A Proposed Secure Cloud Environment Based on Homomorphic Encryption

Rasha Falih Hassan[1], Ali Makki Sagheer[2]

Iraqi Commission for Computers and Informatics, Informatics Institute for Postgraduate Studies, Baghdad, Iraq[1,2]

Abstract: Cloud computing means an information technology arrangement where data and software are saved and treated in a reserved data center. In cloud computing, big amount of users 'data are allowed to de collected on cloud server storage for next use, and any computations on stored data will be implemented in the cloud. To keep the stored data that is on the cloud we necessities have to use an encryption system that can do computations on the encrypted data called homomorphic encryption. In this paper, users' data is never kept in the form of plaintext on public cloud and proposed a Secure Cloud Environment Based on Homomorphic Encryption (SCEHE) to develop the security of cloud computing and keeps the client's data by utilizing partial homomorphic encryption ", Elliptic curve cryptography used to generate algorithm's private key then encrypt all user's data by using it, A new algorithm reduces the time of processing, space of storage and make available high security because of its key generated depends on ECC. The use of 64-bit provides enough security to be used in the system.

Keywords: Secure Cloud Environment, Homomorphic Encryption, Elliptic curve Cryptography

I. INTRODUCTION

In the cloud computing the client put his data on the cloud, and any calculations on his stored data will be applied in the cloud, the best way to deal with encrypted data without decrypting is homomorphic encryption [1], the different security issues related to data security, integrity, privacy, confidentiality, and authentication needs to be mentioned [2]. Cryptography is an significant, effective and well-organized component to settle the secure message between the different units by transferring unintelligible data and just the legal recipient can be able to access the data, The right choice of cryptographic algorithm is essential for secure communication that offers more efficiency, security and accuracy [3]. The strength of encryption rest on the strength of the keys used in the encryption and decryption, , if the key is short or weak this leading to produce weak encryption and vice versa, In this paper, key generation depends on the elliptic curve, so the key strength depends on the ECDLP [4], HE can be either, symmetric or asymmetric. An encryption contains three algorithms: Kegan, Encrypt and Decrypt [5], The right choice of cryptographic algorithm is essential for secure communication that offers more efficiency, security and accuracy [3], Elliptic curve cryptography along with its own specifications like homomorphism for encryption and decryption application will give us less computational complexity with same level of security similar to other algorithms. Elliptic curves if simulated with ElGamal, Paillier and RSA gives enhanced security with smaller key generation so that even small devices like mobile phones, pager etc. [6] .

II. RELATED WORK

In 2016, V.Biksham, Cloud Service Providers (CSP) give privacy and security to the cloud users through cryptographic encryption algorithms. With inquiry , user can entrée the data from the cloud servers over decryption. But frequent decryption of cipher text may lead to exploit the integrity and authentication. In this paper a high security encrypted data is suggested by using "somewhat" and "fully homomorphic" encryption approaches. To supply security to the encrypted data with computations, a secure encryption method called homomorphic encryption which provides calculations on encrypted data without decrypt the cipher text and enhance the performance of the cloud services [7].

In 2016, Ming-quan Hong, Wen-bo Zhao, Aimed at SMC difficult (computation and communication cost), proposed Elliptic Curve Cryptography (ECC) based homomorphic encryption scheme that is dramatically reduces & treats a problem. It displays that the system has advantages in energy ingesting, communication consumption and privacy protection through the comparison experiment between ECC based homomorphic encryption and RSA & Paillier encryption algorithm. Further evidence, the scheme of homomorphic encryption scheme based on ECC is applied to the calculation of GPS data of the earthquake and proves it is proved that the scheme is feasible, excellent encryption effect and high security [8].

DOI 10.17148/IARJSET.2019.6524

In 2016, Mr. Manish M Potey, The cloud service provider stocks the plaintext format of data and user requests to use their own encryption algorithm to secure their data if required. The data requirements to be decrypted when it is to be treated .This paper stresses on loading data on the cloud in the encrypted format using fully homomorphic encryption. The data is kept in Dynamo DB of Amazon Web Service (AWS) public cloud. User's calculation is performed on encrypted data in public cloud storage. When deductions are required they can be copied on client machine. The many security issues associated with data security such as privacy, confidentiality, integrity and authentication desires to be mentioned. [9].

In 2016, Yannan Li, Yong Yu, Bo Yang, Geyong Min, Huai Wu al, Confirm privacy of data in storing and transferring or in use by progressions, searching an encrypted data in cloud, with tools similar to traditional cryptography is a serious problem, In this paper, they suggest a multi cloud architecture of N distributed servers to repartition the data and to permit achieving an FHE. A homomorphism makes secure entrustment of computation to a third party potential. Many conventional encryption structures hold either multiplicative or additive homomorphic property and are now in use for particular application, Homomorphism is a property by which a problem in one algebraic system can be renewed to a problem in alternative algebraic system, be solved and the result later can also be decoded back effectively [10].

In 2017, Xidan Song, Yulin Wang, Fully homomorphic encryption method has disadvantages of large key size and low calculation efficiency, and it is not practical for the secure cloud computing. In this paper, developed a hybrid cloud computing scheme based on the Paillier algorithm which is additively homomorphic, and RSA encryption algorithm which is multiplicative homomorphic. Customer's calculation requests can be described as the combination of simple add and multiplicative operation and the operands. An Encryption Decryption Machine which running in the private cloud processes the encryption according to the type of the operation and upload the cipher texts to the public cloud. The public cloud process calculation without knowing the exact data. Then we run simulations and analyze the results, and the results show that the scheme is practical and efficient [11].

