LIGHT WEIGHT SECURITY SCHEME OF CLOUD STORAGE SYSTEM BASED ON ECC (CURVE25519)

Manoj Kumar, Surayya Farhat*, Mohd Shadab Alam
E-Mail Id: sdmkg1@gmail.com, suryyafarhat@gmail.com, shadab.malikcivil@gmail.com
Department of Mathematics and statistics, gurukul kangri vishwavidyalaya, Haridwar (UK), India

Abstract- This paper focuses on the investigation of security of data in cloud computing environment. Intermittent nature of encryption and description scheme into the cloud computing becomes a challenging task and it also affects the security of the cloud data. The proposed model generates and step-up the secure data but also transforms the generated cloud secure data. Furthermore, to increase the security of data and stability of the overall cloud computing environment the cryptographic algorithm is most important for secure file. The cryptography is referring to share the file into cloud, secure manner, private, and integrity in efficient manner as this is related to hacker hack the file from cloud environment. Secure file or information in cloud computing environment in the efficient manner and taking care of all the problems at the time of file sharing is the most important for cryptographic algorithms. The researchers have researched on various algorithms for overcoming the security issues generating problem during the file sharing or uploading and downloading phase. We proposed lightweight scheme based elliptic curve cryptography on the Curve25519. This curve-based solution will be faster, secure and light-weight for storing the info into cloud storage. The suggested model is tested on CloudSim / Cloud Report environment and the results show its superiority over other existing method.

Keyword: Elliptic Curve Cryptography, Grid-computing, Cloud Computing, encryption, Curve25519, End-to-End Encryption.

1. INTRODUCTION

Lightweight cryptography has been a very essential for the last couple of years, induced by the absence of primitives able to run on devices with more less computing power. We can think of wireless sensor networks, RFID tags, internet of things (IoT). At the central of lightweight cryptography is a deal between lightweight and security. So many cryptographers have enlighten these issues by proposing lightweight block cipher, stream cipher, hash functions and latest authenticated encryption in cloud computing.

Cloud computing is the famous choice for the human being and their work for a purpose addition to price saving; grow production, activity and capability, efficiency, and safety. Alternately, carry folder on a hard drive or a local storage device, it feasible using cloud based storage to save all things in a remote database. Considering an electronic device has accession to the network, it has accession to the data and the software programs to run it. Cloud computing is quiet a kind of recent facility still is being used by distinct organizations from major corporations to small industry, nonprofit able to the government agencies, and balanced individual costumers. Up till now, with the stopping the hacking of data and over the top in faster computing services, Cloud computing innovation is emerging as one of the promising sustainable internet services. By the end of 2019, most of IT companies switching into cloud environment. Most of the companies are to reduce their maintainability cost of important data records. It also affects the security of the company's data. This brings the secure cloud environment showcase development in the coming years [1–3].

Curve25519 uses a fast curve for key exchange scheme for lightweight devices in cloud computing. Secure cloud environment developed from ECC encryption system is the key source of sustainable secure model which includes just about more secure data storage and doesn't devour any secure data source [4–6]. The public key of the receiver can be used with the temporary private key to derive a symmetric key such as an AES (Advance Encryption Standard) key. This key can be used to encrypt the data. Then the data is send together with the temporary public key. Then this key can be used with the static private key to derive the same AES key, which finally can be used to decrypt the data. This way of using Diffie-Hellman key agreement to keep data confidential is called IES (Integrated Encryption Scheme) or ECIES (Elliptic curve Integrated Encryption Scheme) when it is used over Elliptic Curve.

This paper exhibits the investigation on security (with the help of ECC curve25519) and confidentiality of data for the purpose of secure data storage of a system associated with cloud computing environment [7]. This paper presents one method, known as the novel secure model for data storage in confidentiality and security associated with cloud environment [8]. This novel model data storage scheme also provides the privacy of users that registered at the cloud environment. Furthermore, it improves the data security quality of the data and reduces the threats.

2. BACKGROUND OF CLOUD COMPUTING

The definition of cloud computing model, given by the NIST [9], is universally accepted, as "a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction." The cloud term extracted from network layout that was

utilized by network engineers to revealed the location of different network devices and there inter connection. The body of this network layout was like a cloud shown in fig. 1.1.

Fig. 1.1 Cloud Computing Layout

Cloud computing holds all of the excessive raising involved in processing data we carry around or sit and work at away from the device. It moves also of all that work to vast computer clusters out of the way in cyber space. The internet now has became cloud, our data, work ,and all applications are accessible from any of device in which we can connect to the internet, anywhere in the world.

2. STRUCTURE OF CLOUD COMPUTING

There are three services, delivery models, and main four deployment models [10], in The Cloud Computing model: (1) Private cloud: a cloud platform is devoted for specific organization, (2) Community cloud: computing resources are provided for a community and organization.(3) Public cloud can be accessible by the users to use the available infrastructure or to register. And, (4) Hybrid cloud: This cloud is the combination of two different cloud infrastructure (i.e. private and public, private and community etc.). They helps users to optimize the security and infrastructure with more flexibility.
The cloud computing consists of:

2.1.1 Software-As-A-Service (SAAS)

It involves the capabilities provided to the costumers to use the application running on the cloud. It provides license of software application to the consumer, through pay-as-you-go or on-demand model. Eg. Microsoft Office 365 uses this system.

2.1.2 Infrastructure-As-A-Service (IAAS)

It is a method to provide everything from the OS to the server or storage, which is done through IP- based connection (a part of On-demand service). Eg. Rackspace Cloud, Amazon EC2, IBM etc.

2.1.3 Platform-As-A-Service (PAAS)

It is a platform to create software that can be delivered through internet. Other features are similar to the SaaS. The costumer does not have the control over the cloud infrastructure (including network, OS, server or storage) but can control the settings of deployed application for application hosting environment.

