International Journal of Advance Research in Science and Engineering
Vol. No.6, Issue No. 05, May 2017
www.ijarse.com
IJARSE
ISSN (O) 2319 - 8354
ISSN (P) 2319 - 8346

MODIFIED HOMOMORPHIC ENCRYPTION ALGORITHM TO SECURE DATA IN CLOUD COMPUTING

Jasleen Saini[1], Roshan shrivastava[2]

[1]*Lovely Professional University,Phagwara (India)*

[2]*Asst. Professor , Lovely Professional University,Phagwara(India)*

ABSTRACT

Cloud computing offers the best platform in which data is stored and also the data is shared from one network to another. Some organizations have their own cloud to store their information. But there are some security issues in cloud and hence, Attribute-based encryption is becoming a favorable to guarantee data security in cloud computing. In this the set of attributes are used for encryption, the person who has correct set of attributes, can only decrypt the data. The homomorphic encryption is used in which mathematical operations are done on encrypted data without compromising the encryption. Homomorphic operation is done on the cipher text to make the encryption difficult so that any random third party cannot access the data to be transferred on the network. By using this attribute based algorithm having homomorphic encryption, the security aspect and the performance of the algorithm is analyzed. In modified homomorphic encryption both additive and multiplicative operations along with diffie hellman key exchange in which prime numbers are selected which is better than homomorphic encryption. This makes the encryption more difficult to decrypt by the attackers.

Keywords— Diffie Hellman algorithm, Homomorphic encryption,.

I. INTRODUCTION

1. Cloud Computing

It is a type of computing which is in the internet in which data is shared using different processing resources and share it to the other users on demand. It is architecture which is when demanded by the user access of resources which are used as a shared pool for e.g., storage, computer networks, applications, servers and services that can be provided very fast and provided with minimum organized effort. The solution for storage in Cloud computing provide various users and various enterprises having different capabilities to process and store their information in some another databases that can be present at remote areas from the user and can be in another city or in different country in the world. Cloud computing is basically a resource sharing computing to achieve consistency and to increase the level of production.

2. Cloud Computing Security

Cloud security refers to a broad set of policies, technologies, and controls deployed to protect data, applications, and the associated infrastructure of cloud computing. It is a sub-domain of computer security, network security

IJARSE
ISSN (O) 2319 - 8354
ISSN (P) 2319 - 8346

network security, and, more broadly, information security. Because of the cloud's very nature as a shared resource, identity management, privacy, and access control are of particular concern. With more organizations using cloud computing and associated cloud providers for data operations, proper security in these and other potentially vulnerable areas have become a priority for organizations contracting with a cloud computing provider. Cloud computing security processes should address the security controls the cloud provider will incorporate to maintain the customer's data security, privacy and compliance with necessary regulations.

3. Attributed based encryption (ABE)

It gives a methodology by which we can be sure that, if the storage is limited, the loss of data will only be less and minimized to extent. It efficiently binds the control of access policy for the information and the owners or other clients in case of having a server which have generally access of files. ABE can be characterized into two parts depends that whether the attributes are applicable in the cipher-text or whether the access-structure is applying in the cipher-text. The first was the Key-policy based ABE (KP-ABE) that is the initial form of attribute based encryption which was developed. In KP-ABE they modify the attributes along with the data and provide the access structure to every user as a bit of their secret key. But attribute based encryption is more suitable in the present world if the access-structure which be used in the cipher-text and the users must have their attributes present in their secret keys. The second form of ABE is called as cipher-text-policy based (CP-ABE). Both these present schemes were heavily depends on the sharing of secret scheme. This may be largely due to the fact that CP-ABE presents a simple and more suitable way to know the attributes based encryption.

4. Homomorphic encryption

It is a type of encryption which asks for the computation overhead to be taken out on ciphertext, hence providing an cipher form which, when converted into plain text, corrects the outcome of methods applied on the decipher text. This is necessary, sometimes, characteristics in today's transferring method having different compound. Homomorphic encryption will asks for the collaboration of the data joined with various applications without providing the information to any of the applications. For instance, different methods of various services from various companies can check the currency exchange tax, the rate and the shifting, on a transfer without giving it to any of the unauthorized user for decrypted data to any of those applications. Homomorphic encryption methodology is presented by design. This makes it to their use in cloud computing methodology for providing the confidentiality of processed data.