In 2017, Xidan Song, Yulin Wang Homomorphic encryption tools can settle a dispute of data privacy security cloud environment, but there are many difficulties in access the data which is encrypted by a homomorphic Encryption algorithm which kept cloud. In this paper, on the principle of attribute encryption, we suggest a fully homomorphic encrypt scheme which founded on attribute encryption with LSSS matrix. Structure of scheme supports fine-grained cum supple access control along with "Query-Response" instrument to support users to efficiently recover looked-for data from cloud server storages. The structure of scheme should support flexibility to revoke system rights from users without modernizing the key client; it decreases the pressure of the client greatly. Finally, security analysis illustrates that the arrangement can resist collusion attack. An evaluation of the performance from existing CP-ABE scheme indicates that our scheme eases the computation cost significantly for users [6].

In 2019, Adi Akavia, Dan Feldman, Hayim Shaul, Secure report is hypothetically potential with Fully Homomorphic Encryption (FHE), In this paper, executed main reporting system in an open source library and can response such database queries when treating only FHE encrypted data and queries. The experimental results show that Implemented Secure report queries on billions records in minutes on an Amazon EC2 server, compared to less than a hundred-thousand in previous FHE based solutions [12].

In 2019, Ahmed El-yahyaoui , Mohamed Dafir Ech-cherif El kettani, fully homomorphic encryption (FHE) is a intelligent type of encryption schemes that assists working encrypted form of data. It offers efficient techniques for outsourcing calculations over encrypted data to a distant. The resultant scheme is named Verifiable Fully Homomorphic Encryption (VFHE). Presently, it has been demonstrated by many existing schemes that the theory is feasible but the efficiency needs to be dramatically value-added in order to make it practical for real applications. One subtle difficulty is how to efficiently handle the noise. In this paper, present a well-organized and symmetric provable FHE based on a modern mathematic construction that is without noise, the noise is persistent and does not rest on homomorphic evaluation of ciphertexts. The homomorphy of our structure is gained from simple matrix operation processes (addition and multiplication). The running time of the multiplication process of our encryption scheme in a cloud environs has an order of a small number of milliseconds [13].

III. CRYPTOSYSTEM

3.1 Cryptography deals with securing communication channels sensitive data, electronic transactions and other critical information [14], It is concerned with designing a cryptosystem or a cryptographic system that is accomplished of providing services likes authenticity, confidentiality, integrity and nonrepudiation [3]. By confidentiality we mean that a cryptosystem is intended to allow access to information or sensitive data only to authorized users. Authenticity refers to the ability of a cryptosystem to validate the source of data origin. Integrity refers to the assurance of a cryptosystem that a message was not altered in transit intentionally or unintentionally through, modification insertions, and deletion, Non-repudiation refers to the capability of the cryptosystem to provide evidence in instance a dispute arises by a sender appealing that he/she didn't send the data. There are two kinds of cryptography, namely symmetric & asymmetric [14].

DOI 10.17148/IARJSET.2019.6524

3.1 Cryptosystem Types

a) Symmetric Cryptosystem: Symmetric key cryptography it is also known as private key cryptography, is kind of encryption tool in which sender as well as receiver, entities use to share the same key. Is employed similar key for message encryption and decryption, the strength of the symmetric key encryption is conditional by the secrecy of encryption and decryption keys structure that has been choosing as cryptographic standards[3]. Private Key cryptosystems uses the same key for encryption and decryption of the information, so a particular message or collection of several messages may possess different keys. There is constraint of symmetric ciphers is that, the key management process becomes essential to use them securely. Key management process consists of construction, distribution and refreshing of the secret keys, consisted in the communication [15].

b) Asymmetric Cryptosystem: Asymmetric key cryptography is called as Public key cryptography. In this, sender encrypts the message via public key of receiver and receiver decrypts the information using his private secret key. Public key cryptography technique can be utilized for implementing the various Digital signature ways. RSA and DSA are two most noticeable digital sign Mechanisms. Public key techniques are generally based on the very large computational complexity of "hard problems”. The hardness of well-known RSA algorithm is based on hard problem of integer factorization, while hardness of DiffieHellman and DSA algorithms is based on discrete logarithm problem. Recent times, elliptic curve cryptography (ECC) have been developed, in which security is completely based on the number theoretic computationally hard problems involving elliptic curves. Paillier cryptosystem, invented by Pascal Paillier in 1999, is also an example of probabilistic behavioral asymmetric tool for public key cryptography [15].

IV. HOMOMORPHIC ENCRYPTION

In 1978 Ronald Rivest, Leonard Adleman and Michael Dertouzos submitted at the beginning the idea of Homomorphic encryption. Since then, a few evolutions have been prepared for 30 years. The encryption scheme of Shafi Goldwasser and Silvio Micali was suggested in 1982 was a provable security encryption method which reached a notable level of security, it was an additive Homomorphic encryption, however, it can encrypt just a single bit. In the same concept in 1999 Pascal Paillier was also suggested a provable security encryption system that was also an additive Homomorphic encryption. After few years later, in 2005, Dan Boneh, Eu-Jin Goh and Kobi Nissim designed a system of verifiable security encryption, by which we can do an unlimited number of additions with only one multiplication [16].

4.1 Idea of Homomorphic Encryption

HE is new topics of security which make more researchers contract with , because of providing more security for data especially in the cloud computing environment [17], Homomorphic Encryption systems permit to perform operations on encrypted data without knowing the private key (without decryption), the client is the only owner of the secret key. When we decrypt the result of any operation, it is equivalent if the calculation carried out on the raw data [18], HE techniques are partial, somewhat and fully homomorphic encryption with the purpose of a secure store, transfer and dealing with ciphertext in a means that maintains the integrity and confidentiality of data [19].