2.2 Advantages of Cloud Computing

It allows users to back up their important files such as photos, videos, music which can be accessed from any device via native app or from the browser. User needs not to worry about the hard disk crash.
It is also very effective in terms of cost reduction for the business companies because IMT (information management technologies) are costlier to construct, maintain, and operate than could computing.

2.3 Disadvantages of Cloud Computing

Since the user is using a service which is available online via internet so security problem is a big issue when it comes to the personal details, financial information etc.
Several examples showed the threats of natural disaster, power outage, and internal bugs to the server maintained by the cloud computing companies.

3. MATHEMATICALLY BACKGROUND OF ELLIPTIC CURVE

Koblitz proposed the elliptic curve cryptography [11-13] initially and then further explained by Miller in 1985 and define public key cryptography, it has now became a part of the modern cryptography. In cryptography the aim of Elliptic Curve was to reduce key size, memory, less power consumption. A basic and brief introduction of ECC is shown below:

Let E_c be an elliptic curve over a prime finite field F, denoted by E_c/F, which is taken form the weistrass equation, can be defined by

$$y^2 = x^3 + ax + b \quad (1)$$

where, $a,b \in F$ and the discriminate $D=4a^3+ 27b^2\neq 0$

The point O, together with the points E_c over F, is called the infinity point or the point at infinity. Which is used for additive identity and forms an additive group A_d as:

$$A_d=\{(x , y): x,y \in F, E_c(x , y)=0\}U\{0\} \quad (2)$$

Let m(very large) be the order of $\acute{A}_d$, is defined as:

$$mxG \bmod q = O$$

where G is the generator of A_d and A_d is a cyclic additive group (under point addition "+") defined as:

$$S + O = S, \text{ where } S \in A_d. \quad (3)$$

The scalar point multiplication over A_d can be defined as

$$tS = S + S +\ldots..+ S \ (t \text{ times}) \quad (4)$$

Let S and Q are two distinct points on the elliptic curve and $S \neq -Q$. A line joining S and Q are drawn and extended until it crosses the elliptic curve at third point $(-R)$. $-R$ is the point of tangency to the curve. This point $(-R)$ is then reflected over negative x- axis. The addition of the points S and Q is defined as:

$$S + Q = R \quad (5)$$

The security strength of the ECC lies on the complexity of solving the Elliptic Curve Discrete Logarithm Problem (ECDLP) and it insures same level of security of RSA with small bit size key, which is shown in the Table 3.1.

Table-3.1 NIST Recommended Key Size

Symmetric Key Size (bits)	RSA and Diffie-Hellman Key Size (bits)	Elliptic Curve Key Size (bits)
80	1024	160
112	2048	224
128	3072	256
192	7680	384
256	15360	521

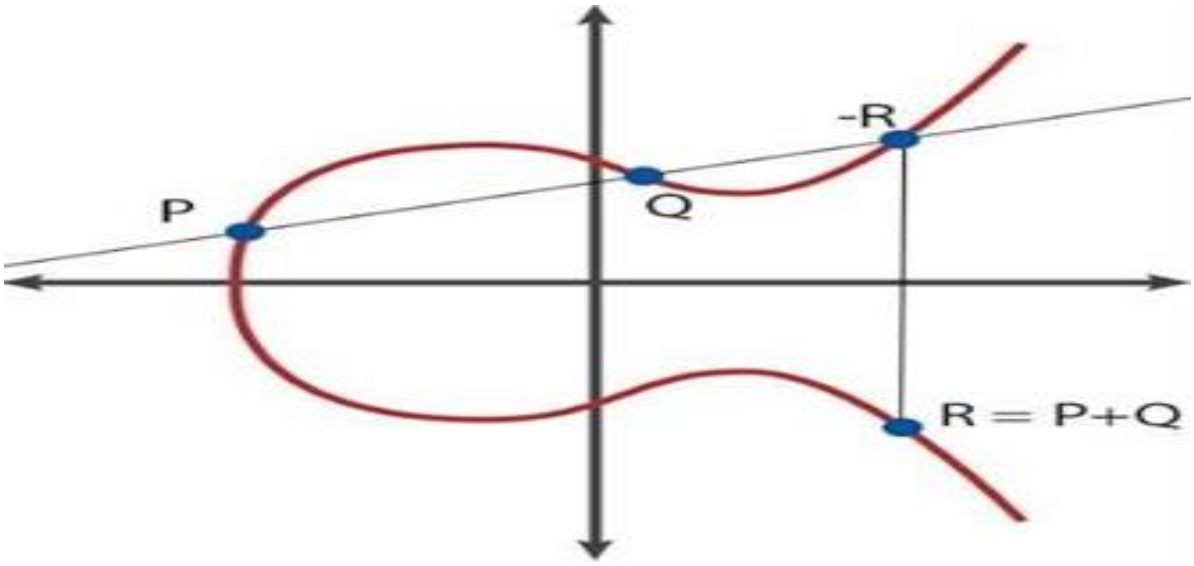

Fig. 3.1 Elliptic Curve

3.1 Montgomery Form of Elliptic Curve

One needs to understand the basic theory of elliptic curve before going through the Curve25519.The basic theory of elliptic curve given by Montgomery form [14], is discussed in this paper. This form introduced by D.L Montgomery after motivated by the EC weierstrass form by miller and koblitz. An elliptic curve over F (in Montgomery form) is defined by an affine equation.

$$M_{(A, B)} = By^2 = x^3 + Ax^2 + x \quad (6)$$

where 2 $A \in 2 + 4Z$ or
$\forall$ (A-2)/4

Where, A and B are parameters in F satisfies $B = 0$ and
$A^2 \neq 4$. Transform the projective plane coordinates $(X : Y : Z)$, with $x = X/Z$ and $y = Y/Z$, we have the projective model of Montgomery form as:

$$M_{(A,B)} : BY^2 = X(X^2+AXZ+Z^2) \subseteq P^2 \quad (7)$$

There is a point $O = (0 : 1 : 0)$ at infinity on $M_{(A,B)}$. It is the only point on $M_{(A,B)}$ where $Z = 0$.

