# A New Cloud Security Technique using Data Partition and Homomorphic Encryption

**Sudesh Sharma[1], Karambir[2]**

[1] Department of Computer Engineering U.I.E.T, Kurukshetra University Kurukshetra, India

[2] Department of Computer Science and Engineering U.I.E.T, Kurukshetra University Kurukshetra, India

## ABSTRACT

***Cloud storage is rapid increasing fact nowadays so cloud data security becomes a more concerned matter at processing stage of data .whenever users want to operate on data stored on un-trusted cloud then it is main concern that their data would be confidential during operation and faster uploading and downloading of data. So in this paper we are using homomorphic encryption which is asymmetric encryption and provides operation on encrypted data giving same result as obtained by operating on plain data. Out of several homomorphic algorithms EcElGamal algorithm has been chosen to encrypt the data which provides high security using less key size and proposes a new approach data partition based EcElGamal. Data Partition refer to dividing the data in blocks of some size in bytes or bits, by this we are reducing load on single server and provides better performance.***

**Keywords:** cloud data security, research issues, cloud security issues, Homomorphic Encryption (HE), EcElGamal, Data Partition.

## 1. INTRODUCTION

Cloud computing is an era of development in computing by providing plenty of services to cloud users. Cloud is gaining popularity among enterprises, business, institutions etc. Today the world of internet in which each and every activity based on internet like most of people are consulting their mail online through webmail clients, writing mutual credentials using web browsers, creating virtual albums to upload their photos of the holidays. They are operating applications and storing data in servers located in Internet and not in their own computers. Something as simple as come into in a web page is the simply way that user needs to begin to use services that reside on a distant server and lets him share private and confidential information, or by means of computing cycles of a stack of servers that could be seen by user's own eyes. And every day there is use of more these services that are called cloud computer services. There are various commercial products of cloud computing like Amazon EC2, Microsoft windows azure platform, Google app engine etc.

### 1.2. Various Research Issues for Cloud Computing

**There are various research issues exist in cloud computing from which some of them we are discussed below [1]:**

a) **Availability:** This is a major concern to all the organizations using cloud services that whether the utility computing service will be capable to provide high or adequate availability of the service or resources agreed for. And for cloud provider this is a worry as they have to prevent "single point of failures" to maintain high availability, for which, the most appropriate step is to allocate the data among several cloud vendors or providers.

b) **Data Lock-In/Interoperability:** Lack of the standardization of data storage and data dispensation techniques makes it difficult for a user to move from one platform to other thus, ensuing in a Lock-in which although attractive to a cloud provider is a bottleneck for the customer. Thus it is required to standardize the APIs so as a service or storage can be deployed across multiple vendors.

c) **Data confidentiality/Security:** is an important issue for all the parties involved in a cloud service whether a provider, customer or the third-party, as a large quantity of sensitive and confidential data resides in cloud which is a source of enormous valuable information. Data in cloud needs to be secured from both the outside and inside attackers.

d) **Data Communication Bottlenecks:** Modern applications are more data intensive as compared to the earlier ones, for which, data is required to transferred across the boundaries of cloud swiftly So, to transfer such large amounts of data at a high speed cloud vendors have to take in considerations the traffic and the communication overhead in terms of cost.

e) **Performance:** I/O performance rather than memory performance are more serious issues in a cloud computing environment, if these are matched with in the traditional computing practices. Also the appropriate scheduling policies in a virtualized environment concern the performance to a great extent.

f) **Scalability:** Scalability can be discussed in terms of Storage or Service. Scalable storage points to the cloud definition which presents cloud as an infinite capacity store and Pay-as-you-go rule. Thus, to provide scalable storage cloud should be proficient to scale up/down to the storage demands of the customer. Talking in terms of services the cloud should be capable to handle automatically scale up/down according to the load or the requirement of customer whether of processing capability or of resources.