4.2 Homomorphic Encryption Example:-

HE concept is shown in the following Figure 2.5. When the user wants to add two numbers for instance 5 and 10 the product is 15, the two numbers are encrypted through multiplied with 5, then the sum of the encrypted number is 75 as a result that is stored on the cloud server, the user download data from cloud and recovered the original text [20].

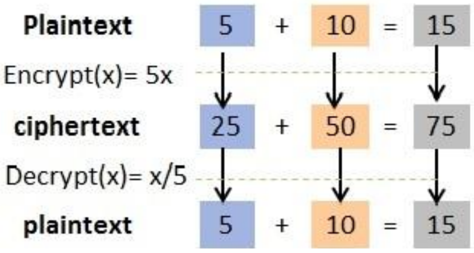


Figure 1: Homomorphic Encryption Example [20].

4.3 Functions of Homomorphic Encryption:

Homomorphic Encryption has four function of as in figure (2.4):

1. Function of KeyGen: It is an algorithm that takes parameter of security (SP) to generate secret key (sk) and public key (pk), (pk, sk) KeyGen (SP).

2. Function of Encryption (Enc): It is algorithm, it uses plaintext and (sk) to produce a ciphertext (c), c = Enc (sk, m).
3. Function of Evaluation (Eval): It is algorithm, the server uses function f designed for evaluating the ciphertext, and it's done with using function f and pk (Eval(f, pk, c)), where c= (c1, ... , ct) and t means the number of circuit inputs. Hence, Dec (sk, Eval (f, pk, c)) = C (m1, m2 …… mt), Where C is a computation executes in the plaintext.
4. Function of Decryption (Dec): It is algorithm produces a plaintext (m) that comes from ciphertext and sk m = Dec (c, sk), Hence, after evaluation, original text obtained as follows Dec(sk, Eval(f, pk, c)) as in Figure (3) [21]. Assume that (m1), (m2) ∈ M and (c1) and (c2) ∈ C then m1 = Dec (c1) and m2 = Dec (c2) and this lead to: Dec (c1*c2) = m1*m2, When multiplication operation group applied in C and M, consecutively [9].

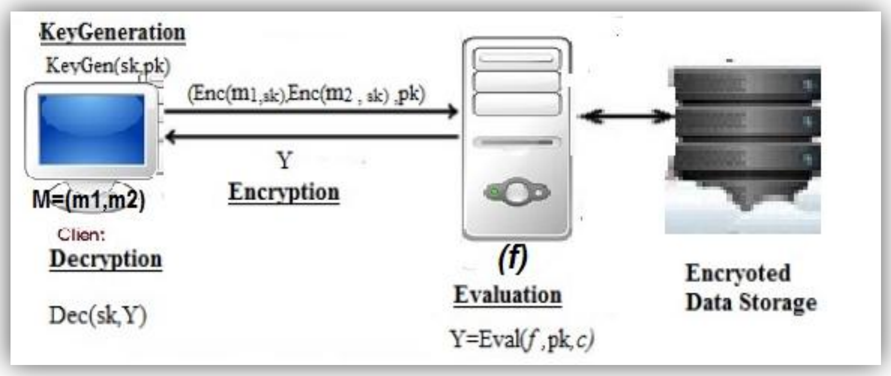


Figure 3: Homomorphic Encryption functions [4]

4.4 Homomorphic Encryption Features:

Assume that:
(m1, m2) ∈ M, (c1 and c2) ∈ C then
c1 = Enc (m1)
c2 = Enc (m2)
P is the prim number.

• Additive Homomorphic Encryption
Enc (m1 + m2) mod p = c1 + c2 mod p.

• Multiplicative Homomorphic Encryption
Enc (m1 * m2) mod p = c1*c2 mod p [1].

4.5 Representation Example of Homomorphic Encryption in cloud computing:

A Homomorphic Encryption (HE) example for a sample cloud application is clarified in Figure (2). In this scenario:

1. (Step 1) the client C first encrypts his private data.
2. (Step 2) sends the data that has been encrypted to the cloud servers S.
3. (Step 3) when the client wants to implement a function, f ( ) over his own data, he transmits the function to a server.
4. (Step 4) the server executes a homomorphic operation on the encrypted data using the Eval function, i.e. computes f ( ) blindfolded
5. (Step 5) sends the encrypted result to the client.
6. (Step 6) the client recovers the data with his own secret key and obtains f (m).

In this example explaining the homomorphic operation Evaluation (Eval), there is no need for the private key of a client at the server side and permits the operations for instance multiplication and addition on the client's encrypted data [5]

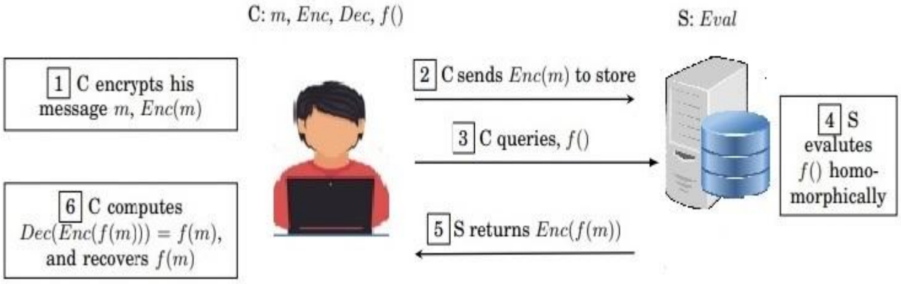

C: m, Enc, Dec, f()
S: Eval
1 C encrypts his message m, Enc(m)
2 C sends Enc(m) to store
3 C queries, f()
4 S evalutes f() homo-morphically
5 S returns Enc(f(m))
6 C computes Dec(Enc(f(m))) = f(m), and recovers f(m)

Figure 2: client server HE functions scenario, where (c) is client and (s) is server [1].