A typically MONTGOMERY curve is shown in fig. 3.2.

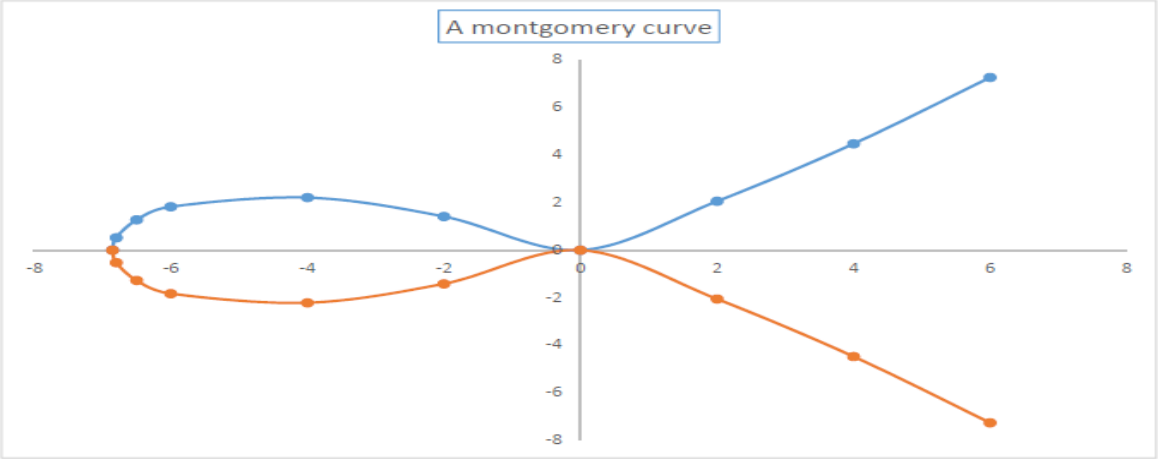

Fig. 3.2 Montgomery Curve ($3y^2 = x^3 + 7x^2 + x$)

3.2 Basics of Curve 25519

In cryptography, Curve25519 offers 128 bit of security. The purpose of introducing this curve was to obtained new speed record for high security of Diffie-Hellman computations. The representation of data flow from secret keys through public keys to a shared secret shown below.

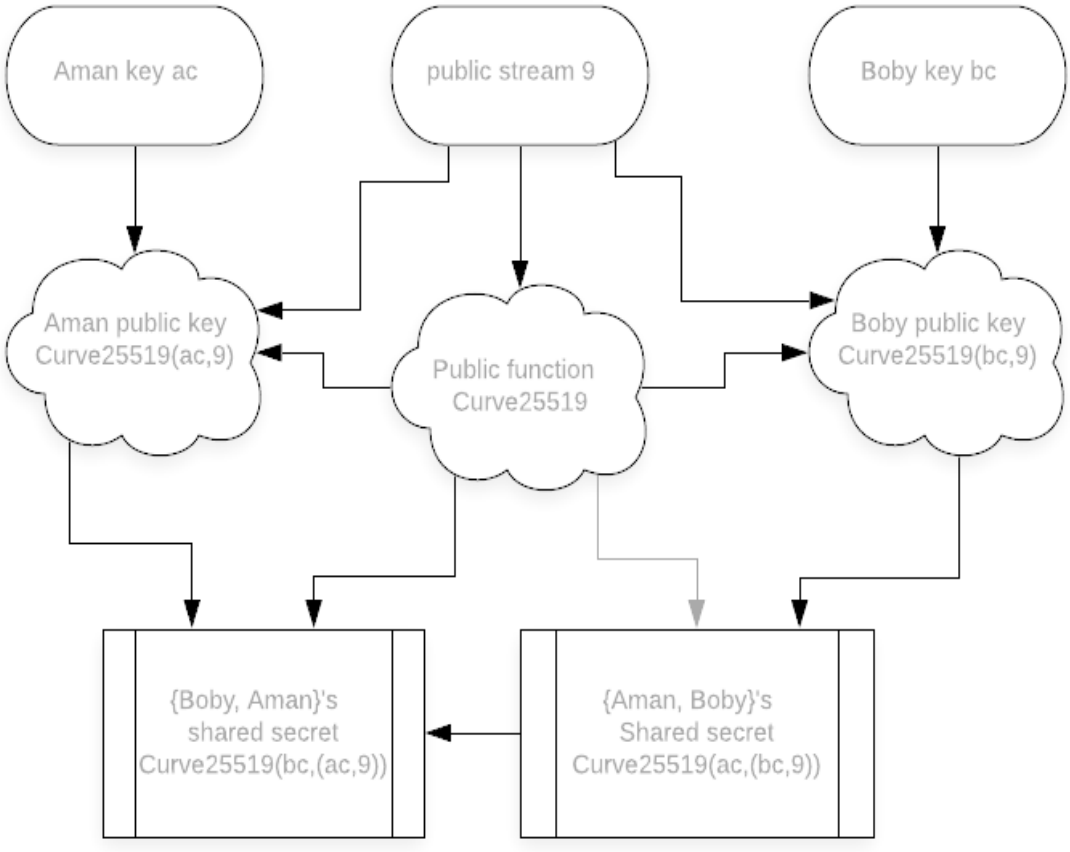

Fig. 3.3 Curve25519 Shared Secret

In cryptography, D.J Bernstein [15] has found widespread use of curve25519 which comes from montgomery form (3). This curve25519 is the form of Montgomery curve with $A = 486662$ and $B = 1$ defined over Fp (field) with prime $P\text{^}2255\text{-}19$ and it uses the base point $x=9$. This prime satisfies $p = 1 \bmod 4$, and is of index 8.
One point of Curve25519 is the set (E) of all roots of the polynomial satisfying equation

