Cloud Computing Security Framework based on Elliptical Curve

Ayman Helmy Mohamed
Computer Science Dept.
Faculty of Computers and Information
Helwan University, Egypt

Aliaa A.A. Youssif
Computer Science Dept.
Faculty of Computers and Information
Helwan University, Egypt

Atef Z. Ghalwash
Computer Science Dept.
Faculty of Computers and Information
Helwan University, Egypt

ABSTRACT
Technological advancements in cloud computing due to increased connectivity and exponentially proliferating data has resulted in migration towards cloud architecture that employ dynamic load and access balancing. Cloud computing is technology where the clients' can use high end services in form of software that reside on different servers and access data from all over the world. With a promising technology like this, it certainly abandons clients' privacy, management of data and services, putting new security threats towards the assurance of data in cloud. However, there are some security concerns when clients handle and share data in the cloud-computing environment. The security threats such as maintenance of data integrity, data hiding and data safety dominate clients concerns when the issues of cloud security come up. The big data and time-consuming encryption calculations related to applying any encryption method have proved as a hindrance in this field. Cryptography is knowledge of protecting the information for providing encryption techniques

In this paper, Cloud computing security framework was tested. The framework ensures a trusted cloud environment that controlled by both the client and the cloud environment. The proposed solution secures the movement of data between client and cloud end. the non-breakability of ElGamal based on Elliptic curve cryptography for data encryption/decryption was used along with Diffie Hellman based on elliptic curve mechanism for connection establishment, The data integrity verification is done by introducing a message digest tag for updating data based on SHA-256. The proposed encryption mechanism uses the combination of elliptical cryptography and SHA-256 methods.

General Terms
Cloud computing, Security, elliptic curve discrete logarithm.

Keywords
cloud computing, Elliptic curve, Diffie Hellman based on Elliptic curve, cloud storage, Security, cryptography, ElGamal, hash256, ECDSA, DSA.

1. INTRODUCTION
Cloud computing is a new emerging technology that is arousing the interest of industry and academics. It helps in overcoming the out of cost control in terms of electricity generation, personal hardware and storage. The cloud computing allows consumers and businesses to use applications software without installation, in addition to the ability to access personnel files with any computer has access to the internet. Therefore, it encourages a significant numbers of enterprises to move to it [1]. According to U.S. National Institute of Standards and Technology (NIST): "Cloud computing is 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" [2].

Cloud computing delivers services over the Internet. Cloud services can be accessed through a web service application-programming interface (API), or through a Web-based client interface. The cloud services can be rented in a pay per use manner to fully featured applications, to software development and deployment environments, and to computing infrastructure assets such as network-accessible data storage and processing [3]. Cloud computing bases on the concept of virtualization. Virtualization separates everything such as application software, platform and infrastructure from the hardware you are working on it [4]. Cloud computing promotes availability. It composed of five essential characteristics named Abstraction of infrastructure, Resource democratization, Service-oriented architecture, Elasticity/dynamism, and Utility model of consumption and three service models named software as a service, platform as a service and infrastructure as a service and four deployment model named public cloud, private cloud, hybrid cloud and community cloud [5].

Cloud computing security issues posed great challenges to the industry, academics and information technology community. It has been argued by critics that it is not secure and data integrity compromises. Security is the primary obstacle hindrance the wide adoption of this promising computing model. Especially for customers whom their confidential data is consumed and is produced during the computation process [3].

Cryptography means hidden, that means hide the existence of information or make the information secure from intruders. There are two types of cryptography symmetric and asymmetric cryptography. Encryption of data is the process of converting data from normal plaintext to unreadable cipher text. This conversion involves huge and very complex mathematical computation. In Cloud computing domain, there are set of important policies, which include issues of privacy, anonymity, security, liability and reliability. The most important of these issues is the data security and how cloud providers assures it. Most effective technique to protect clients' data is its encryption. Different encryption schemes for protection of data have been in use for many decades [6].