DOI 10.17148/IARJSET.2019.6524

V. ELLIPTIC CURVES CRYPTOGRAPHY

Elliptic Curve (EC) has been introduced at the beginning in cryptography by Miller1 and Koblitz2, depends on the hardness of the Elliptic Curve Discrete Logarithm Problem (ECDLP) [22] ,like homomorphism for encryption and decryption implementation along with their own specifications will give us less computational complexity with similar level of security [6].

Elliptic Curve Cryptosystem (ECC) is a technique of public key cryptography, is based on the structure of algebraic and discrete logarithms of an elliptic curve over finite fields. Definitions of elliptic curve (EC) based on two kinds of finite field:

1-prime field Fp, where p is a large prime number
2-Binary fields [23, 24].

The key sizes of ECC are smaller, faster encryption, more efficient and better security for the same level of security compared with other systems of public cryptography such as RSA [24].

As a result of these advantages of elliptic curve, a number of studies have been presented by many researchers. For example, (Williams Stallings in 2011 presented study about ECC in his book5, Hongqiang in 2013 proposed an approach to create a random number k and speed up calculating the scalar multiplication in the encryption and decryption methods, An execution of ElGamal ECC for encryption and decryption a message is also suggested by Debabrat Boruah in 2014, Meltem Kurt and Tarik Yerlikaya in 2013 presented a modified cryptosystem by using hexadecimal to encrypt data. [25].

Elliptic Curve Cryptography (ECC Compared to currently prevalent cryptosystems such as RSA, ECC offers the same security with smaller key sizes. By the help of table which provides approximate Comparable with key sizes for symmetric and asymmetric-key cryptosystems depended on the famous algorithms for attacking them [6].

Table1: Comparable Key Sizes (in bits) [6]

Symmetric	ECC	DH/DSA/RSA
80	163	1024
112	233	2048
128	283	3072
192	409	7680
256	571	15360

5.1 Elliptic Curve Discrete Logarithm Problem (ECDLP)

One of the very interesting open problems in cryptography is the understanding of a trapdoor on discrete logarithm, in order to solve the DLP is hard only if declared parameters are used, while it is easy by using a private key (trapdoor key) [23,16].

Definition (DLP): For specific group G, let x, y ∈ G, recall that in the DLP, is to find an integer a ∈ Z so that $x^a = y$. The DLP can be used in many finite groups in addition to the multiplicative group over a prime filed Fp, this knowledge can be increased to random groups and, particularly, for elliptic curve groups [26,27].

Definition (ECDLP): The Elliptic Curve EC, let P, Q ∈ EC, recall that in the ECDLP, is to find an integer k ∈ Z where ($1 \leq k < n$), such that KP= Q. It's easy to determine the points Q and P, but the difficulty deceits in finding integer k from multiple points of P*k, even if the knowledge of Q, P of EC. The ECDLP over Fq is more fixed than the DLP in Fq, This specify makes the cryptographic technique built on the ECDLP much more secure than that built on the DLP [27,24].

5.2 Definition of Elliptic Curve over Prime Field:

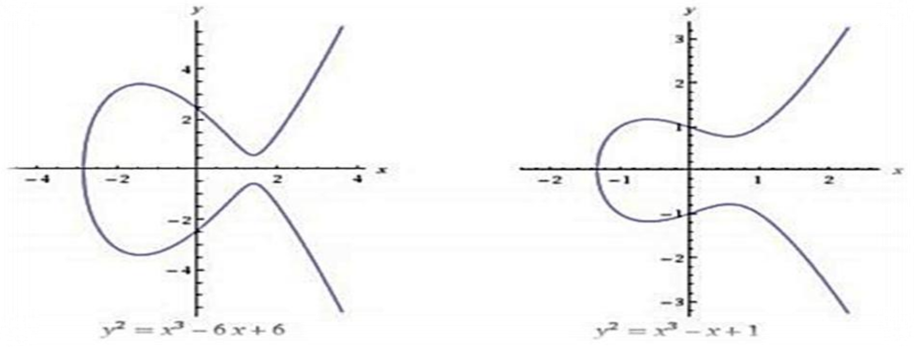


Figure4: Generated Elliptic Curve Points [23]

Let q refer to the prime number, an elliptic curve EC over a prime field Fq is given in the following equation 1:

$$\text{EC: } y^{2}\ (\text{mod } q)^{=}\ x^{3} + ax + b\ (\text{mod } q) \qquad \ldots\ldots.. \qquad (1)$$

Where a, b ∈ Fq and must satisfy the equation that: $4a^3 + 27b^2 \neq 0$ (mod p), so the group of elliptic curve points E (Fq) are generated when all points of (x, y) satisfy the equation 1 of elliptic curves with a point ∞ (called the point at infinity) [30]. Figure (4) represent example of the generated elliptic curve points:

5.3 Arithmetic Structure of Elliptic Curve

An equation with two variables and coefficients represented elliptic curve. For cryptography, the variables and coefficients are limited to components in a finite field, which effects in the meaning of a finite abelian group and are real numbers in which, at elliptic curves.

