphic property).

End.

V. Partial Homomorphic Encryption

Partial homomorphic encryption 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 algorithms.

VI. Implementation of RSA as PHE using a Case Study on 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.
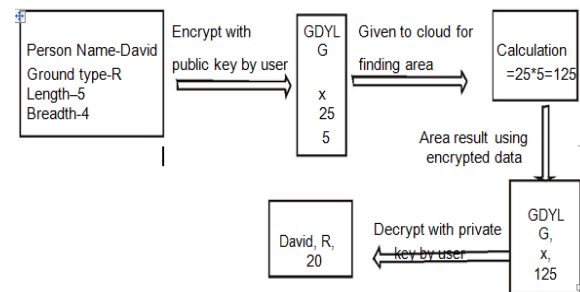

Figure 3: Computation of area in cloud

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.

VII. 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 directions has to be driven towards fully homomorphic encryption technique.

References Références Referencias

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 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 Conference on Control System, Malaysia
8. Ogburn Monique, Turner Claude and Dahal Pushkar (2013), "Homomorphic Encryption", *Proceeding Computer Science*, Vol. 20, pp. 502-509
9. Rastogi Garima and Sushil Rama (2015), "Cloud Computing Implementation: Key Issues and Solution", Proceedings of IEEE Conference INDIACOM, pp. 173-179.
10. Gentry C (2009), "Fully Homomorphic Encryption Using Ideal Lattices", *ACM Symposium on Theory of Computing*, pp. 169-178.
11. Garima Rastogi and Rama Sushil (2015),"cloud computing security and homomorphic Encryption" pp 48-58.