The rest of this paper is organized as the following. issues of cloud security is presented in section 2. Literature review is presented in section 3. Problem statement along with elliptic

curve cryptography are presented in section 4, 5 respectively. Diffie Hellman key based on elliptic curve is presented in section 6. The proposed solution is presented in section 7. Finally, results, conclusion and future scope are presented in section 8, 9 respectively.

2. ISSUES IN CLOUD SECURITY

The cloud computing issues of security named confidentiality, integrity and availability; known as the AIC. Availability is the attestation that data will be available to the client in a perpetual manner irrespective of location of client. It either used to ensure that legitimate clients are not, by accident or by malicious intervention, denied access to services to which they are entitled [2, 3]. Integrity is the assurance that the data sent is same as the message received. It was ensured by Firewalls and intrusion detection. Cloud computing is an elastic environment. Due that, the traditional technologies to ensure data integrity may not be effective. Migrate out of or into the cloud, storage will consume the client bandwidth and amount of time. A demand for directly verify the integrity of data in cloud storage without having first to download the data and then upload it [2, 3]. Confidentiality is avoidance of unauthorized exposé of client data. It was ensured by security protocols, authentication services and data encryption services. The common solution for data confidentiality is data encryption [2, 3].

3. LITERATURE REVIEW

A secure distributed storage system is developed. The distributed storage system not only supports secure and strong information storage and retrieval, but also lets a client forward his information within the storage servers to a different client while not retrieving the info back. The proxy re-encryption supports encoding operations over encrypted messages. It represents forwarding operations over encoded and encrypted messages. It was analyzed the parameters for copies of a message sent to storage servers. Along with determining the number of storage servers queried by a key server. These parameters permit additional flexible adjustment between the amount of storage servers and robustness [7]. It presents a secure cloud storage system that gives secure information storage and secure data forwarding practicality during a decentralized structure. Moreover, every storage server severally performs encoding and re-encryption and every key server severally perform partial decryption

The author proposed elliptical curve cryptography for encryption and decryption. It [8] was reviewed various authentication and encryption mechanism that are applied for data security in cloud computing. It implemented the ECC algorithm for the data security in the cloud computing. It was resulted that elliptic curve cryptography is a good approach with smaller size of data for security purposes. ECC is a secure tool. It was used to secure the Cloud Application. It provided a robust and secured model for the development and deployment of secured application in Cloud. [1]. Algorithms for data storage security are explored in cloud and desktop environment [9]. There are various algorithms such as ECC, RSA, RC4 and ElGamal have been suggested and discussed for cloud computing. It used to prevent any unauthorized client trying to access the confidential data to access the cloud. It was very noticeable that the usage of ECC in wireless devices is more preferable than public key encryption. It helps in longer the lifetime of batteries or of the devices.

A modified RSA algorithm is developed in [10] based on offline storage and prime number. A modified RSA was speeded due to storing all parameters in RSA algorithm before starting. It was used the index for transferring the public and private at the time of encryption and decryption of data between the networks. The modified RSA algorithms is based on original RSA algorithm but it used three prime numbers p, q and r instead of two (p & q). Elliptic curve cryptographic schemes was analyzed for cloud applications. It [11] was compared with RSA based schemes. The results recorded that elliptic curve scores over RSA because of less key generation time. The overall difference increases with the growth in key size.

It was secured the movement of data at client and server end using the non-breakability of elliptic curve cryptography. ECC was used for data encryption and Diffie Hellman Key Exchange mechanism for connection establishment. The proposed model was composed of three security checkpoints: authentication, key generation and encryption of data. [6]. It was proposed hiding data locations from the clients when they store and retrieve it. The data is stored at multiple places over the information space. It can be retrieved using proper authentication [12]. It was implemented in infrastructure as a service layer to improve confidentiality and integrity of virtual machines. It was also integrated with secure resources management schemes to get more controlled isolation environment. Finally, a prototype was implemented to demonstrate the system feasibility and performance.

