International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2013): 4.438

A Framework for Secure Cloud Computing Based on Homomorphic Encryption

Ravneet Kaur[1], Harmandeep Singh[2]

Punjabi University Patiala, Punjab(India) Punjabi University Patiala, Punjab(India)

Abstract: ***Cloud computing is a paradigm in which services (Iaas, Paas, Saas) are leased to the user as per imposition. cloud computing entire data resides over a set of networked resources, this data can be accessed through virtual machines like mobiles PC etc. Cloud computing reduces hardware , maintenance and installation cost. But security is major issue that prevents users for cloud computing. When we relocate data to cloud we use standard encryption technique to secure the data. But when we have to do computations on data hoarded on cloud then we have to decode data i.e. provide private key every time which is not a secure method. In this paper we have proposed a technique that enhance two security features of the cloud. One is authenticity and another is data security. Normally we check authenticity by user name and password, but here we have added one more parameters i.e. MAC address. IP address is used get location of users.Data security is improved by applying homomorphic operations on DES algorithm to encrypt the data at server side. This modified DES algorithm proves to give better results than DES algorithm in terms of performance.***

Keywords*:* Cloud computing, Homomorphic Encryption, DES algorithm

1. Introduction

So our main area of concern is "whether the confidential data kept on the central storage on cloud is secure or not". We use various cryptographic techniques (symmetric or asymmetric)to encrypt our data so that our data reaches to destination safely. In private key cryptography if two parties wants to securely communicate then they should share a secret key .In order to encrypt and then decrypt the data, both parties must have same key. So all the secure communication is limited to those who have a pair of keys. But there is a problem with this technique that how to securely communicate to exchange key , if key has to change from time to time before securely transmitting the message. Public key cryptography solved this problem. In this security relies on hard mathematical problems and each party need a public key for encryption and private key for decryption [1].

Additionally two parties wants to be sure about confidentiality and integrity. For each of these constraints appropriate solution is devised and implemented. However ,only a couple of months after the publication of RSA algorithm paper[2], Rivest et al. asked the question whether it is possible to work with encrypted data, without the need of decrypting it first[3].That question started the research for homomorphic encryption systems. Why we needed this type of systems e.g. we have some confidential information and we send it to servers for some computation and we do not want to give private key to server .So homomorphic encryption can help to preserve this privacy policy.

Homomorphic encryption allows to perform operations on cipher text without knowing the private key and result is same as operations performed on raw data. If encryption scheme is homomorphic then cloud can still perform meaningful computations on data, even though it is encrypted. So in this way it can guarantee the security of cloud computing data. In this paper we have tried to resolve two security issues of cloud i.e. authenticity and data security. Main focus is on data security. We have applied binary homomorphic operations on DES algorithm to enhance the data security, in this way we have modified basic DES algorithm and compared its performance with simple DES algorithm on basis of various factors like Throughput, Encryption time

Decryption time, Cipher text size. We have maintained log tables which store information of each user weather authorized or unauthorized .we will discuss it later in this paper. We have improved authentication mechanism by comparing ipaddress at time of login to get location of user.

2. DES Algorithm

DES is a block cipher. It encrypts the data in a block of 64 bits. It produces 64 bit cipher text. The key length is 56 bits. Initially the key is consisting of 64 bits. The bit position 8, 16, 24, 32,40,48,56, 64 discarded from the key length. Basic working of DES algorithm is shown in fig. 1

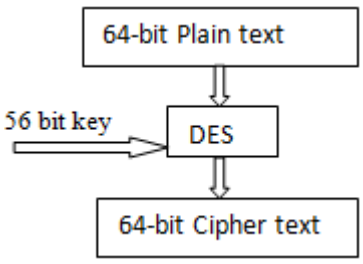

Figure 1: Basic working of DES Algorithm

DES is based on two fundamental attributes of cryptography: Substitution (confusion) and transposition (Diffusion). DES consists of 16 steps, each of which called as a Round. Algorithm[4]. In the first step, the initial 64-bit plain text block is handed over to in Initial Permutation (IP) function[5].The Initial permutation is performed on plain text[6].The initial permutation produce two halves of permuted block: Left Plain text (LPT) and Right Plain Text (RPT) [7] . Now, each of LPT and RPT goes through 16 rounds of encryption process, each with its own key. Detailed working of DES algorithm is shown in fig. 2