### 1.3. Various Security Issues

a) **Data Security:** The endeavor data is stored outside the endeavor boundary and also needs additional security checks to ensure data security [2].

b) **Network Security:** All data stream over the network needs to be secured, to prevent outflow of sensitive information [2] and DoS Attack is also a network security issue which occurs in cloud computing when the target system gets overloaded with service requests by any intruder and as a result the system will not be able to process further requests send by any intended user. The resources will be unavailable for intended user to process the requests.

c) **Data Access:** A company will have its own security policies based on which each employee can have right to use a particular set of data. These security policies must be adhered by the cloud to avoid intrusion of data by unauthorized users [2].

d) **Data Segregation:** Data of several users will be resides at the same location; Intrusion of data becomes possible in this environment [3].

### 1.4. Data Security on Cloud

As data are stored on cloud in huge amount so security of that data is main concern for cloud provider from outside attackers. Encryption is best way to make data storage on cloud secured, mainly encryption is of two types that is symmetric and asymmetric encryption. It is the technique for converting the plain text message or information in the coded form so that only legitimate user can access and read the plain message.

a) **Symmetric encryption:** In this system encryption as well as decryption both can be performed with the single key (private key) example- AES (Advanced Encryption Standard). Main issue is to share the secret key whenever some calculation required on data.

b) **Asymmetric encryption:** In relation with previous scheme, this scheme introduces a key pair one is to encrypt and another to decrypt. The encryption key is public, as the decryption key remains private.

c) **Homomorphic Encryption :** It lies in the asymmetric encryption but it have additional property of homomorphism which allows it to perform operations on encrypted data by providing confidentiality of that data at process time because it would not be decrypted for calculations. The basic operations performed by homomorphic algorithms are addition and multiplication. Because with these operations on encrypted bits we can calculate any function on encrypted bits.

#### 1.4.1. Some homomorphic algorithms are RSA, Paillier , ElGamal , EcElGamal.

a) **RSA:** given by R. Rivest, Adi Shamir, and Leonard Adlemen in 1978. It is asymmetric algorithm supports multiplicative homomorphism. Minimum key length for RSA [4] is 1024 bit.
Paillier: Pascal Paillier, the French mathematician, has proposed the new cryptographic algorithm named Paillier Cryptosystem Algorithm [5] in 1999. It has an additive homomorphic property.

b) **Elgamal:** Elgamal encryption algorithm is proposed by the Taher Elgamal in 1984[6]. Elgamal encryption algorithm is public key algorithm and is multiplicative homomomorphic.

c) **Elliptic Curve Elgamal:** Elliptic Curve Elgamal was introduced as more beneficial than the ElGamal .it is an asymmetric cryptographic encryption technique using finite points on elliptic curve and the security depends on the size of key. Elliptic Curve Cryptography (ECC) provides high level security using smaller key size and easy implementation depending up on discrete logarithmic problems. A 256-bit ECC is considered to be equivalent to 3072-bit RSA [7].

## 2. RELATED WORK

Craig Gentry [8] proposed the first FHE scheme, solving a unfasten problem on ideal lattices. Deyan Chen and Hong Zhao [9] mainly focused on data safekeeping and confidentiality protection issues which was big problem faced in cloud computing at the enterprise level. Maha Tebaa et al. [10] employed a security method called HE, which perform the operations on encrypted data without decrypting that data hence, does not required for raw entries and thus increase confidentiality of data. Simon Fau et al. [11] proposed a first evaluation towards the practical use of (FHE) to perform true calculations. W. Wang et al. [12] overcome the performance bottleneck of the Gentry, Halevi FHE by introducing algorithmic optimizations. FENG Chao and XIN Yang [13] suggested Gentry-style HE scheme as a slow key generation algorithm and proposed key generation algorithm by choosing eigen values of primary matrix that enables one to create a more practical partial HE. G. jeeva Rathanam and M. R. sumalatha[14] designed a secured storage system in which data was stored on server by dynamic data operation with Partitioning Method and Improved Adaptive Huffman Technique and Improved RSA Double Encryption Technique . Kamal Kumar Chauhan et al. [15] concerned on many standard encryption and suggested fully homomorphic and partial homomorphic methods were not feasible and not so easy to implement for cloud computing. Ali Azougaghe et al. [16] mainly focused on some threats in cloud and presented a simple more generalized security holding architecture for inter-cloud data sharing. The security to data was provided by using algorithms (AES) Advanced Encryption Standard) and Elgamal [4]. Babitha.M.P and K.R Remesh Babu [17] presented a simple secure system for storage of data on cloud using 128 bit AES algorithm and for authentication an SMS alert method was used which provide an alert SMS to the registered user's number if any other person tried to fetch his/her data file. Yasmina Bensitel and Rahal Romadi [18] focused on cloud computing and its adoption in different domain, and described the role of HE technique for privacy preserving data sharing in the cloud. It might be either additive or multiplicative homomorphic Therefore, proposed a system that ensures secrecy of data by using partial HE algorithms RSA [2] and Paillier [3].