A. Adding and doubling point over Elliptic Curve

Addition Point

Let point P1 = (x1, y1), P2 = (x2, y2), where x1 ≠ x2 means that P1 ≠ P2. P1, P2 belong to EC (Fp) defined in Equation (5)

The summation of P1 and P2 creates another point P3= (x3, y3) which is also belong to EC (Fp). Add two points on the elliptic curve shown in the following [25]:

If P1≠P2 with x1=x2 and y1≠y2 then P1+P2 = O

If P1≠P2 with x1≠x2 then

P1 + P2 = P3= (x3, y3) (2)

Where

$$\lambda = \frac{(y_1 - y_1)}{(x_2 - x_1)} \qquad (3)$$

x3= (λ2 – x1 –x2) mod p (4)

y3=(λ (x1 – x3) –y1) mod p (5)

x3= (λ2 – 2x1) mod p (8)

y3= λ (x1 – x3) –y1 mod p (9)

- **Doubling Point**

If $P_3 = P_1 + P_1 = 2\ P_1$ and $P_3 = (x_3, y_3)$ (6)

Where

$$\lambda = \frac{(3(2\,x_1)+a)}{(2y_1)} \qquad (7)$$

$x_3 = (\lambda^2 - 2x_1)$ mod p (8)

$y_3 = \lambda\ (x_1 - x_3) - y_1$ mod p (9)

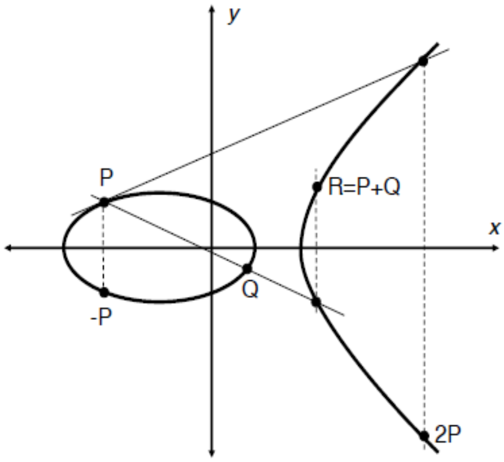


Figure 5: the Addition and doubling point of EC [4]

DOI 10.17148/IARJSET.2019.6524

VI. THE PROPOSED SECURE CLOUD ENVIRONMENT

6. 1 General Model of proposed Secure Cloud Environment

A Proposed Secure Cloud Environment allows users to access the contents in a protected manner, and keep user Data from any crack or unauthorized entree, so that, Data still protected, no one can access it even service providers, which are considered sensitive information to conserve privacy, So that, System can be accessed only the user who has the secret key. The encryption is done by submitting user's data and sending them via a secure channel (SSL) to the server to encrypt all user data and stored in server storage, which are accessed during login phase by the user without decryption using the homomorphic encryption to be protected user's privacy**.**

6.2 Illustrated Steps of General Model as bellow:-

1. The client enters his Mail (e-mail) and password (pass) in the process of logging into the system.
2. The client sends over the network (over the SSL connection). Client's mail and password (pass) to the server.
3. The key of the proposed algorithm generated depending on the combination of Mail (e-mail) and password (pass) which putting it to SHA-256 to get (d) which is used as big integer which multiplied by a base point (G) by using (Elliptic curve (ECC) to get a secret key which used in encryption.
4. The secret key which is generated from previous step used to encrypt the data of user and kept in a storage server.
5. At any time ,the user want to log in another time to request for a stored data by using his e-mail and password again, the server receives client's mail and password (pass), uses them to generate secret key based on PHE by given it to the homomorphic encryption functions and obtain the secret key and output, and hence search for the encrypted data in the local database and given the encrypted results to the homomorphic encryption function to obtain the output with using evaluation function, if the output is matched, the user's data is found then decrypted to the user, Figure 5 shows general model of a system.

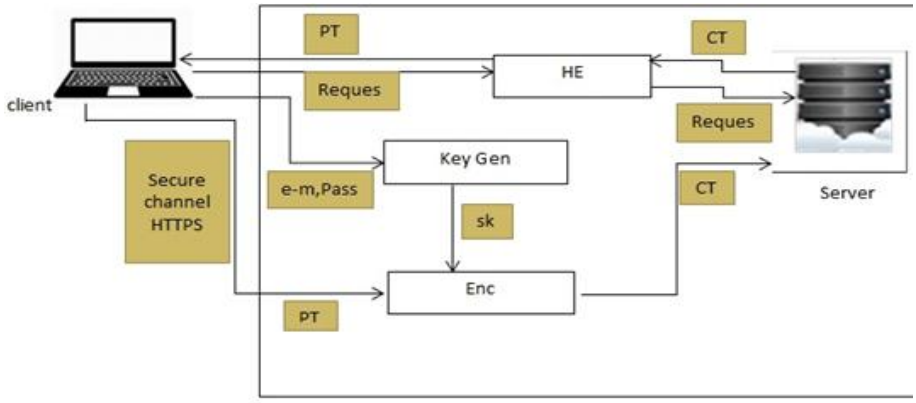
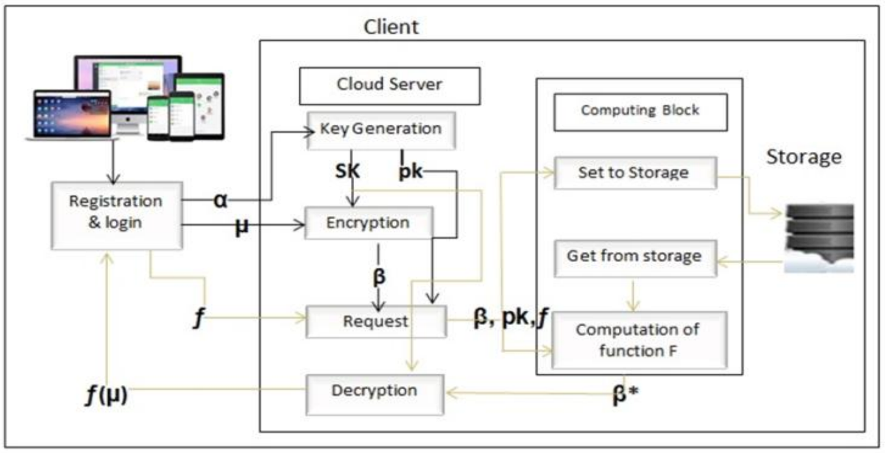