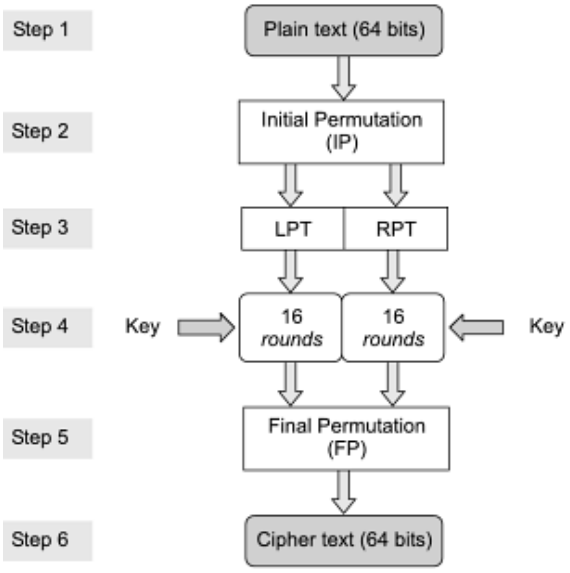

Figure 2: Detailed working of DES Algorithm

a) From the 56-bit key, a different 48-bit Sub-key is generated using Key Transformation.
b) Using the Expansion Permutation, the RPT is expended from 32 bits to 48 bits.
c) Now, the 48-bit key is XORed with 48-bit RPT and resulting output is given to the next step.
d) Using the S-box substitution produced the 32-bit from 48-bit.
e) These 32 bits are permuted using P-Box Permutation.
f) The P-Box output 32 bits are XORed with the LPT 32 bits.
g) The result of the XORed 32 bits are become the RPT and old RPT become the LPT. This process is called as Swapping.
h) Now the RPT again given to the next round and performed the 15 more rounds[8]. After the completion of 16 rounds the Final Permutation is performed[9][10].

3. Cloud Computing, Its Security Issues And Homomorphic Encryption

Definition [11]:By cloud computing we mean: The information technology (IT) model for computing, which is composed of all the IT components (hardware, software, networking, and services)that are necessary to enable development and delivery of cloud services via internet or a private network.
Cloud computing is basically an idea that data and programs can be stored centrally on cloud and then can be accessed from anywhere in the world through internet using PC's, laptops or phones.
In cloud computing available service models are:
1) Infrastructure as a service
2) Platform-as-a-service
3) Software as a Services

Four deployment models have been identified for cloud architecture solutions, described below:
1) Private cloud
2) Community cloud
3) Public cloud
4) 4.Hybrid cloud

4. Cloud Computing Security Issues

Cloud computing provides us an advantage that data can be stored centrally in the cloud and can be accessed from anywhere and anytime. This brings many advantages, including data ubiquity, flexibility of access, and resilience. Since cloud computing keeps data outside the control of owner so it introduces security issues also .
This facility raised various security questions like privacy , confidentiality, integrity etc.
Some main issues of cloud computing are:
1) Access control: Cloud based data is usually accessed by many insecure protocols and API's over the public network.
2) Data location: User don't know the way user data is stored, where it is stored, data recovery, data encryption and data integrity problem.
3) Authentication: User cannot be classified as no. of users as user changes dynamically as well as user use different resources. The cloud is typically a shared resource, and other sharers (called tenants) may be attackers.
4) Data security: In cloud computing model, a cloud service provider has a major role to play. He has the right to access anything which is stored on cloud.
5) Data recovery: Don't know where data is located so Cloud service provider must tell what will happen to data in case of disaster and how it will be recovered.

Now the question arise "How to keep client private data confidential?."

Therefore, the genuinely unique challenge posed by cloud computing security boils down to just one thing: the data in the cloud can be accessed by the cloud provider. The cloud provider as a whole (or its employees individually) can deliberately or inadvertently disclose customers' data. Moreover, the cloud provider may have subcontractors (typically, a "software-as-a-service" provider will subcontract to an "infrastructure-as-a-service" provider), and the subcontractors may also have access to the data. This paper addresses the question of how a customer could secure its data from malicious or negligent cloud providers.