II. PROPOSED WORK

A. In this section, we describe system model for proposed scheme. In this section, we proposed the following:

- To make the data encrypted which is difficult to understand by using attribute based encryption.
- To reduce the un-authorizations by using the diffie hellman key exchange algorithm.
- To reduce the space utilizations by using the small bits of keys in the algorithm.
- To reduce the time consumption in the algorithm.
- To use the prime numbers as the prime numbers have no factors which is difficult to decrypt.

IJARSE
ISSN (O) 2319 - 8354
ISSN (P) 2319 - 8346

B. Research Methodology

The study is generally presented on to generate model for modified homomorphism disk encryption technique. The present technique will give suitable key management services and key storage. This can encompass the reliability and safety of the existing homomorphism encryption technique. In the new model, safe path development algorithm can be used for the management of key and sharing of key. The safe path development techniques are Diffie- Hellman and RSA. The Diffie- Hellman technique is very safe and suitable algorithm. In this, Diffie-Hellman technique if two users-Master and Slave wants to transfer the data. Before the start of the sharing of data, safe path is developed. Both the users choose their own arbitrary number. Based on the selected arbitrary numbers, safe path and the key is generated.

Diffie Hellman key exchange algorithm is embedded for permission process. In the cloud network, it explains the node from where the data is to send and the destination node. To develop safe path between transferring parties, every party choose a arbitrary prime factors g and n, chosen factors will become public keys for both the users. The node from where the data transfers will became master and the destination node will become slave and the master and the slave selects the private keys ‘a’ and ‘b’ simultaneously. The master checks the new number “M” from the chosen the public and the private numbers.

$M= g^a \bmod n$ (equation 1)

The Slave checks the new number “S” from the chosen public and the private factors

$S= g^b \bmod n$ (equation 2)

The Master user and the slave user share their processed value “M” and “S” through the middle nodes. So, When Slave gets “M” and the Master receives "S", then both the parties will check mode inverse value.

When the master gets the value “S” from slave and process the new value “K1” from the received “S” value.

$K1=S^a \bmod n$ (equation 3)

Slave gets the value “M” from the master user and processes the new value”K2” through the received “M”

$K2=M^b \bmod n$ (equation 4)

After processing K1 and K2, both the users develop the safe path, by processing the new key K. If both the users who are sharing data to each other, have same K1 and K2 values, the safe path is developed between the users

$K=K1+K2$ (equation 5)

The sharing of data begins between the users when safe path is developed between between the users. The transferring of data between both the parties is secured with the public keys. Both the users uses their own private keys to decrypt the sharing of data.

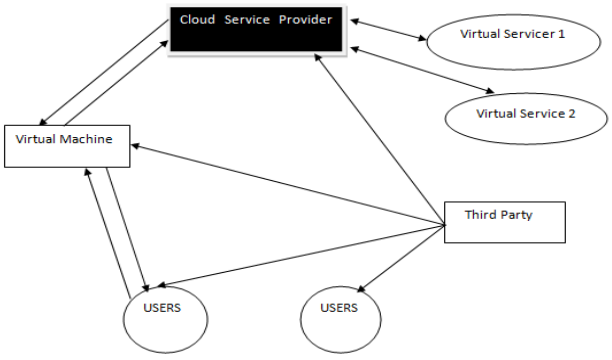

Fig 1: Cloud Service Provider

IJARSE
ISSN (O) 2319 - 8354
ISSN (P) 2319 - 8346

In fig1, a service provider cloud is shown which is related with the virtual servers bidirectionally. It is also related with the virtual machines. Hence, there are various users that are available in the network. These parties are related with the virtual machines to share the data between them. The third user is also shown in the given network. This user is joined or attached with the virtual machines, cloud service provider and the users. So, in the present work, the homomorphic encryption algorithm is used in the virtual machine. But this technique has no key transferring and the key development method. Due to this limit of this algorithm, the safety of the network is on the risk level. The chance of the attack is high in this algorithm. To solve this problem in the virtual machine, the Diffie-Hellman technique is used on the virtual machines in spite of the homomorphic encryption algorithm. In Diffie-Hellman technique, the key transferring and the key organization technique is used in it so that suitable safety is given to a network. The public and the private keys are transferred between the sender and the receiver first. Hence, after the transferring of the keys between the users then the transferring begins between the user and the virtual machine. So, it is shown that the safe path is developed between them.