Elliptic Curve Digital Signature Algorithm (ECDSA) was proposed for creation and verification of digital signatures. ECDSA was used to secure data uploaded by clients. A design for providing a secured environment for activities in the Cloud was proposed in [13]. It was used to provide solution for a secured Cloud environment with improved performance in computing power and battery resource usage. The proposed solution was very attractive for mobile applications. It provided a robust and secured model for the development and deployment of secured application in the Cloud [1].

It was used elliptic curve discrete logarithm problem (ECDLP) to ensure the result of secure computation outsourcing and Uses Duality Theorem (DT) to derive a set of necessary and sufficient condition for result verification. It was explored a robust algorithms to achieve concurrent programming, explore the sparsity structure of problem for further efficiency improvement, establish formal security framework, and finally extend results to a big data server-storage on outsourcing in cloud [3].

4. PROBLEM STATEMENT

Cloud computing security has become a hot topic in industry and academic research. However, data security in cloud computing has become the leading cause of hampering its development. As soon as the client logs in cloud system for the first time, client should make an account in the cloud system. As the account created. Client may be individual or organization. Client has large amount of data to be stored in cloud. On the other hand, client has a thin or dump devices to access the cloud services. Cloud storage server (CSS) is entity. It has large storage space and computation resources for client. Cloud service provider (CSP) maintains it. In this work Clients will be enabled to ensure that their data being correctly stored and maintained.

5. ELLIPTIC CURVE CRYPTOGRAPHY

An elliptic curve (E) is the set of solutions to Weierstrass equations

$$Y^2 = X^2 + AX + B \quad (1)$$

$$4A^2 + 27B^2 \neq 0 \quad (2)$$

The equation (1) is represent elliptic curve equation where "A" and "B" is a constant, together with an extra point "O" where A and B must satisfy equation (2) [14].

The elliptic curve discrete logarithm problem (ECDLP) is: given an elliptic curve named "E" defined over F_p and two points, $g, q \in E$ find an integer x such that equation number 3 is satisfied.

$$Q = r * g \ (mod\ q) \quad (3)$$

Where (r) is called the discrete logarithm of Q to the base g. Figure 1, and 2 represent graphically points' addition and points' doubling respectively.

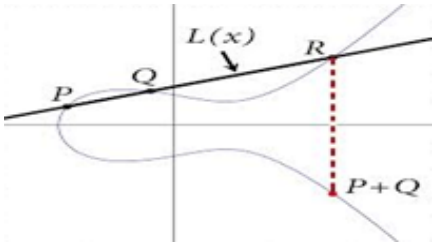

Fig 1. Point addition of P and Q

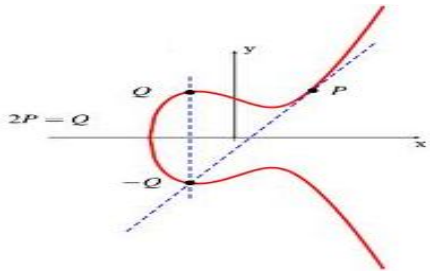

Fig 2. Point doubling representation

Point addition algorithm for two points p_1, p_2 over a curve named E it has also a point at infinity named O:

Add (E, p_1, p_2)

If $p_1 == O$ then return p_2

If $p_2 == O$ then return p_1

If $(p_1.x == p_2.x)\ and\ (p_1.y \neq p_2.y\ or\ p_1.y == 0)$

Then return "O"

If $((p_1.x == p_2.x)$ then $\lambda = (3\,p_1.x^2 + A)/2p_1.y$

Else $\lambda = (p_2.y - p_1.y)/((p_2.x - p_1.x)$

$p_3.x = \lambda^2 - p_1.x - p_2.x$, $p_3.y = \lambda(p1.x - p_3.x) - p_1.y$.