So we can protect data by encrypting it before sending it to the cloud provider, But to do calculations we have to decrypt it every time. Until now it was impossible to encrypt data and trust a third party to keep them safe and able to perform distant calculations on them, So to allow cloud providers to perform computation so encrypted data without decrypting them requires using the cryptosystems based on Homomorphic encryption.

5. Homomorphic Encryption

Homomorphic Encryption systems are used to perform operations on encrypted data which is kept on the cloud without knowing the secret key (without decrypted), the client is the only possessor of the secret key. When we decrypt the result of the operation, it is the same as if we had carried out the calculation on the raw data. Definition: An encryption is homomorphic, if: from Enc (a) and Enc(b) it is possible to compute Enc(f(a, b)), where f can be: +, ×, _ and without using the private key .

Among the Homomorphic encryption we distinguished according to their operation to assess on raw data. The additive Homomorphic encryption (addition of the raw data) is the Pailler [12] and Goldwasser-Micalli [13] cryptosystems and the multiplicative Homomorphic encryption (only products on raw data) is the RSA [14] and ElGamal [15] cryptosystems.

6. Proposed System

Our work require collecting evidences about the users who have registered in our cloud, authenticated users who have logged in and unauthorized users who tries log in .All information about these users is maintained in log tables. Log table contain fields as shown in following table 1.

Table 1: Fields in Log table

FIELDS	DESCRIPTION
User name	Username for login
Password	Password for login
Macaddress	Mac address of a machine from which person has logged in
Ipaddress	It will tell the location and network from where person has logged in. It is also used for authentication purpose
Location	We can get location of person through ipaddress
Login time	time at which person has logged in
Logout time	Time at which person has logged out.
Time spent	Total time spent by person
Authentication	It tells weather the user has successfully logged in or is unsuccessful.

So this log table is useful in knowing the nature of users. We can detect intruders if they try to access through new added security layers of Macaddress and IP address. We have applied homomorphic encryption on simple DES algorithm. We have compared this modified DES with simple DES algorithm on basis of some performance measures like throughput, ciphertext size, encryption time, decryption time. We have compared both projects on basis of four factors i.e. Throughput, Ciphertext size, Encryption time, Decryption time. Flowchart of proposed work is shown in fig. 3.

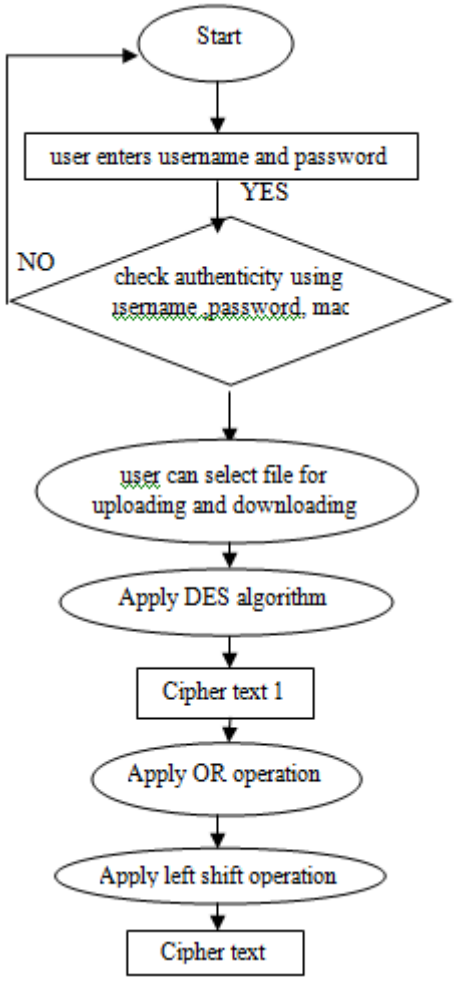

Figure 3: Proposed work flowchart

1) **Throughput:**It is number of bits transferred per unit time . Its unit is byte/sec.
   Throughput = uploaded file size ÷ Delay time.
   Delay time is time difference between start uploading time and end uploading time.
2) **Encryption time:** Time taken by server to encrypt any file. Its unit is nano seconds.
   Encryption time = Encryption end time - Encryption start time.
3) **Ciphertext size:** It is length of encrypted data. Its units is in bytes.
   Ciphertext size = Length of encrypted data
4) **Decryption time:** It is time taken by server to decrypt any file. Its units is nano seconds.
   Decryption time = Decryption end time - Decryption start time.