$$y^2 = x^3 + 486662x^2 + x \quad (8)$$

This set E form a group Over the field(F).
Every point written as affine coordinates in conjunction with a coordinates (x, y) with a further point written referred to as O called the point at infinity that time that doesn't have a define in affine coordinates.
We can define Curve25519 is the set of all lines of roots of the polynomial in projective coordinate system as

$$Y^2 Z = X^3 + 486662XZ^2 + XZ^2 \quad (9)$$

over Fp, each line written as homogeneous coordinates $(X:Y:Z)$ for X,Y,Z not all zero,Where $(X:Y:Z)$ Define the same line as $(\lambda X:\lambda Y:\lambda Z)$ for any nonzero λ. The point at infinity has homogeneous coordinates $(0:1:0)$, and thus need not be treated separately as it is in affine coordinates. Any other point with affine coordinates (x,y) can be define with homogeneous coordinates $(x:y:1)$, and any line with homogeneous coordinates $(X:Y:Z)$ for nonzero Z can be define with affine coordinates $\left(\frac{X}{Z},\frac{Y}{Z}\right)$.

4. PROPOSED SCHEME

Our propose cloud storage system, is more secure, faster to store the data into cloud system. The trust is important aspect between cloud provider and user, that is be buildup by data security by the cloud service provider. Our model working between user and cloud server uses the start to finish encryption for entire files and video storage on cloud storage with the goal that nobody, not even provider, shall approach to the content of user data.

4.1. Illustration of The Suggested Scheme

Our propose cloud storage system, is more secure, faster to store the data into cloud system. The trust is important aspect between cloud provider and user ,that is be buildup by data security by the cloud service provider .Our model working between user and cloud server uses the start to finish encryption for all files and video storage on cloud storage so that no one, not even provider, shall access to the content of user data. Next step is to security on to cloud storage place. This will be achieve by second phase the system where data broken into the n file size and encrypt broken data using encryption algorithms and stored into the different cloud nodes. Encryption and decryption process having steps perform into cloud node.
The digital security on cloud has become even more important. It's verified with a cryptographic lockage, and simply the receiver has the keys. Moreover, the keys change with each and every cloud storage that is sent.
Our Propose model is using CURVE25519 for key generation Curve 25519 is one of the most widely used ECC methods for lightweight cryptographic schemes.

4.2. Working Step of Proposed Scheme

- Cloud User registers their information and gets the start secured file sharing to the Cloud Meta Server. Our proposed
- model shall be using End-to-End encryption.[16]
- Cloud Storage Nodes are enabling for sharing and storing the data.
- For each storage node the server of the data to be stored start to finish encryption.
- Afterward, start to finish encryption, Server gets the current available storage node keys (n) and generates the MD5 (message digest) value for the storing the file like video ,file and mp3 generates the MD5 values for each Storage key.
- Apply the xor Hash function between the File Name MD5 value and the Cloud StorageNode keys MD5 values.
- Generate the 16 byte key (dynamic key) from the above exor output.
- The random keys and the cloud node ip's will be stored by the server.
- Cloud server split the files into n parts.
- Encrypt each File with their appropriate keys using algorithm(ECC Encryption algorithm).
- Afterward, We need to convert encrypted file into jar file.
- Afterward, send the Jar file to their appropriate cloud nodes for storing purpose.
- The download request then cloud server will generate the MD5 value for their requested file and generate the MD5 value for the random keys of storage file.
- Apply the xor Hash function between the requested File Name MD5 value and the Storage keys MD5 values.
- Get the files from the storage nodes. Unzip jar files.
- 15Decrypt the files using decryption algorithm(ECC Decryption algorithm) and combined the file split parts, then done End-to-End encryption between user and cloud server.

4.3. Proposed Cloud Scheme Encryption Technique Keys

CA= Sender
CB= Receiver
CServer=Cloud Server
CI= Cloud Identity key
CS= Cloud Signed
CO=Cloud On Time Password
CE= Cloud fleeting Key for session start
CD= Cloud fleeting Key for Chain Key
Round Keys
Cloud Master Key= CM (Drive the Cloud Root Key)
Cloud Root Key= CR(Use to drive Cloud Chain Key)
CC= Chain Key(Cloud Chain Key)
CFM_{ck} =Cloud File Sharing Key

In the cloud storage system, there are Keys description as given below:

4.3.1 Cloud_ID Key Pair

An endless Curve25519 cloud key pair, is to be produced at the cloud software by user(I).

4.3.2 Cloud_Signed CPre Key

A moderate term Curve25519 key pair, produced at registration time on cloud server , signed(S) by the Cloud_IDKey, and rotated based on periodic timed.

4.3.3 Cloud_One-Time CPre Keys

A Curve25519 cloud key pairs queue for Cloud one time utilize(O), produced at user registration on cloud software, and replevisable as required.

4.4 Cloud_Session Key Types

4.4.1 Cloud_Root Cloud_Node Key

A 32-byte value which is for utilize to create Cloud Storage Keys(R).

4.4.2 Cloud Storage Key

A 32-byte esteemed value that is utilized to make Cloud Data storage Keys fleeting key (E) for session initiation, fleeting key(D) for chain key generation.