In our method, Diffie-Hellman algorithm is using for the safe channel development and for the mutual authorization. In the proposed technique, only two messages are reqired to share between the two machines and a safe path will be developed. It is safer than the present authorization process. It takes low time to authorize the parties and it expands the performance of the mobile model in the topology. Diffie Hellman key exchange algorithm gives the security against the attack. In Diffie -Hellman technique, there is no privilege for the accumulation or sharing of the PIN key. So it secures the network devices from attacks.

C. Algorithm:

Selected node suppose user1

1. Login
2. Key generation
   - 2.1 Enter prime numbers
   - 2.2 Enter random numbers by client and cloud service provider
   - 2.3Secret key generation and secure channel establishment
3. OTP (One Time Password) generation
   - 3.1 cloud server will set count 1=0,...count5=0 for respective user at its side.
   - 3.2 Cloud Server will request for the OTP from user 1
   - 3.3 user1 enter (secret key+count) as OTP
   - 3.4 server match it because server knows both secret key and count of each user.
     - 3.4.1: count1++; // so for user 1 it will be count1=1; for remaining user their count will be still 0;
     - 3.4.2 if (secret_key+count(x) ==secret_key+count(y))

       { Access granted;display message by server : print "please enter the operation";}

       else{ display message by server:

       print(" wrong password, your login number is count1);}
4. client will enter the operation using HMAC digest

IJARSE
ISSN (O) 2319 - 8354
ISSN (P) 2319 - 8346

4.1: hmac(already generated secret key || v, file1,ver1 || sha1)

{if(ope==v)

{ server will check the file name and version;

if(file1,ver1== file1,ver1)

{print "file is valid";}

else{print file is invalid, please replace the file

}}

if(ope==I) { insert new file file2}

5. encryption/decryption

6. data operation

7. logout;

note: // 1.at client side, user will enter prime number, random number for generating secret keys, once generating secret key user will enter otp , after inserting otp, user will enter operation (Insertion) with corresponding file name(file1 or file2).//

III. COMPARISON WITH EXISTING TECHNIQUE WITH NEW TECHNIQUE

The key generation is small ie., the bits are small which can be stored in less space in proposed system as compare to the existing system. The small key can be selected because further prime numbers are used as public keys for encryption and no space is required for that key.