7. Experimental Results

This system has been designed and implemented in java language to enhance security between client and server. The table below represents experimental results of comparison of both the algorithms. We have uploaded file of 2.8 MB and its corresponding factors for both the algorithms in shown in below table 2 and graphs.

Table 2: Comparison of SHE DES and DES

Algorithms → / Factors ↓	Modified DES	DES
Encryption Time	281	474
Cipher Text	2882	2931
Throughput	7565	2931
Decryption time	327	486
File size	2.8	2,8

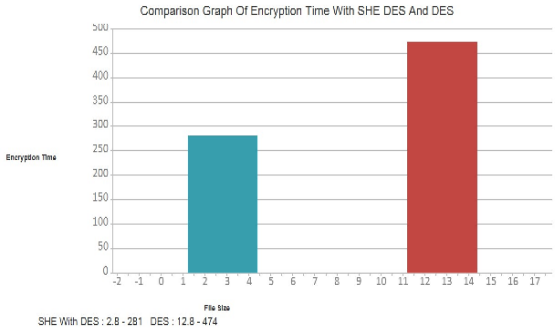

Figure 4: Comparison of Encryption time and file size.

Graph in fig 4 represents, a file of size 2.8 MB takes 281 ns encryption time using proposed algorithm(SHE DES) which is lesser than DES encryption time of 474ns.

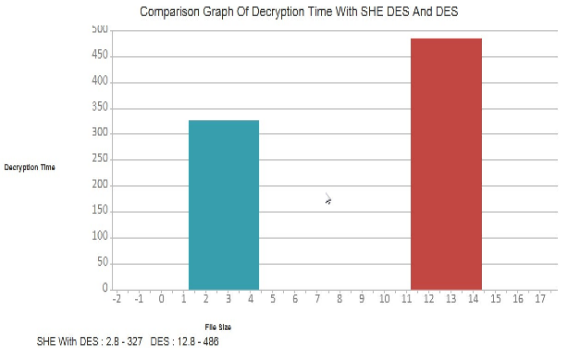

Figure 5: Comparison of Decryption time and file size.

Graph in fig. 5 represents, a file of size 2.8 MB takes 327ns encryption time using proposed algorithm(SHE DES) which is lesser than DES encryption time of 486ns.

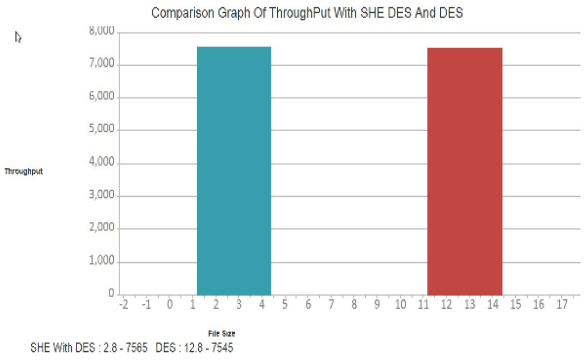

Figure 6: Comparison of throughput and file size

Graph in fig 4 represents, for a file of size 2.8 MB throughput is 7565 bytes/sec using proposed algorithm(SHE DES) which is more than DES throughput of 2931 bytes/sec.

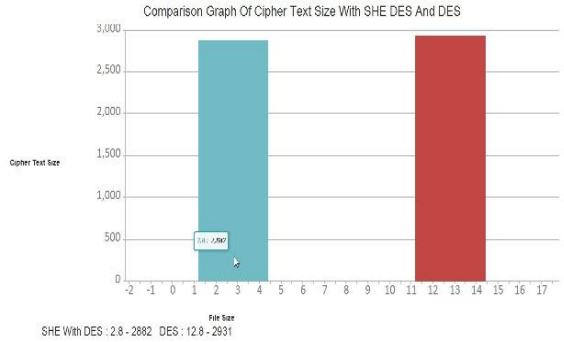

Figure 6: Comparison of ciphertext size and file size.

Graph in fig 6 represents, for a file of 2.8 MB cipher text size is 2882 bytes using proposed algorithm(SHE DES) which is lesser than DES ciphertext size 2931bytes.