4.4.3 Cloud Data Storage Key

A eighty byte key esteemed value that is utilized to encrypt data storage information. 32 bytes for HMAC-SHA 256 key, 32 bytes for AES-256 key, and 16 bytes are utilized for a dynamic access .[17]

4.5 To Cloud Data Storage Session

The cloud user petitions the public Cloud_IDKey I_B for cloud node, public Cloud_Signed CPre Key S_A for user, and Cloud_Single-Time CPre Key O_A for user.Fleeting key E_B for cloud. The Cloud server restores the mentioned open public key qualities.
Cloud_OneTime Key is only utilized once, thus it is eliminated from Cloud server restores later on being mentioned. If the cloud nodes newly batch of Cloud_OneTime CPre Keys are utilized and therefore the cloud node has not replevisable them, no Cloud_OneTime CPre Key are going to be came back. The cloud user store the cloud nodes similar Key as Icloud node, the Cloud_Signed Cloud_Pre Key as Scloud node, and therefore the Single-Time Pre Key as Ocloud node.The cloud user procreates an transitory key pair of Curve25519, Euser_Cloud and loads own Identical Key as CIuser.The cloud user calculates a cloudnodemaster key(M) as cloudmaster_secret = ECDHA(CIuser, Scloud node) || ECDHA(Euser_Cloud , Icloud node) || ECDHA(Euser_Cloud , Scloud node) || ECDHA(Euser_Cloud , Ocloud node). If there is no Cloud_One Time Pre Key, the final ECDHA is omitted.The cloud user uses HKDF to create a Cloud_Root Key and Cloud Storage Keys from the cloudnode_master secret.

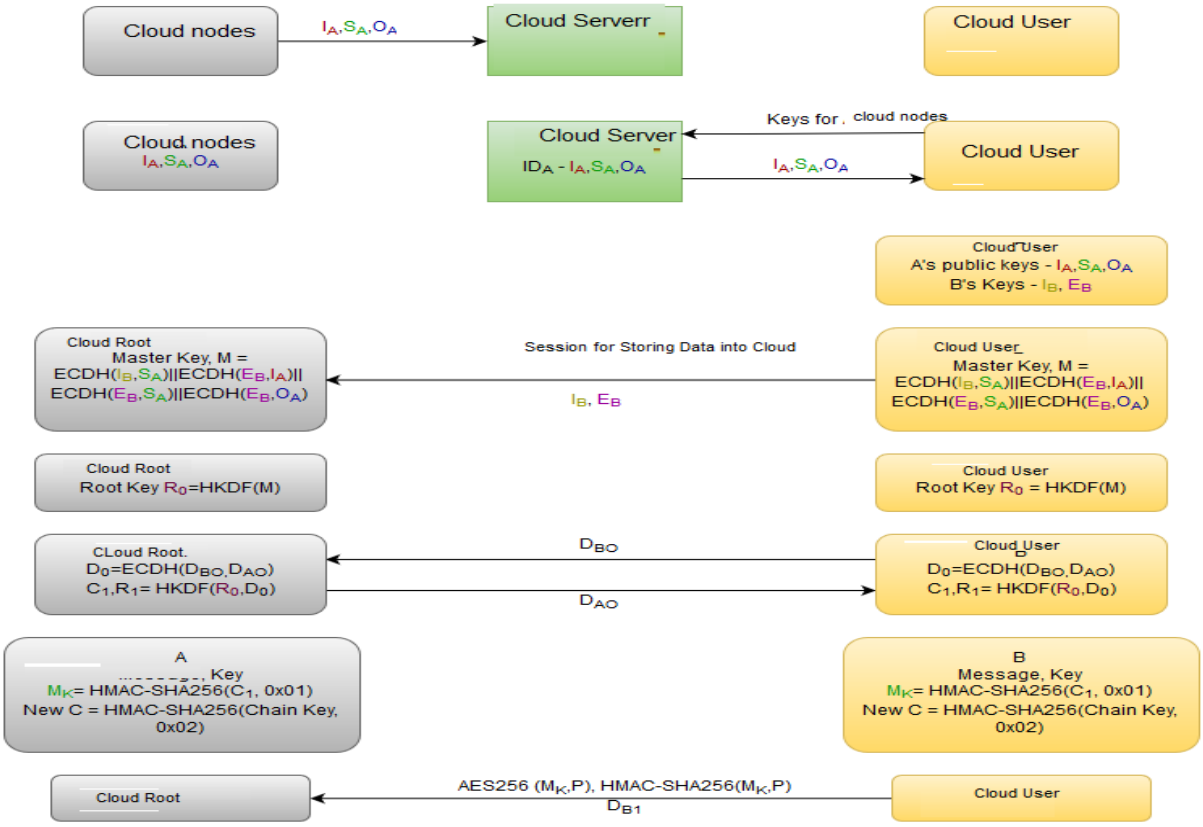

Fig. 4.1 Cloud Storage Encryption

4.6 Cloud Storage Encryption Process

This scheme will provide high level security of data. The cloud security model for data sent/download to cloud storage End to End encryption. Afterward we apply encryption algorithm and file is split into two or more parts ,store into different clouds node and apply encryption ,decryption algorithm . The last step before sending the file is converted splits files into jar file. Our scheme is using hash cloud data using SHA256 to obtain a condensed version of data. This is for assure the data integrity. The hash code may be an operate of all the bits of the info storage and provides an error-detection potential. An amendment to any bits within the knowledge storage leads to a change to the hash code. The concatenation of (data storage + hash code + the encryption–decryption key) is encrypted exploitation AES256. The hash code gives the structure or excess expected to accomplish authentication. Since encryption is employed to the whole data storage in addition to hash code, privacy is additionally given. In this model we change the encryption–decryption key every session to provide the key confidentiality. Our ECC-based algorithm is light-weight algorithm. This curve based algorithm is faster, secure and light-weight for storing the information into cloud storage.