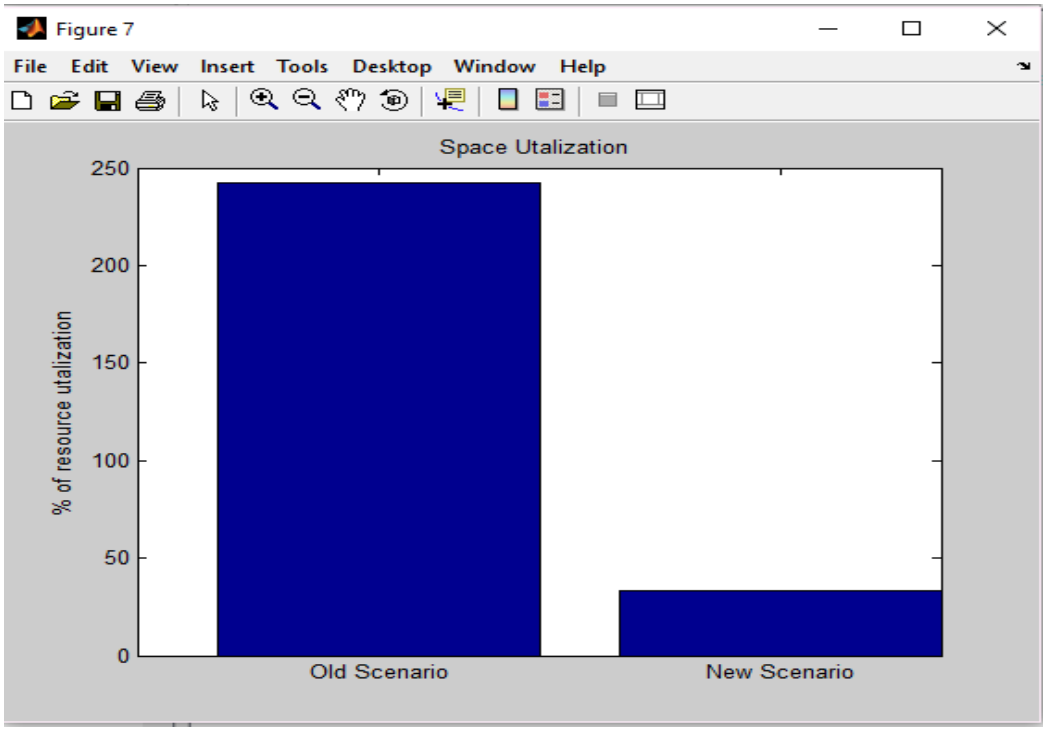

Fig2: Comparison of Space utilization

The execution time of the proposed system is less as compare to the existing system because in existing system

IJARSE
ISSN (O) 2319 - 8354
ISSN (P) 2319 - 8346

the resources which are used are more as compare to the proposed system.

The method used for the execution time is as follows:

Execution time = resources*unit(1.5)

The execution time for the existing system is:

Execution time = 2*1.5

= 3ms

The execution time for the proposed system is:

Execution time = 1.8*1.5

= 2.7ms

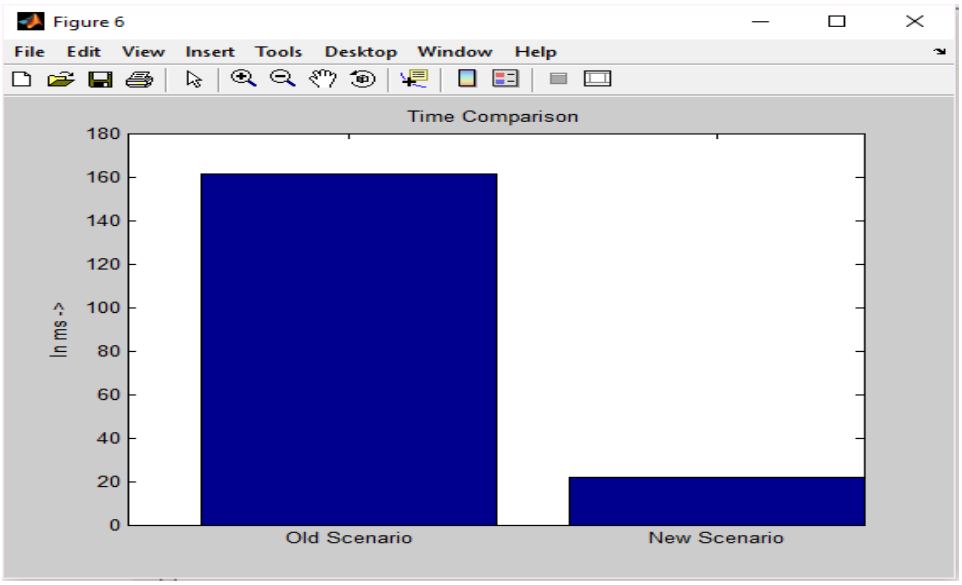

Fig3: Time Comparison

Comparison of Proposed and Existing System

Parameters	Proposed Scheme	Existing Scheme
Space Utilization	More	Less
Execution Time	More	Less
Possibilities of Attacks	More	Less

IJARSE
ISSN (O) 2319 - 8354
ISSN (P) 2319 - 8346

IV. CONCLUSION

Attribute based encryption (ABE), presents a technique through which we can make sure that if the storage is minimum, the loss of data will only be limited. It effectively makes a boundary for the policy for the access control for the data and the clients or users inspite of getting a server using access to files. It extends the scope of the computations which can be applied to process encrypted data homomorphically. Modified homomorphic includes both additive and multiplicative operations along with Diffie Hellman method in which prime numbers are used. The modified homomorphic encryption is used with this to make the algorithm difficult to understand by the attackers. This is better than the homomorphic encryption as less space is used by the keys. It can be rushed by an unauthorized party without providing the data and its internal information.