Figure 5: General Model of a system

6.3 Private Model of secure cloud Environment



$\mu=(\mu_1, \mu_2, \mu_3 \ldots \mu_t), \beta=(\beta_1, \beta_2, \beta_3, \ldots \beta_t), \beta^*=(\beta^*_1, \beta^*_2, \beta^*_3, \ldots \beta^*t)$

Figure 6: Structure of the proposed private cloud model

DOI 10.17148/IARJSET.2019.6524

The proposed Cloud Environment model is a secure private cloud which will be used by the client to upload his data and store the encrypted data on the cloud. At the first time, the client logs in web site and if he be successful, the client will upload his file and then encrypt it by using a private key that will be generated. When the client wants to decrypts this file, this key also will be used. Database of the cloud in Homomorphic Encryption format will be used to save an encrypted file. Figure (6) shows the Structure of the proposed private cloud model where all the different components and their communication with each other are shown here bellow. All the processes bellow depend on this architecture.

Where α is a user's login information (e-mail and password) or what is called with a security parameter, µ is an uploaded file, sk is a secret key, pk is a public key, β is an encrypted file, f is the function, β* is a result of computation of function f that applied on the encrypted file (β) and f (µ) is a result of computation of function f that applied on the uploaded file (µ).

6.4 Security model

The security of the secure cloud environment depends on Cryptography of Elliptic Curve, which used Elliptic-Curve Discrete Logarithm Problem (ECDLP) for key generation.

Security Model depends on the proposed algorithm

1. Key Generation:

- d= SHA-256 (e-mail + Password)
- k = d (G) where G is the base point, so G= (x, y)
- k = (k_1, k_2)
- We depend k_1 as the secret key, sk=k_1.

2. Encryption

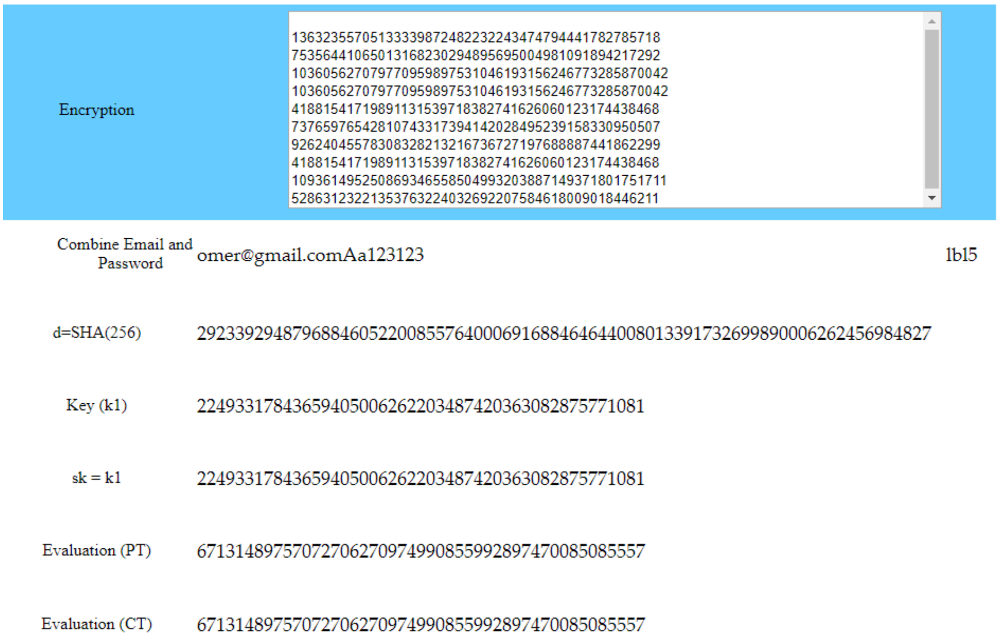
1. Convert data chars (M) and SHA-256 (password) to ASCII where M= (m_1, m_2… m_n).
2. c_i = m_i* k_1.
3. C is the output of all encrypted attributes

3. Decryption

1. mi = c_i * $k1^{-1}$
2. M is the output of all decrypted data

6.5 Example of Secure Cloud Environment system

If User e-mail= omer@gmail.com and password = Aa123123, user's data for instant is (Nowadays, there are many universities around the world and each of them may have up to 10 thousand students. To handle this large number of students may cause a problem especially in terms of the student attendance. Attendance is one of the important factors that affect the students' performance in class the attendance in the majority of) as show in Figure.