4.7 Illustrative Example

In this paper we are working for security
Example
Suppose User registration number is **9789901234**
Cloud Server Node :**012345678965432**
This is Crypt Key Generation
CloudStoragekey Crypt Key Generation
This is base of key generation
CUseriD Seed: **9789901234**
CloudUserNumer: **012345678965432**
Password: store in this folder
0-26 of key These are padding information that is require for Key Generation
Padding: **Multiple variations** "0x0,0x0,0x1 / 0x0,0x1,0x1 / 0x0,0x1,0x2 / Etc"
Random Key: 98C0405227C6DDA4BF61B93C844712794B8ECED89D434E16A448F7648CCB8414
Random IV: 386B2E9A8B5551C576C76BCA256FF2B2 (unused since Cloud storage key)
Paired Key: C792C53169EBD5D960F3D369BC335BD00A8EA847B3952359D941AC7AC0D8EC5F
Static IV: 28C35A3AA77C0B661D750A840081F68B (key zeroed)
Static Key: 8627527AED8CF3345362B7D156B12B58E552BDF8BDFA4215B76DD525EAEFD9E1
DECRYPTION KEY
Algorithm for Cloud Server Key
This algorithm is for cloud server key generation that is used for end to end encryption

4.8 Cloud Storage Key Generator Algorithm

```
Procedure cloudgenerateKey
String Cloudseed, String cloudnodenum;
  if (!cloudnodenum.matches("^[0-9]{15}$"))  then
    display ("\nInvalid CloudUserNumer Number (Expecting 15 Digits)\n");
 otherwise
    String randomIV = gen_RandomIV();
    String staticIV = gen_RandomIV();
    String randomKey = gen_RandomKey();
    String staticKey = getStaticHex(Cloudseed,randomKey);
    String pairedKey = getStaticHex(CloudNodevalue,randomKey);
 endif
```

4.8.1 Procedure Cloud Retrieve Key

```
String Cloudseed, String cloudnodenum, String file
    File dbFile = new File(file);
    if (!cloudnodenum.matches("^[0-9]{15}$")) then
      display"\nInvalid CloudUserNumer Number (Expecting 15 Digits)\n")
    otherwise
      InputStream DB = new BufferedInputStream(new FileInputStream(dbFile));
      byte[] Data = new byte[67];
      DB.read(Data);
      byte[] rKey = new byte[32];
      System.arraycopy(Data, 3, rKey, 0, 32);
      byte[] rIV = new byte[16];
      System.arraycopy(Data, 35, rIV, 0, 16);
      byte[] sIV = new byte[16];
      System.arraycopy(Data, 51, sIV, 0, 16);
      DB.close();
            String randomIV = bytesToHex(rIV);
```

```
String staticIV = bytesToHex(sIV);
String randomKey = bytesToHex(rKey);
String staticKey = getStaticHex(Cloudseed,randomKey);
String pairedKey = getStaticHex(CloudNodevalue,randomKey);
display ("\nMode: CloudStoragekey Crypt Key Recovery")
```

5. RESULT AND DISCUSSION

5.1. Cloud Report Simulation Tool

CloudReport is an extended version of CloudSim[8] and this tool is for utilize for cloud computing algorithm. This simulator is easy to utilize and when we run the code on netbeansgui interface is to display in fig. 3.3. This simulation tool interface screen provides the button for running the simulation, provides to environment button that will create cloud environment for running the algorithm and Provider option is for setting the datacenter values shown in fig. 4.1. The Customer button is for authorize customers cloud environment whose want to sharing the information into cloud computing environment. Afterward, running the simulating the result, need to Run simulation button for environment as shown in fig. 5.1 and fig. 5.2.