## 3. SYSTEM MODEL

### 3.2. Problem Statement

The data storage in cloud is easy and provides large space for data but it store data at remote locations. In cloud the data can be fetched at anytime from anywhere. But the security of data and confidentiality of secret data during communication have to consider when using cloud services. Cloud storage provides several benefits to its customers at minimum cost and efforts but even though there exists some security issues. Among all of issues one important issue is the confidentiality of user's data and another is performance for uploading and downloading the data on cloud. One common solution to maintain confidentiality is encryption but for processing speed as well as bandwidth utilization there must use fast encryption algorithm. In cloud computing where large amount of data stored there may be need to operate on data according to customer demand then it is necessary to use an efficient secured environment which can compute on encrypted data and also which can reduce overhead on cloud server due to large files. All these objectives can be fulfill by our new approach of data partition based EcElgamal which makes faster the computation and also uploading and downloading of files and provides high security with less key size. It uses elliptic curve thus produce small cipher text and creates discrete logarithmic problem for an intruder.

### 3.3. Use of Homomorphic Algorithm

The homomorphic algorithm follows two main properties that is either these are additive homomorphic or multiplicative homomorphic [10] . The proposed algorithm we are using can be additive or multiplicative .

Additive property: according to this property addition operation is performed on encrypted data. let us consider bob and alice are in communication then bob who had stored his data in encrypted form to the alice server .now if bob wants some computation on his data he said to alice to do it, using homomorphic encryption it is easy to carry computation on encrypted data .

$$EP\ (M1+M2)= EP\ (M1)+EP\ (M2)$$

EP denotes public key for encryption ,M1 and M2 are plain data,+ denotes homomorphic addition operation.

Multiplicative property: in this multiplication operation is performe on encrypted files.

$$EP\ (M1*M2)= EP\ (M1)*EP\ (M2)$$

EP denotes public key for encryption, M1andM2 are plain data ,* denotes homomorphic multiplicative operation.

Working of EcElGamal: Using EcElgamal some elliptic curve C will be used on which some finite group of points lie.C= E mod P where E is elliptic curve and P is any large prime if bob and alice wants to communicate then bob chooses some point q on E that is public and also choose some secret A which is always kept his side.bob has no need to share his secret with alice. Bob computes another point B =q*A. alice choose some random integer K .
Alice: y1=Kq, y2= M+KB where M is original message , y1and y2 are encrypted form sends to bob by alice.
Bob: . Bob then decrypts (y1,y2) using secret A.

$$M=y2-Ay1$$

### 3.3.1. Why EcElgamal Encryption?

Elliptic Curve ElGamal was introduced as more beneficial than the ElGamal .It is an asymmetric cryptographic encryption technique using finite points on elliptic curve and the security depends on the size of key. Simple ElGamal algorithm was invented in 1985 by Taher ElGamal. It is used as asymmetric cryptographic to encrypt symmetric key since it is not so efficient because it takes large bit size that is at least 1024 bit and also time consuming. Therefore EcElGamal algorithm is more advantageous than other algorithms as it uses elliptic curves due to which the size of cipher text can be reduced significantly. For example-any traditional cryptosystem like RSA takes the minimum key size 1024 bit for any file and for the same the EcElGamel reduced the key size of 160 bit. Elliptic curves are used on finite integers which range in a finite domain. ECC in analog of ElGamal as it uses the group of points on an elliptic curve which are easy to find so it will reduce the time for encrypting and decrypting the data also it is homomorphic algorithm so it works on encrypted data thus producing high confidentiality and less bandwidth utilization because when operations are carried out on file's data if this algorithm is used it does not required secret key from user /data owner so the data would not attacked by any intruder and after calculation the encrypted result will be stored in cloud database whenever user want to retrieve it CSP (Cloud Service Provider) send it to user in encrypted form and thus user can decrypt it by secret key and the result will be same as it would obtained by performing operation on plain data . It provides same security as RSA but in less key size and reduced cipher text thus more beneficial than other cryptographic algorithms.