Encryption

13632355705133339872482232243474794441782785718
75356441065013168230294895695004981091894217292
10360562707977095989753104619315624677328587004 2
10360562707977095989753104619315624677328587004 2
41881541719891131539718382741626060123174438468
73765976542810743317394142028495239158330950507
92624045578308328213216736727197688887441862299
41881541719891131539718382741626060123174438468
109361495250869346558504993203887149371801751711
52863123221353763224032692207584618009018446211

Combine Email and Password omer@gmail.comAa123123 1b15

d=SHA(256) 29233929487968846052200855764000691688464644008013391732699890006262456984827

Key (k1) 224933178436594050062622034874203630828757 71081

sk = k1 224933178436594050062622034874203630828757 71081

Evaluation (PT) 671314897570727062709749908559928974700850 85557

Evaluation (CT) 671314897570727062709749908559928974700850 85557

Figure 10: Secure Cloud Based on Homomorphic Encryption (SCPHE).

Step1: log in Secure Cloud Environment by using E-mail and password.
Step2: generate a secret key
SK = d*G (gx, gy) = (k_1 ,k_2)
Set k_1 as a secret key based on Elliptic curve used for encryption.
Step3: Encryption of data Based on partial Homomorphic Encryption (PHE)
Step5: Matching of evaluation of stored encrypted data and user's request encrypted data.
Step 6: if the matching is true, users' data is found then decrypted for user.

6.6 phases of secure cloud environment Implementation steps:

There are 6 phases (Setup, KeyGen, Encryption, Decryption, Evaluation and Matching) Figure 3.4shows these phases.

a) Setup (): It has an EC security parameter (SP) as an input, where it was based on the standard NIST values (p, a, b, G, n) and assigned to this function for use in generating the secret key in the Key Gen.
b) KeyGen (SP, S): It is accepted EC parameter domain SP, and some of the data (S), where S or (e-mail || pass), the outputs of this algorithm is the secret keys sk which associated with some Data S.
c) Encryption (sk, p, PT): This algorithm is used to encrypt all users Data (PT), so the input is PT, sk and p, the output is ciphertext C.
d) Decryption (sk, p, C): This algorithm is used to decrypt all encrypted Data. Input data is a ciphertext C, sk and p, the output is getting the original data (PT) as in Figure 3.4.
e) Plaintext Evaluation (e-mail||SHA-256(pass), pt, Mult-sk, Mult-pt, ouput): get result1 after applying computation on pt.
f) Ciphertext Evaluation (e-mailct||Pct, mulct, ouput): get a result2 after applying computation on ct.
G) Matching (e-mail, pass, e-mailct, Pct., true or false): Matching the resilt1 and result2 to be either true or false.

6.7 Security of A proposed Secure Cloud Environment

1-The use of homomorphic encryption (HE) in the secure cloud Environment provide great protection for user data that became completely encrypted and no one could know the information even if the server database was hacked by Hacker.
2- The security of the ECC algorithm depends on the difficulty of ECDLP. ECC currently appears to be implemented on a 64-bit to provide nearly the same security level against the attacks of hackers compared with algorithms like (RSA Elgamal, paillier). This variation in the length of the keys has led to improve and speed in performance and less storage requirements, table1present the comparison between proposed algorithm and RSA in terms of execution time.

Table 3: Comparison between A proposed algorithm and (RSA, paillier, Elgamal) in execution time

Process in (ms)	Proposed algorithm	RSA	Elgamal	paillier
Encryption	35	228	79	983
Decryption	7	258	6723	726
Evaluation	9	258	23	829

VII. CONCLUSION

1-power of cloud can be exploited if user is able to carry out computation on encrypted data. Homomorphic Encryption technique allows computing with encrypted data ,thus Homomorphic encryption is a true option in the process of Information security, since they could not obtain identified information except encryption, where it was implemented successfully and gave results and strong security to maintain the privacy of users in less time and power of security.

2-The use of ECC means the powerful and efficient asymmetric algorithms for the given key length, and it is good, particularly for security applications where it restricted in power calculation and integrated circuit area, e.g. PC cards, wireless devices and smart cards.

REFERENCES

[1]. Sarah Shihab Hamad, Ali Makki Sagheer "Public Key Fully Homomorphic Encryption", Journal of Theoretical and Applied Information Technology, Vol.96, No 7, 2018.
[2]. Manish M. Potey, C.A. Dhote, Deepak H. Sharma, "Homomorphic Encryption for Security of Cloud Data ", 7th International Conference on Communication, Computing and Virtualization, Elsevier, 2016.
[3]. Muhammad Faheem Mushtaq, Sapiee Jamel, Abdulkadir Hassan Disina, Zahraddeen A. Pindar, Nur Shafinaz Ahmad Shakir, Mustafa Mat Deris, " Survey on the Cryptographic Encryption Algorithms", (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 8, No. 11, 2017.

[4]. Marwan Majeed Nayyef, Ali Makki Sagheer, Sara Shhab Hamad " Attribute Based Authentication System using Homomorphic Encryption", Journal of Engineering and Applied Sciences, 2018.

[5]. Kalpana Gudikandula, P. Kumar,Ravilla VenkataKrish, "Homomorphic Encryption Environment-Service Provider based Encryption and Decryption Endpoints for Third-party Cloud Provider", Journal of Computer Science IJCSIS, Pennsylvania, USA Vol. 15No.7, 2017.

[6]. Yong Ding, Xiumin Li," Policy Based on Homomorphic Encryption and Retrieval Scheme in Cloud Computing", International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC), IEEE, 2017.

[7]. D.Chandravathi and Dr. P.V.Lakshmi," A New Hybrid Homomorphic Encryption Scheme for Cloud Data Security", Advances in Computational Sciences and Technology, Volume 10, Number 5 pp. 825-837, 2017.