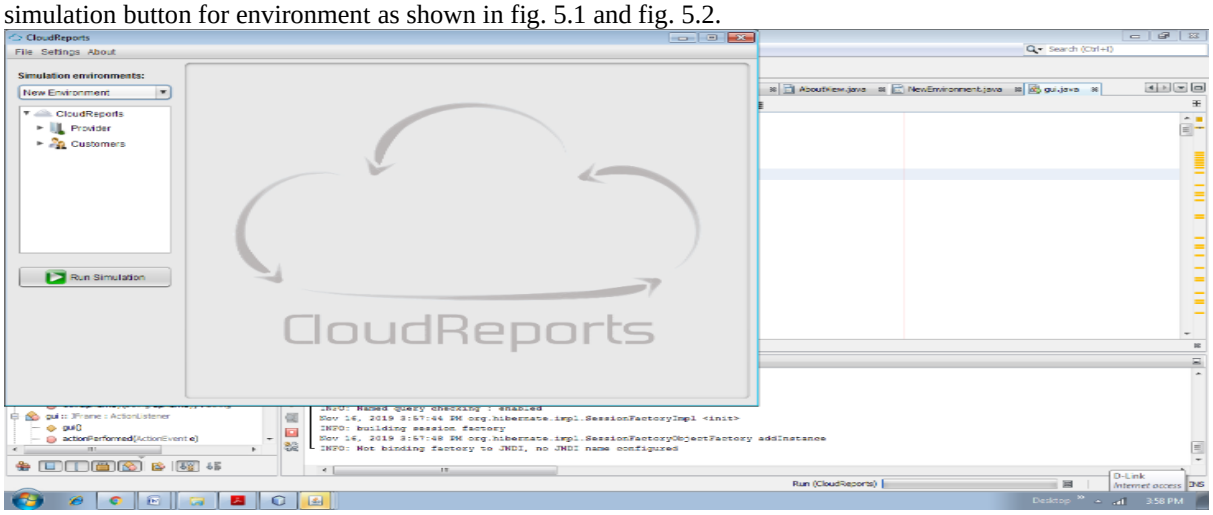

Fig 5.1 Open Cloud Report

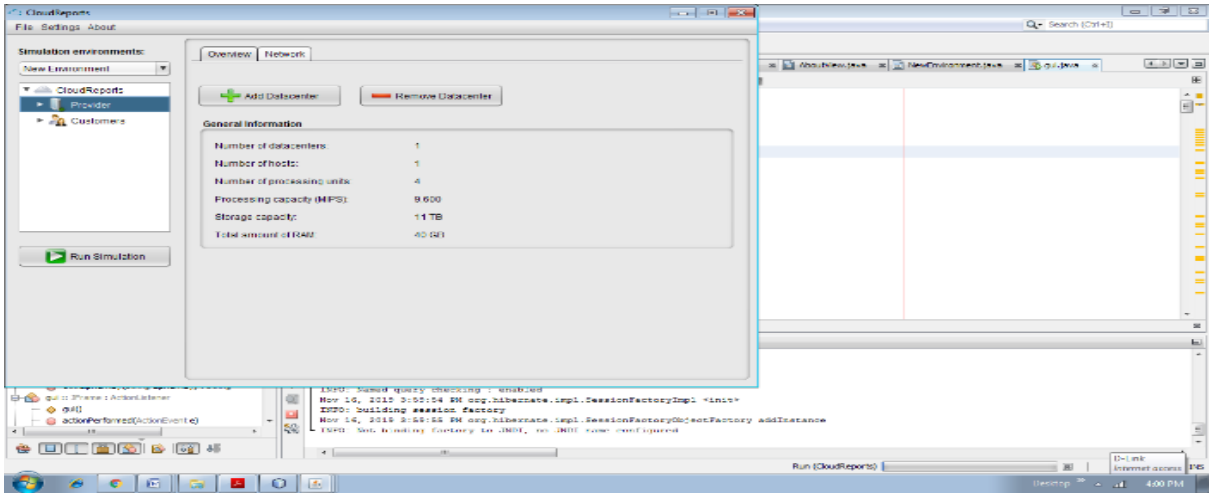

Fig. 5.2 Data Center and Environment

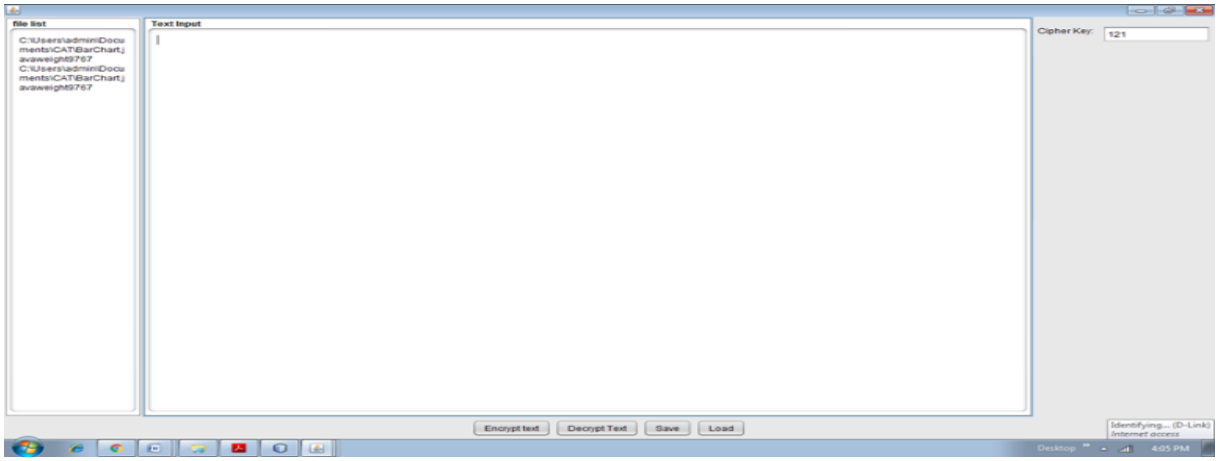

Fig. 5.3 Simulation on Cloud Report

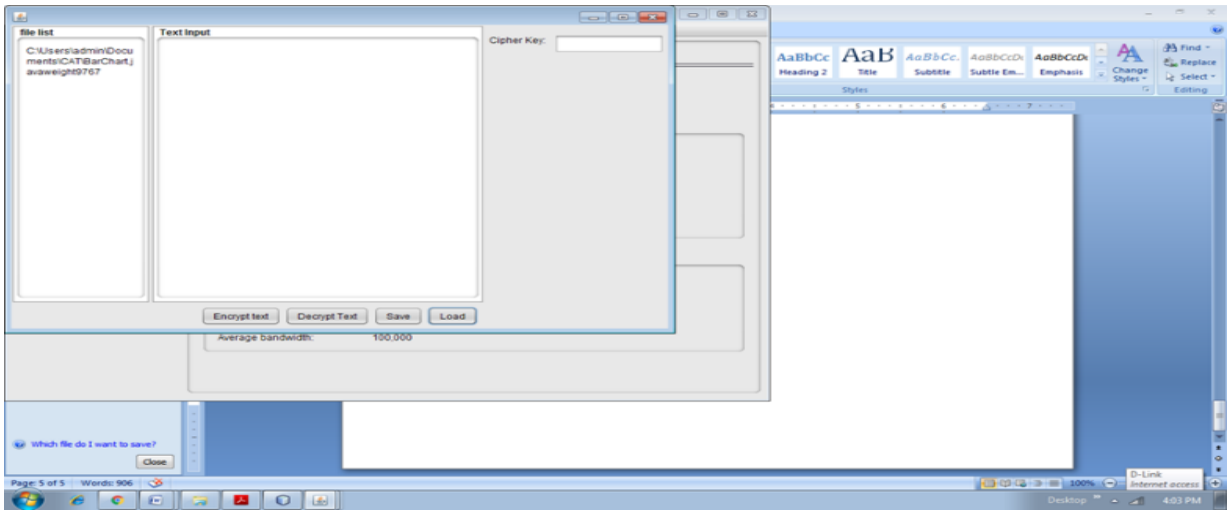

Fig 5.4 Simulation on Cloud Report

Table- 5.1 Splited Parts of into Cloud Storage

Split File Names	Converted File	File Size	Cloud Storage Location
Partfile1.java	Partfile1.jar	128 bits	CloudNode one
Partfile2.java	Partfile2.jar	128 bits	CloudNode two

Table-5.2 Required Key Length in Bits for Equivalent Security

Symmetric	RSA/DH	ECDH
80	1024	160
112	2048	224
128	3072	256
192	7680	384
256	15,360	512