### 3.3.2. Data Partition Algorithm

This algorithm is designed to outperform the task of dividing the data in to blocks of size in bits and then these blocks are encrypted using same key of EcElGamal algorithm and after this these blocks are stored individually in cloud database. This procedure is carried out to lessen the load on cloud server due to large files because it is difficult to maintain the large file's encryption as a whole and also expensive. In proposed work blocks of size $2^{10}*10^{2}$ bits have been taken.

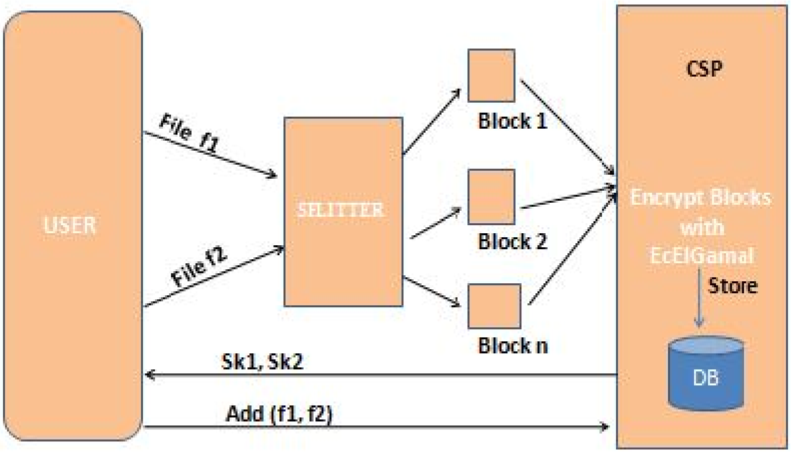

**Figure 1** Block Diagram of Addition Operation on Files using Data Partition based EcElGamal Encryption

**USER:**

Step1. User sends plain files f1 and f2

Step2. Intermediate step involved by Splitter on file f1 and f2 to divide them in to blocks of size in bits. These blocks then transfer to cloud provider.

Step3. User demands for addition operation on files.

**CSP (Cloud Service Provider):**

Step1. Obtain blocks of files and encrypt these blocks using common key of EcElGamal algorithm i.e. public key pk1 and pk2. Merge them to obtain encrypted files Ef1 and Ef2 on which computation has to be performed.

Step2. Generated secret keys sk1 and sk2 for files f1 and f2 are sent to the user to keep them private to the user side.

Step3. Obtain the addition operation request from user

Step4. Process the addition on encrypted files without sharing the secret keys from user and produce file Ef3 which is encrypted and stores the result of two encrypted files – Ef3=Ef1+Ef2

Step5. User can retrieve this Ef3 file which stores the result of computation which user demand for on his files .the result will be same as the user obtain from applying operation on plain files.

**Analysis:** steps involved at cloud server are greater for symmetric encryption in which server required for secret key each time whenever some operation has to be applied on files and the files which are stored encrypted thus decrypted by using secret key so there can be intrusion or confidendility of data can be disturbed due to sharing of secret key thus a better security method using homomorphic algorithm that is EcElGamal in Figure 1is introduced which reduces the number of steps on cloud server side because operation is carried on encrypted files and provide the same result as the user desired to produced by performing on plain data. Load on server get reduced using Data Partition Algorithm which generated blocks of files which are encrypted using EcElGamal and then the final encrypted result stored in cloud database (DB).