The criteria in performance of every algorithm is that encryption time, decryption time, and cipher text size should be less and throughput should be more. During comparison of SHE DES this criteria is achieved and SHE DES give better results.

8. Conclusion and Future Scope

This paper represents enhanced the security features of the cloud. Cloud computing is an emerging area within the field of information technology. Security of data is main issue that hamper its growth. We have worked on two parameters of security, one is authenticity and another is data security. For authenticity here when user enters username and password then ipaddress and macaddress signature also get checked and stored at backend in encrypted form. So in this way we have enhanced the security mechanism.

We have maintained a log table(login time, logout time, ipaddress, Mac address, authentication etc) where we maintain entries of authorized users and unauthorized users also who tried to logged in. from this log table we come to know about session of logged in users and information about unauthorized users i.e. macaddress of their machine, ipaddress.

Data security is improved by applying homomorphic encryption technique to encrypt the data at server side. we have used binary operations OR gate and left shift operator. We have applied these operations on DES algorithm and computes various factors of this modified DES algorithm like throughput, Encryption time, Decryption time, Cipher text size. We also compute same factors of simple DES algorithm. For uploading a file of 2.8 MB above table 2 shows that modified DES give better results by showing more throughput, less encryption time, less decryption time and less cipher text size. Hence it is conclude that enhanced DES algorithm give better security and performance.

We have implemented homomorphic encryption on DES algorithm only. In future work these binary operations of homomorphic encryption can be applied to other algorithms also and results can be compared to analyze which algorithm provide better security in cloud with better performance. This somewhat homomorphic encryption can also be converted into fully homomorphic encryption.

References

[1] J. Katz and Y. Lindell (2007), “Introduction to Modern Cryptography (Chapman& Hall/Crc Cryptography and Network Security Series)”, Chapman &Hall/CRC.
[2] R. Rivest, A. Shamir, and L. Adleman (1978), “A method for obtaining digital signatures and public-key cryptosystems,” Communications of the ACM, vol. 21, pp. 120–126.
[3] R. L. Rivest, L. Adleman and M. L. Dertouzos (1978), “On data banks and privacy homomorphism,” Foundations of Secure Computation, Academia Press, pp. 169–179.
[4] M. E. Hellman(1979), "DES will be totally insecure within ten years", lEEE Spectrum.
[5] Alani, M.M.(2010), " A DES96 - improved DES security ", 7th International Multi-Conference on Systems, Signals and Devices.
[6] Seung-Jo Han , Heang-Soo Oh , Jongan Park (1966)," IEEE 4th International Symposium on Spread Spectrum Techniques and Application Proceedings ".
[7] Manikandan. G, Rajendiran.P, Chakarapani.K, Krishnan.G, Sundarganesh.G (2012),"A Modified Crypto Scheme for Enhancing Data Security", Journal of Theoretical and Advanced Information Technology.
[8] Shah Kruti R., Bhavika Gambhava (2012),"New Approach of Data Encryption Standard Algorithm", International Journal of Soft Computing and Engineering (IJSCE).
[9] William Stallings(2005), " Cryptography and Network Security Principles and Practices", Prentice Hall.
[10] Sung-Jo Han, Heang-Soo Oh, Jongan Park(1996), "The improved Data Encryption Standard (DES) Algorithm",Department of Electronic Engineering, Chosun University. South Korea, IEEE.
[11] Vic (J.R.) Winkler (2011), “Securing the Cloud, Cloud Computer Security, Techniques and Tactics”, Elsevier.
[12] Pascal Pailler (1999), “Public-key cryptosystems based on composite degree residuosity classes”. In 18th Annual Eurocrypt Conference (EUROCRYPT'99), Prague, Czech Republic .
[13] Julien Bringe and al.(2007), “An Application of the Goldwasser-Micali Cryptosystem to Biometric Authentication”, Springer-Verlag.
[14] R. Rivest, A. Shamir and L. Adleman (1999), “A method for obtaining digital signatures and public key cryptosystems”. Communications of the ACM, 21(2) :120-126, 1978. Computer Science, pages 223-238, Springer.
[15] Taher ElGamal (1985), “A public key cryptosystem and a signature scheme based on discrete logarithms”. IEEE Transactions on Information Theory.