The plaintext encoding should be done before encryption. The decoding should be done after decryption. ECC Encryption and Decryption methods can only encrypt and decrypt points on the curve and not messages. The Encoding convert message to a point on the curve. The decoding converts a set of point to a message. Encoding and decoding are important functions in Encryption and Decryption in ECC. Koblitz's [15] method was discussed to represent a message to a point on the curve and vice versa. The Execution time for encoding and decoding depend on the values of ECC domain parameters. The Execution time for Decoding is negligible compared to that of Encoding.

6. DIFFIE HELLMAN KEY EXCHANGE BASED ON ELLIPTIC CURVE

Diffie Hellman [16] establishes a shared secret. It can be used for secret communications by exchanging data over a public network. Table 2 illustrates the general idea of the key exchange by using Elliptic curve points instead of a very large number. The key part of the process is that Alice and Bob exchange their secret points in a mix only. Finally, this generates an identical key that is mathematically difficult to reverse for another party that might have been listening in on them.

Table 1. Diffie Hellman based on ECC for key exchange

Bob	Alice
Common agreement on elliptic curve[1, 18, 19], base point (g) (8, 5) and number of points on the curve (n) [19]	
Select private key (K) [3]	Select private Alice (s)[7]
Generate public key as point on the curve = multiple (g, k) = (15, 18)	Generate public key as a point on the curve. = multiple (g, s) =(18, 4)
Exchange the public keys = multiple (g, s), =(18, 4)	Exchange the public keys = multiple (g, k), (15, 18)
Generate shared secret based on public point and the private key = multiple (g, s*k), =(1,18)	Generate shared secret based on public point and the private key = multiple (g, k*s), = (1, 18)

7. PROPOSED SYSTEM

The proposed trusted computing system model describes the working procedure of cloud client and working procedure of cloud admin. The scenario started with a demand from clients to modify or upload data in the cloud environment. The cloud requests from the clients to create accounts. The clients started with filling in a form to get an approval from the cloud to host the clients 'data. During the creation of account, they are trying to agree upon a shared key. This key used as an identifier for client during navigation and host of the data in the cloud.

The client wants to send a message to cloud. The client hashes the message. It generates a message digest in a form of integer. The client concatenates the message with integer digest. This message is encrypted by the client. The encrypted message send to the cloud. The cloud get the encrypted message and identify the client based on the shared key. The cloud decrypts the message. The decrypted message is verified by generating the signature. At that stage, the data is available for update, modify, insert or delete.

The proposed model established a connection between client and cloud environment. The connection establishment used Diffie Hellman key based on Elliptic Curve Cryptography Algorithm. This algorithm used to generate a shared secret key between both of them. The shared secret key is used to provide a secure message integrity and message authentication, along with nonrepudiation of message and data confidentiality. It occurs during the account creation. It represents one time password. The shared secret key generates each login time. To generate the shared secret key, the cloud

and the client must agree on a set of parameters ones. These parameters named the elliptic curve equation, the parameter values of the curve (a, b), the field number (a big prime number q), order of the curve, and a base point named (g).

In each login time, the client generates a private key named($user_{ID}$). This key is used to generate a public client key based on the elliptic curve called($user_{pup}$), where $user_{pub} = user_{ID} * g$. On the other side, the cloud generates a private key named ($cloud_{ID}$). This key is used to generate a public cloud key named $Cloud_{pub}$, $Cloud_{pub} = cloud_{ID} * g$.

The client used $cloud_{pup}$ along with his private $user_{ID}$ to generate the shared secret key named $S_{key} = user_{ID} * .Cloud_{pub}$. At a synchronize time the cloud generates same shared secret key $S_{key} = cloud_{ID*} user_{pub}$. This key used to identify and to authenticate the client in the cloud environment. It represents a key for resource accessing on the cloud.

Pseudo code for Diffie Hellman based on elliptic curve in cloud side:

Step 1: common agreements between client and cloud provider.