## 4. PROPOSED ALGORITHM

In this section we are providing the detailed algorithm which is used to encrypt the data on cloud . A new approach of Data partition based EcElGamal algorithm is used for providing better security and less time consuming during encryption and decryption of files on cloud whenever users upload and download the files. This will gives better performance results than simple EcElgamal and other homomorphic algorithms.

### 4.1. File Upload

Whenever any file F is uploaded it will include partition of file in to blocks by splitter in some part size in bit.These parts are then encrypted by using algorithm EcElGamal and encrypted parts will be send to the cloud storage.
The algorithm uses the set of following functions:
NUM_ PARTS (F): Return number of parts of file F.
ENC_EcElGamal (P, Pk): Encrypt each part with EcElGamal algorithm with public key Pk.
Send _to_Cloud(P'): send the encrypted parts to the cloud storage

```
Algorithm 4.2.1 ENC_File(F)
    Send PLAIN File F
    /*Algorithm to partition the file by splitter*/
    /*Algorithm to encrypt each parts of file */
    /*to transform clair text of each part from P by EcElGamal algorithm to cipher parts P' 6.*/
    for P ← 1 to NUM_ PARTS (F) do
      P'=ENC_EcElGamal (P, Pk)
    Send _to_Cloud(P')
    Save the P' in DB
    End for
```

### 4.2. File Download

This algorithm got two phases ; in the first phase, the algorithm decrypts each parts from P'. in the second phase ,merge the parts to get the original file F.
The algorithm uses the set of following functions

NUM_PARTS(F): This function returns number of parts of file F
DEC parts_EcElGamal(P’,Sk): decrypts each parts with secret key of EcElGamal algorithm.
Merge_parts_save(P.length): merge all parts and get the original file F.

```
Algorithm 4.3.1  DEC_File(P’)
/*algorithm to decrypt the file stored in parts from cloud storage*/
/*to transform each part’s cipher text in P’ in to plain text in P*/
/*phase 1:Decrypt  each cipher part from P’ by EcElGamal algorithm*/
For P’ from 1 to size of (P’) do
    P=DEC parts_EcElGamal(P’, Sk)
End for
/* phase 2: merge all parts to get original file Mfile same as file F*/
Merge_parts_save(P.length)
For P from  1 to parts.size do
    Whole=parts.get
    Cnt+=P.length
    Mfile =write(whole)
    Return  (Mfile)
End for
```

### 4.3. File Operation

Whenever user wants to do some operation on two homogenous files stored on cloud then using homomorphic algorithm EcElgamal it would not be required to decrypt the files for processing on it. Consider two files f1 and f2 of same length stored on cloud database and contained some numeric data on which user wants addition operation then additive EcElGamal algorithm can be used to perform operation on encrypted data and the result obtained would be same as user required.

For this the set of following functions being used:
Generate_EcElGamal_key pair (kp): This function returns key pair for EcElGamal algorithm i.e.private key and public key.
**Enc_file1:** This function encrypt the file1 content and return encrypted file f1’
**Enc_file2:** this function encrypt the file2 content and return file f2’
Enc_Add(f1’, f2’): This function process the addition operation on encrypted files and return encrypted result which are stored in file called as Ef3.

```
Algorithm 4.4.1  Homo files_operation(f1, f2)
/*Send plain file f1 and file f2 to cloud provider by user and then */
/* cloud provider generate key pair of ecelgamal algorithm */
/*and encrypt the files f1 and f2 plain content*/
/*using EcElgamal public keys pk1 and pk2 for files f1 and f2 respectively*/
/*and obtain the encrypted files f1’ and f2’*/
f1’=Enc_file1(f1, pk1)
f2’=Enc_file2(f2, pk2)
/*apply addition operation on encrypted files*/
/*and obtain resultant encrypted file Ef3 containing*/
/*operation results in encrypted form*/
Enc_Add(f1’, f2’)
Ef3=f1’+f2’
/*Store Ef3 in cloud database*/
```

## 5. RESULTS AND ANALYSIS

The implementation is done in java language using net beans 8.1 text editor and the results are obtained on hardware pc of hp intel Core i5 processor having 8 GB RAM. In Figure 2 the files used are small file is in KB file size, medium and large are in MB file size. Results are obtained on text content files for Figure 2, Figure 3, and Figure 4 but also work for other files.