REFERENCES

[1] SU Mang, LI Fenghua, SHI Guozhen, GENG Kui, XIONG Jinbo (July 2016), "A user- centric data Secure creation scheme in cloud computing", University of science and Tech., Nanjing, China, Vol 25, N0.4.

[2] Lifeng Li, Xiaowan Chen, Hai Jiang (June 2016), "Parallelizing cipher text policy attribute based encryption for clouds", College of Info. Science and Tech., Chin,.

[3] HUANG Qinlong, MA Zhaofeng, YANG Yixian (October 2015), "Attribute based secure data sharing with efficient revocation in cloud computing", Information security center, Beijing, China, vol 24, No. 4.

[4] Xianping Mao, Junzuo Lai, Qixiang Mei, Kefei Chen, Jian Weng (November 2015), Generic and Efficient Constructions of Attribute-Based Encryption with Verifiable Outsourced Decryption", Volume: 13, pgs 533-546.

[5] A. Abbas and S. U. Khan (2014),"A Review on the State-of-the-Art Privacy-Preserving Approaches in the e-Health Clouds", Volume: 18, Pages: 1431 – 1441.

[6] Junbeom Hur (October 2013), "Improving security and efficiency in attribute based data sharing", vol 25, No. 10.

[7] Ming Li, Shucheng Yu, Yao Zheng, Kui Ren, Wenjing Lou(2013),"Scalable and secure sharing of personal health records in cloud computing using attribute-based encryption", *IEEE Transactions on Parallel and Distributed Systems*, Vol.24, No.1, pp.131–143.

[8] Junbeom Hur(2013), "Attribute-based secure data sharing with hidden policies in smart grid", *IEEE Transactions on Parallel and Distributed Systems*, Vol.24, No.11, pp.2171–2180.

[9] Junbeom Hur (2013), "Improving security and efficiency in attribute-based data sharing", *IEEE Transactions on Knowledge and Data Engineering*, Vol.25, No.10, pp.2271–2282.

[10] Kan Yang, Xiaohua Jia, Kui Ren, Bo Zhang (2013) "DAC-MACS: Effective data access control for multi-authority cloud storage systems", *Proceedings of IEEE INFOCOM 2013*, Turin, Italy, pp.2895–2903.

[11] Xun Yi, Mohammed Golam Kaosar, Russell Paulet, and Elisa Bertino (2013), "Single-database private information retrieval from fully homomorphic encryption", *IEEE Transactions on Knowledge and Data Engineering*, Vol.25, No.5, pp.1125–1134.

IJARSE
ISSN (O) 2319 - 8354
ISSN (P) 2319 - 8346

[12] Zhiguo Wan, Jun'e Liu, and Robert H. Deng (2012) A hierarchical attribute-based solution for flexible and scalable access control in cloud computing", *IEEE Transactions on Information Forensics and Security*, Vol.7, No.2, pp.743–754.

[13] Junbeom Hur and Dong Kun Noh (October 2011), "Attribute based access control with efficient revocation in data outsourcing systems", vol 22, No. 7.

[14] Junbeom Hur and Dong Kun Noh (2011), "Attribute-based access control with efficient revocation in data outsourcing systems", *IEEE Transactions on Parallel and Distributed Systems*, Vol.22, No.7, pp.1214–1221.

[15] Shucheng Yu, Cong Wang, Kui Ren, and Wenjing Lou(2010), "Achieving secure, scalable, and fine-grained data access control in cloud computing", *Proceedings of IEEE INFOCOM 2010*, San Diego, CA, USA, pp.1–9.

[16] John Bethencourt, Amit Sahay Brent Waters(2007), "Ciphertext-policy attribute encryption", *Proceedings of 2007 IEEE Symposium on Security and Privacy*, Berkeley, CA, USA, pp.321–33