[8]. Ming-QuanHong , Peng-YuWang ,Wen-BoZhao,"Homomorphic Encryption Scheme Based on Elliptic Curve Cryptography for Privacy Protection of Cloud Computing", 2nd International Conference on Big Data Security on Cloud (Big Data Security), IEEE International Conference on High Performance and Smart Computing (HPSC), and IEEE International Conference on Intelligent Data and Security (IDS), IEEE, 2016.

[9]. Kamal Benzekki , Abdeslam El Fergougui, Abdelbaki El Belrhiti El Alaoui," A Secure Cloud Computing Architecture Using Homomorphic Encryption", (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 7, No. 2, 2016.

[10]. R.Manjua, A.Shajin Nargunam and A. Rajendran,"Multimodal Biometric Authentication system Based Performance Scrutiny", Medwell Journal 2014.

[11]. Xidan Song ,Yulin Wang,," Homomorphic cloud computing scheme based on hybrid homomorphic encryption", 3rd International Conference on Computer and Communications (ICCC) ,IEEE, 2017.

[12]. Adi Akavia, Dan Feldman, Hayim Shaul," Secure Data Retrieval on the Cloud: Homomorphic Encryption meets Coresets ", IACR Transactions on Cryptographic Hardware and Embedded Systems, 80-106, 2019.

[13]. Ahmed El-yahyaoui and Mohamed Dafir Ech-Cherif EL kettani," A Verifiable Fully Homomorphic Encryption Scheme for Cloud Computing Security", Technologies, 7, 21, 2019.

[14]. Shishay Welay Gebregiyorgis "Algorithms for the Elliptic Curve Discrete Logarithm and the Approximate Common Divisor Problem", Fulfillment of the Requirements for the Degree of Doctor of Philosophy in Mathematics the University of Auckland, 2016.

[15]. Rivest, Ronald L., Len Adleman, and Michael L. Dertouzos. "On data banks and privacy homomorphisms." Foundations of secure computation 4.11 (1978): 169-180.

[16]. Maha Tebaa , Saïd El hajji, Abdellatif EL GHAZI, "Homomorphic Encryption Applied to the Cloud Computing Security", Proceedings of the World Congress on Engineering , U.K, London, 2012.

[17]. Yunchuan Sun, Junsheng Zhang, Yongping Xiong, "Data Security and Privacy in Cloud Computing ", International Journal of Distributed Sensor Networks, 2014.

[18]. Maha Tebaa, Said El hajji, "Secure Cloud Computing through Homomorphic Encryption ", International Journal of Advancements in Computing Technology (IJACT), Volume5, 2013.

[19]. Prof.S.D.Pingle, "Survey of Latest Trends in Cryptography and Elliptic Curve Cryptography", International Journal of Scientific Research and Education, Volume 4, Issue 05, 2016.

[20]. Yatao Yang , Shuang Zhang , Junming Yang, Jia Li, and Zichen Li, "Targeted fully homomorphic encryption based on a double decryption algorithm for polynomials", Tsinghua Science and Technology, Vol. 19, No. 5, pp. 478-485, 2014.

[21]. Yatao Yang, Shuang Zhang , Junming Yang, Jia Li, and Zichen Li, "Targeted fully homomorphic encryption based on a double decryption algorithm for polynomials", Tsinghua Science and Technology, Vol. 19, No. 5, pp. 478-485, 2014.

[22]. V. Biksham, D. Vasumathi, PhD" Homomorphic Encryption Techniques for securing Data in Cloud Computing: A Survey", International Journal of Computer Applications (0975 - 8887) Volume 160 - No.6, 2017.

[23]. Sarita Kumari" A research Paper on Cryptography Encryption and Compression Techniques ", International Journal Of Engineering And Computer Science ,Volume 6, Page No. 20915-20919 , 2017 .

[24]. Sunuwar, Rosy & Suraj Ketan Samal. "Elgamal Encryption using Elliptic Curve Cryptography", Cryptography & Comp Security, 2015.

[25]. Dawahdeh, Ziad E., Shahrul N. Yaakob and Ali Makki Sagheer, "Modified ElGamal elliptic curve cryptosystem using hexadecimal representation", Indian Journal of Science and Technology, Vol. 8, No. 15, 2015.

[26]. Ali Makki Sagheer, "Elliptic curves cryptographic techniques, International Conference Signal Processing and Communication Systems (ICSPCS), IEEE, 2012.

[27]. Patel, Sankita J., Ankit Chouhan and Devesh C. Jinwala, "Comparative evaluation of elliptic curve cryptography based homomorphic encryption schemes for a novel secure multiparty computation", Journal of Information Security, Vol. 5, No. 1, 2014.

[28]. Dave Thompson" ELLIPTIC CURVE CRYPTOGRAPHY", partial fulfillment of the requirements for Departmental Honors in the Department of Mathematics Texas Christian University Fort Worth, Texas , 2016.

[29]. Shazia Tabassam" Security and Privacy Issues in Cloud Computing Environment ", Journal of Information Technology & Software Engineering,, Vol.96, No 7,2017.

[30]. Samta Gajbhiye, Sanjeev Karmakar, Monisha Sharma" Study of Finite Field over Elliptic Curve: Arithmetic Means", International Journal of Computer Applications, Volume 47, No.17, 2012.

[31]. Samta Gajbhiye, Sanjeev Karmakar, Monisha Sharma" Study of Finite Field over Elliptic Curve: Arithmetic Means", International Journal of Computer Applications, Volume 47, No.17, 2012.

DOI 10.17148/IARJSET.2019.6524