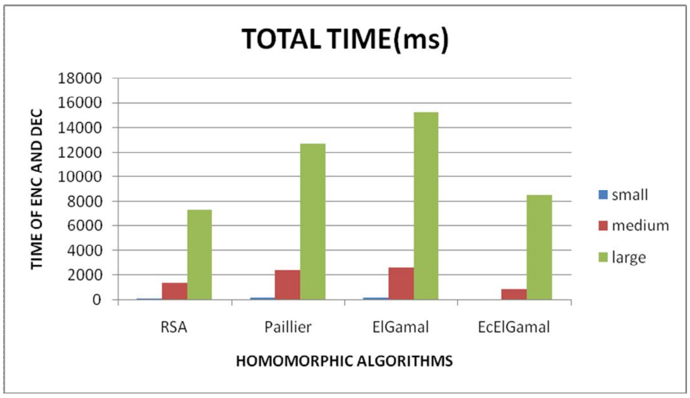

**Figure 2** Comparison of Various Homomorphic Algorithms on different file sizes

From this Figure 2 it is concluded that for small and medium files EcElgamal is much faster than other algorithms but for large files RSA and EcElgamal takes near about time for encryption and decryption. But due to increase in key length of RSA it is considered as un-useful for future security concerns however EcElgamal and RSA are similar but Elgamal Encryption using Elliptic Curve (EcElgamal) is more advantageous than other homomorphic algorithms because it have less key size and easy implementation than RSA and others hence it provides better results.