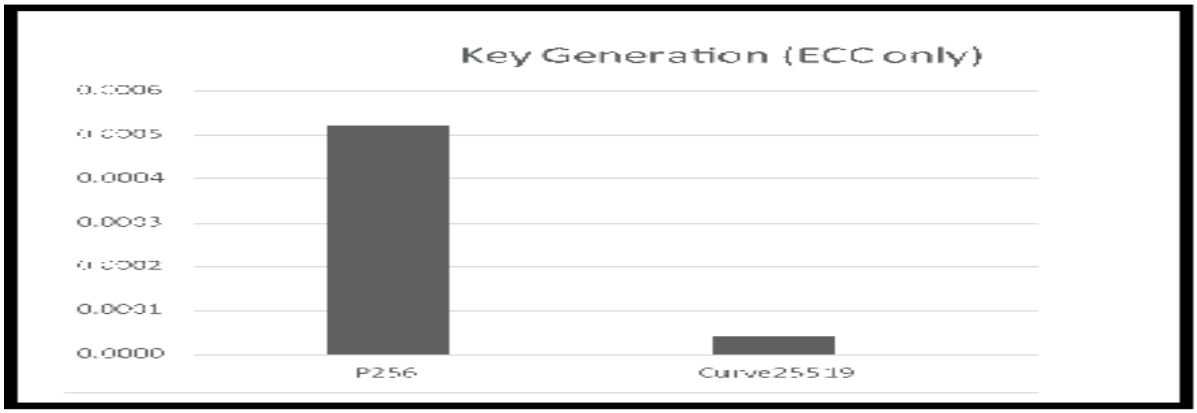

Fig. 5.5 Time Graph (in seconds) For Single Key Pair Generation

Fig. 5.6 Time Graph (in seconds) For A Secret Key Exchange

CONCLUSION

In this paper, we proposed an CURVE25519 base security scheme for cloud environment user. We use simulation tool Cloud Report that is a GUI simulation tool. According to our simulation result, the CURVE25519 key generation based cryptography model is efficient and secure model for cloud user. A novel security model is designed for store data storage and secure cloud environment system. The main purpose of the security model is to reduce hacking of data information as well as keep sensitive and confidential information. Moreover, several methods of ECC model are explained in literature. The efficacy of the designed model is tested in Cloud Reports. The security behavior in terms of encryption is also measured. The study reveals that the encryption quality has significantly improved.

REFERENCES

[1] Y. Li, K. Gai, L. Qiu, M. Qiu and H. Zhao, "Intelligent cryptography approach for secure distributed big data storage in cloud computing," Information Sciences, vol. 8, no. 5, pp.1-13, 2016.
[2] "Cloud Computing Definition Gartner". http:// www.gartner.com/newsroom/id/1035013
[3] S. Manjula, M. Devi, and R. Swathiya, "Division of data in cloud environment for secure data storage," 2016 IEEE International Conference on Computing Technologies and Intelligent Data Engineering, pp. 265-269, 2016.
[4] R. Buyya, R. Ranjan and R.N. Calheiros. "Modeling and Simulation of Scalable Cloud Computing Environments and the CloudSim Toolkit: Challenges and Opportunities. High Performance Computing & Simulation", 2009. HPCS'09. International Conference on. IEEE, 2009.
[5] P. Mell and T. Grance, "The NIST definition of cloud computing,", Computer Security Division, Information Technology Laboratory, National Institute of Standards and Technology, Gaithersburg, pp. 1-7, 2011.
[6] M.A. Nadeem, "Cloud Computing: Security Issues and Challenges," Journal of Wireless Communications, vol. 1, no. 1, pp.10-15,2016.
[7] H. Casanova, A. Legrand and M. Q. Simgrid: "A generic framework for large-scale distributed experiments. In Computer Modeling and Simulation", 2008. UKSIM 2008. Tenth International Conference on, p. 126-131, 2008.
[8] A. Vashistha, R. Porwal and A.K Soni, "A Taxonomy of Scheduling Algorithms for Cloud Computing", IJCSI, pp67-71,2015.
[9] Peter Mell, and Tim Grance, "The NIST Definition of loud Computing, "2009, http://www.wheresmyserver.co.nzlstorage/medialfaq] files/clouddef- vI5.pdf, Accessed April 2010.
[10] N. Y Chong "Cloud Computing Challenges in a General Perspective" Issue 1. Volume 3. January 2019.
[11] Victor S. Miller. "Use of elliptic curves in cryptography". In Hugh C.Williams, editor, CRYPTO'85, volume 218 of LNCS, pages 417–426,Santa Barbara, CA, USA, August 18–22,. Springer, Heidelberg, Germany, 1986.
[12] N. Koblitz. "Elliptic curve cryptosystems". Mathematics of Computation,48:203–209, 1987.
[13] W. Stallings, "Cryptography and Network Security Principles and Practices", Prentice Hall, 4th Edition, pp 420-430, 2009.
[14] S. Iskandar "Elliptic Curve Cryptography: Algorithms and Implementation Analysis over Coordinate Systems" 24 November 2014.
[15] D.J.Bernstein, "Curve25519: New Diffie-Hellman speed records in Public Key Cryptography"—PKC, Lecture Notes in Computer Science. Vol. 3958. New York, USA: Springer; pp. 207-228, 2006.
[16] M. Mamatha and P. Kanchan, "use of digital signature with diffiehellman key exchange and hybrid cryptographic algorithm to enhance data security in cloud computing." International Journal of Scientific and Research Publication, vol. 5, no. 6, pp.1-4, 2015.
[17] R. Manro, T.D.S. Dua, and A.S. Joshi, "Ensures Dynamic access and Secure E-Governance system in Clouds Services – EDSE," International Journal of Applied Engineering Research, vol. 11, no. 1, pp. 731-737, 2016.