(Elliptic curve, base point named g, a big prime number named q and finally the order of the curve named N)

Step 2

In cloud side:-

Generate a private key named $\boldsymbol{cloud_{ID}}$ as an integer. Where 0 < $\boldsymbol{cloud_{ID}}$ < N,

Compute $\boldsymbol{Cloud_{pub}} = \boldsymbol{cloud_{ID}} * \boldsymbol{g}$

In client side:-

Generate a private key named as an integer where 0< $\boldsymbol{user_{ID}}$ <N

Compute $\boldsymbol{user_{pub}} = \boldsymbol{user_{ID}} * \boldsymbol{g}$

Step 3: Exchange public key between cloud and client.

Step 4: generate shared key for both client and cloud.

$\boldsymbol{S_{key}} = == \boldsymbol{cloud_{ID}} * \boldsymbol{user_{pub}}$

The client wants to send a message to the cloud. The message should be signed. The client signs the message through calculating a hash function of the message using hash256 algorithm to generate a message digest as an integer $hash_{ID}$. The client generates another secret key named $user_K$ in the range$[1,\ q-1]$. The client uses the elliptic curve and the shared secret key to calculate a point on the curve named $point_s$. The signature will be represented as a two numbers(x, y). These numbers computed using the following equations.

$$point_s = user_K * S_{key} \quad (4)$$

$$x = point_s . x\ (mod\ q), \quad \text{Where } x <> 0 \quad (5)$$

$$y = user_K{}^{-1} * (\ hash_{ID} + x.user_{ID})\ (mod\ q). \quad (6)$$

The signature will be the pair (x, y)

The signature is used to authenticate between application components so that data transmitted only to known parties. For Securing API and Interfaces, the shared secret key used to ensure strong authentication and access controls with encrypted transmission. The client can access the cloud by applying their shared secret key, which is monitor by the cloud environment. They can demand the cloud environment according to their choice and need.

The signed message encrypted using ElGamal cryptosystem based on elliptic curve. The client selects a private integer named $user_{priv}$. This private number used to generate a public key name $user_{pub}$ for encryption process. This public key represents a point on the curve. The encryption based on Elgamal algorithm generates two points on the curve named point c1 and point c2.

$$point_{c1} = user_{priv} * S_{key}\ (mod\ q) \quad (7)$$

$$point_{c2} = user_{priv} * user_{pub}\ (mod\ q). \quad (8)$$

In cloud side, the message received in encrypted format. The cloud decrypts the message. The decryption implemented based on ElGamal algorithm. The decrypted message will go through another iteration to verify the signature of the message. The signature verified through the following steps.

$$w = y^{-1}\ (mod\ q) \quad (9)$$

$$u_1 = w * hash_{ID}\ (mod\ q) \quad (10)$$

$$u_2 = w * x\ (mod\ q) \quad (11)$$

$$p_{new} = S_{key} * u_1 + user_{pub} * u_2 \quad (12)$$

$check\ p_{new_x} == x$
$if\ true\ the\ update, modify, insert\ or\ delet$

$Else, deny\ the\ access.$

8. RESULTS

This paper designed experiments to test the proposed model. Simulation is implemented to compare the execution time of the proposed framework based on two groups. It named ECC group and integer group respectively. The computed execution time was for encryption, decryption, signature and verification based on proposed security model. All experiments were conducted on an Intel core 2.26 GHZ with a 512 KB cache, and 2 GB of RAM. The system was running Microsoft OS "windows 7", and Python 2.7.6 for the mathematical operations, the NumPy plugin for Python was used. Further, all graphs in this paper were also made in Python using the matplotlib plugin. Algorithms used the crypto library of charm cryptography framework. All experimental results represent the mean of 180 trials.

The test data was 360 randomly generated messages. It was 192-bit message size. It was generated using Pythons build in random number generator. Thus, the messages were set from the beginning and throughout all of the experiments. The testing was done by making 180 encryption and decryptions of each message, for each cryptosystem using keys based on each of the security parameters. The key size for integer group was 1024 bit. The tested curve for ECC was Prime192v2 (x9.62 curve over a 192-bit prime field). The security parameters were suggested by Janus D. Nielsen as frequently used message spaces for public-key encryption. Finally, the procedure was repeated 180 times for each cryptosystem. The x-axis represents the number of iterations, and the Y-axis represents the consumed time to do the operation of each algorithm.