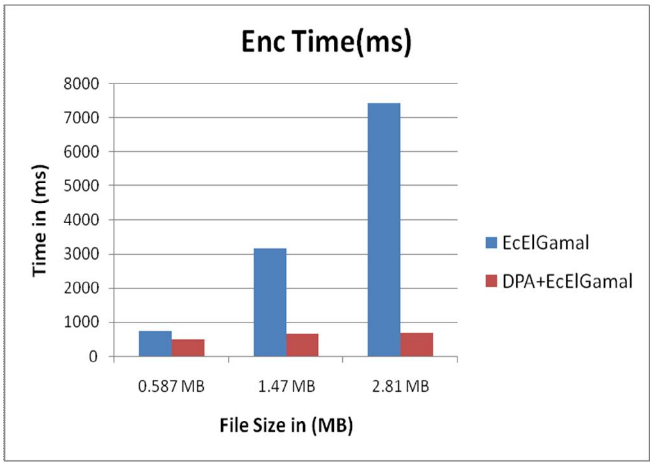

**Figure 3** Total Time for Uploading

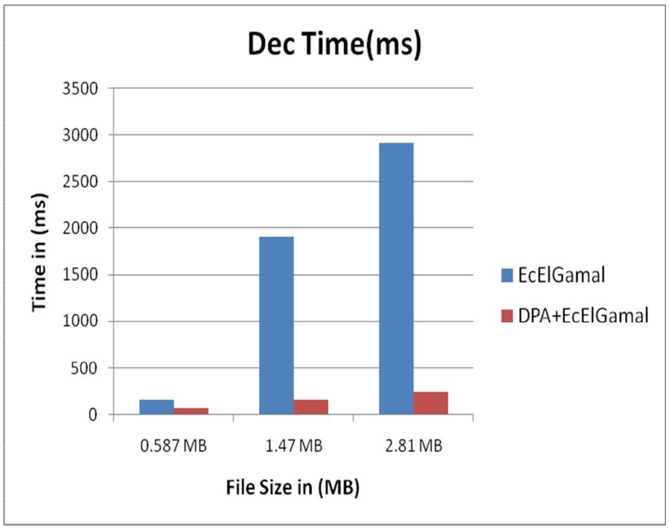

**Figure 4** Total Time for Downloading

Performance Analysis: The Figure 3 and Figure 4 shows uploading and downloading of text content files using simple EcElGamal and Data Partition Algorithm (DPA) based EcElGamal. As it is clear from these graphs that data partition based EcElGamal gives better performance on different file sizes. Uploading of file includes data partition and encryption time for parts in DPA based EcElGamal while in simple EcElGamal includes encryption time for whole file which cause heavy burden on cloud server. Downloading of file includes decryption of each parts and merging of parts in DPA based EcElGamal whereas in simple EclGamal includes decryption of whole file and thus cause extra overhead and consume more time. For small files DPA based EcElgamal gives 40-50 % better performance than simple EcElgamal but as the file size increases the performance of simple EcElgamal degrades more and DPA based EcElgamal algorithm gives approx. 90% better performance for uploading and downloading of files.

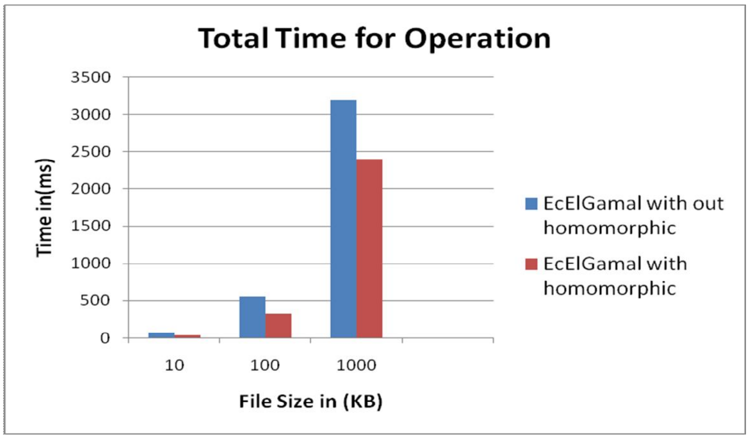

**Figure 5** Total Time for Addition Operation using EcElGamal with Homomorphic v/s EcElGamal without Homomorphic

EcElGamal with homomorphic property is faster and more secure than simple EcElGamal because it apply operation on encrypted data and gives the same result as obtained by applying the operation on plain data hence the cloud server does not aware about the real data contents. The cloud server does not required to share the user's secret key so key transfer time not included whereas in case of simple EcElGamal the decryption of files is required to operate on data and follows dec-add-enc these three aspects so it is not a good choice for data processing and encryption. So finally EcElgamal with homomorphic algorithm is better for both confidentiality and processing speed by providing better processing speed for operation on files than EcElgamal without homomorphic which is clear from above Figure 5 also. In this Figure 5 the results are obtained on numeric data files of same length.

## 6. CONCLUSIONS

In this paper some security issues and research issues in cloud computing have been studied. Performance for input output and Confidentiality are the issues out of several issues studied. We have proposed a solution to improve security and confidentiality of private data with high performance for uploading and downloading files in cloud environment by implementing data partition based EcElGamal algorithm which is homomorphic algorithm. DPA based EcElgamal algorithm gives 80-90 % better performance than simple EcElgamal for uploading and downloading of large text content files 1.47 MB and 2.81 MB. We have also show the use of homomorphic property for giving better performance during any computation on data by applying operation on homogenous files using EcElGamal with homomorphic vs EcElGamal without homomorphic. EcElgamal with homomorphic gives near about 40% better processing speed for computation on data on small files like 10KB files at processing stage but as the files size increase encrypted addition becomes complex in EcElgamal with homomorphic but still it perform better than EcElgamal without homomorphic. Future work can be done on the concern of integrity check of data at processing stage.

## REFERENCES

[1] A.Fox, R. Griffith, A. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, and I. Stoica, "Above The Clouds: A Berkeley View of Cloud Computing.", Dept. Electrical Engineering and Computer Sciences, University of California, Berkeley, Rep. UCB/EECS , 2009

[2] Rabi Prasad Padhay, Manas Ranjan Patra, and Suresh Chandra Satapathy, " Cloud Computing: Security Issues and Research Challenges", International Journal of Computer Science and Information Technology & Security, Vol. 1, No. 2, 136-146, December 2011

[3] Marios D. Dikaiakos, Dimitrios Katsaros, Pankaj Mehra,George Pallis, and Athena Vakali, "Cloud Computing: Distributed Internet Computing for IT and Scientific Research", IEEE Internet Computing Journal, Vol. 13, No. 5, pp. 10-13, September 2009

[4] R. Rivest, A. Shamir, and L. Adleman. "A Method for Obtaining Digital Signatures and Public key Cryptosystems", Communications of the ACM, Vol. 21, No. 2, pp. 120-126, 1978.

[5] P. Paillier, "Public-key Cryptosystems Based on Composite Degree Residuosity Classes", In the Proceedings of International Conference on the Theory and Applications of Cryptographic Techniques ( EUROCRYPT), prague, Vol. 1592, pp. 223–238, Springer 1999.

[6] Taher ElGamal, "A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms", IEEE Transactions on Information Theory, pp. 469-472, 1985

[7] Rosy Sunuwar and Suraj Ketan Samal, " Elgamal Encryption using Elliptic Curve Cryptography", Cryptography and Computer Security, University of Nebraska, Lincoln, December 2015.

[8] Craig Gentry, "Fully Homomorphic Encryption using ideal Lattices", In Proceedings of the 41th Annual ACM Symposium on Theory of Computing, pp. 169–178, 2009.

[9] Deyan Chen and Hong Zhao, "Data Security and Privacy Protection Issues in Cloud Computing", In the Proceedings of International Conference on Computer Science and Electronics Engineering From IEEE Computer Society, Hangzhau, China, pp.647-651, March 2012.

[10] Maha Tebaa, Saïd El Hajji, and Abdellatif El Ghazi "Homomorphic Encryption Applied to the Cloud Computing Security", In the Proceedings of the World Congress on Engineering, London, U.K, Vol.1, July 2012

[11] Simon Fau,Renaud Sirdey, Caroline Fontaine, Carlos Aguilar-Melchor and Guy Gogniat, "Towards Practical Program Execution over Fully Homomorphic Encryption Schemes", In the Proceedings of 8th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing from IEEE Computer Society, Washington, DC, USA, pp.284-290, October 2013.

[12] W. Wang, Y. Hu, L. Chen, X. Huang, and B. Sunar, "Exploring the Feasibility of Fully Homomorphic Encryption" , IEEE Transactions on Computers, Vol. 99, pp. 1, 2013.

[13] FENG Chao and XIN Yang, "Fast Key Generation for Gentry-Style Homomorphic Encryption", The Journal of China Universities of Posts and Telecommunications, Vol. 21, No. 6, pp. 37-44, December 2014.

[14] G.jeeva Rathanam and M.R. Sumalatha, "Dynamic Secure Storage System in Cloud Services", In the Proceedings of IEEE International Conference on Recent Trends in Information Technology, Chennai, India, April 2014

[15] Kamal Kumar Chauhan, Amit K.S Sanger, and Ajai Verma, "Homomorphic Encryption for Data Security in Cloud Computing", In the Proceedings of IEEE 14th International Conference on Information Technology, Bhubaneswar, India, pp.206-209, December 2015.

[16] Ali Azougaghe, Zaid Kartit, Mustapha Hedaboui, Mostafa Belkasmi, and Mohamed El marraki, "An Efficient Algorithm for Data Security in Cloud Storage ", In the Proceedings of 15th International Conference on Intelligent Systems Design and Applications, pp.421-427, 2015.

[17] Babitha.M.P and K.R Remesh Babu, "Secure Cloud Storage Using AES Encryption", In the Proceedings of IEEE International Conference on Automatic Control and Dynamic Optimization Techniques, Pune, India, pp.859-864, 2016

[18] Yasmina Bensitel and Rahal Romadi, "Secure Data Storage in the Cloud with Homomorphic Encryption", In the Proceedings of 2nd International Conference on Cloud Computing and Applications, Marrakesh, Morocco, May 2016.