Figure 3, 5-show decryption and encryption time of ElGamal cryptosystem based on elliptic curve. The x coordinate describe the number of iteration, the y-axis shows the time in microseconds. Figures 4, 6 show decryption and encryption time of ElGamal cryptosystem based on integer group. Figure 7, 9-show digital signature and verification based on elliptic curve, and figure 8, 10 show the digital signature and verification based on integer group. Figure 11 sows a histogram representation of digital signature and encryption/decryption time based on integer group and ECC group respectively. Table 2 shows the results of the experiments based on the applied parameters, the third column represents the mean execution time for each algorithm, and the fourth column represents the key size for each cryptosystem.

Table 2. Numerical results based on applied algorithms

Algorithm		Mean in (MS)	Keys size
ElGamal based on ECC group	Enc.	8844	160
	Dec.	8836	160
ElGamal based on Integer group	Enc.	9050	1024
	Dec	9177	1024
ECDSA	Sign	1836	160
	Ver.	1827	160
DSA	Sign	6847	1024
	Ver.	6794	1024

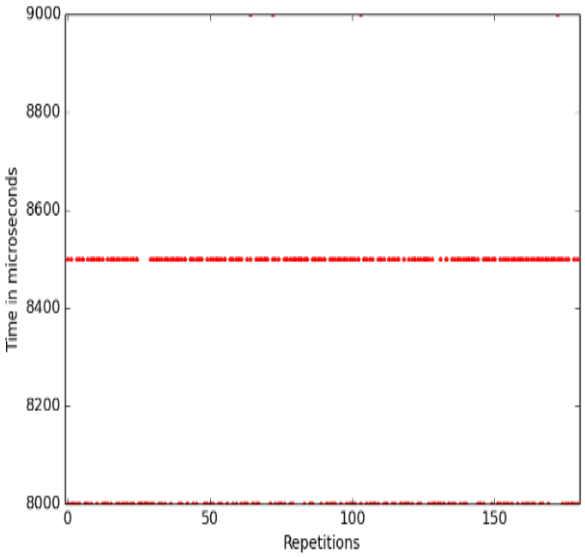

Fig 3: Decryption based on ECC group

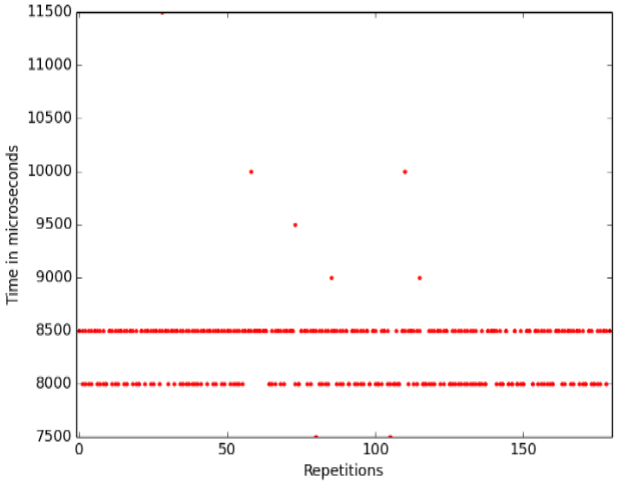

Fig 4: decryption based integer group

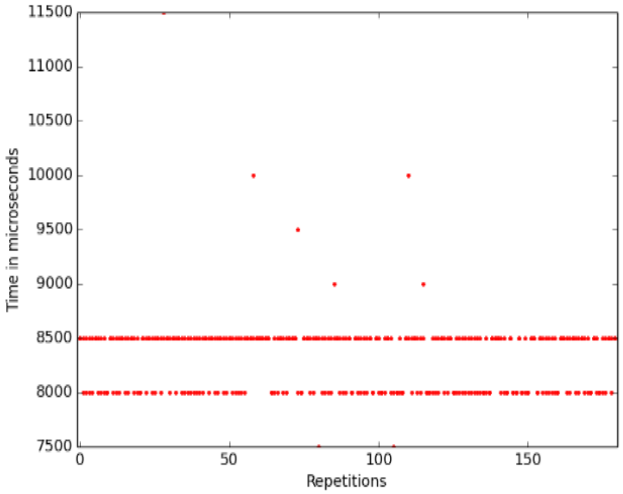

Fig 5: Encryption based on ECC group

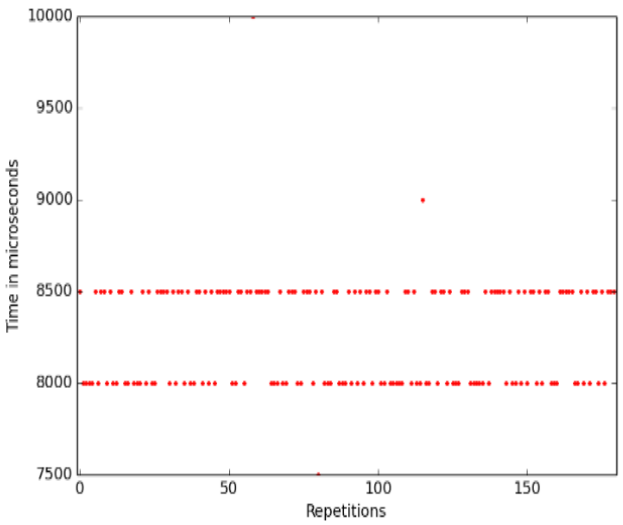

Fig 6: Encryption based integer group

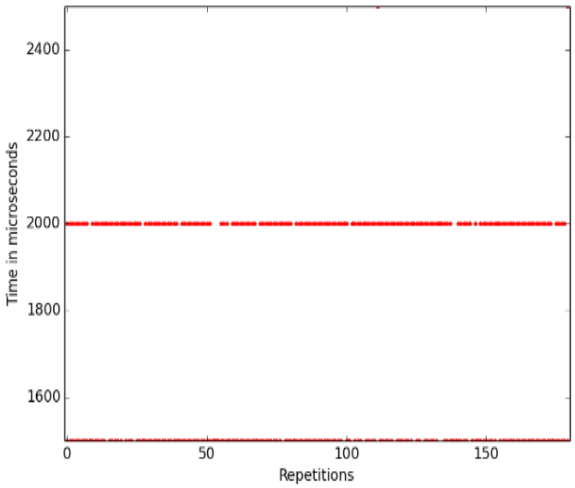

Fig 7: DSA based on ECC group

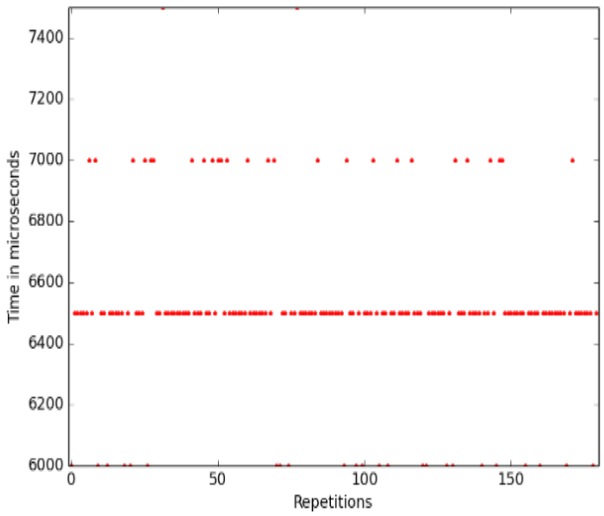

Fig 8: DSA based Integer Group

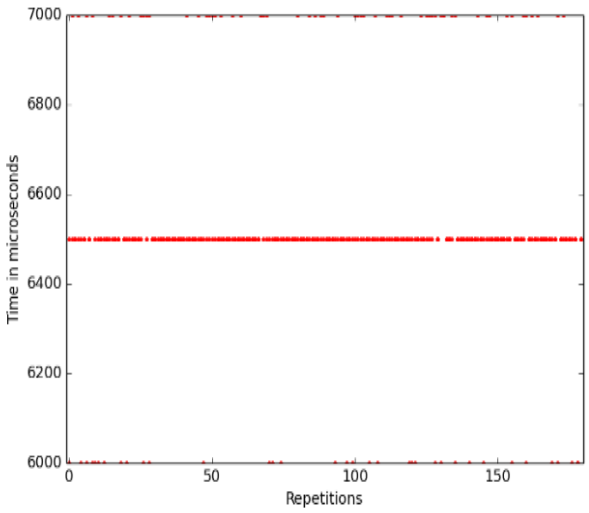

Fig 9: DSA verification based on Integer group

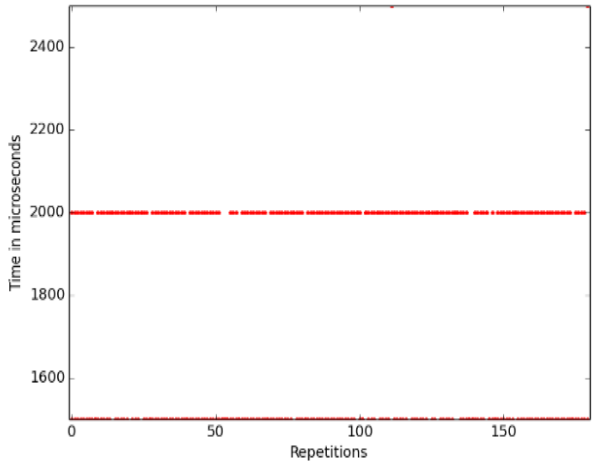

Fig 10: DSA verification based ECC Group

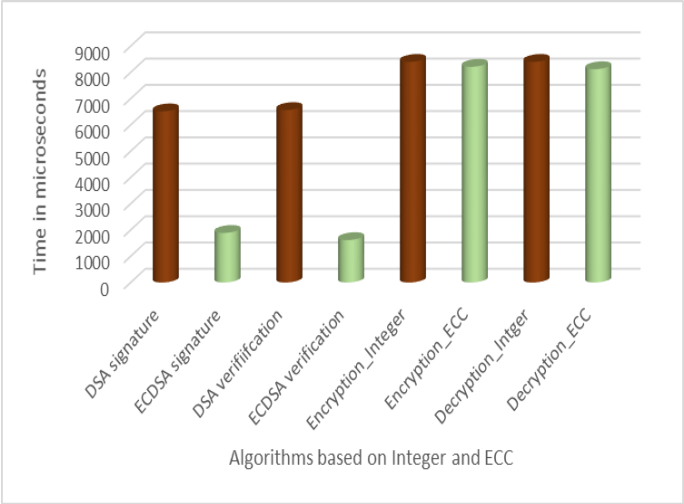

Fig 11: Comparison between algorithms based on integer and ECC.

9. CONCLUSION AND FUTURE SCOPE

Cloud computing as a business model has great potential to change IT industry. More clients and organizations will go to cloud for daily computer use as it abstracts the complexity of computing. Security is main concern, which prevents large organization for using cloud. A service provider needs to ensure that applications are safe from all possible attacks. The client, on the other hand needs to ensure that data is safe from intruders. Number of client will increase which would lead to increase in number of keys held at given point of time. Data security can be very good assured by use of linear cryptographic algorithms put the massive amount of data in cloud resents a hindrance to the idea. Therefore, the proposed system has the advantage of linear cryptography and exponential cryptography for encrypting/decrypting the data.

The average time for encryption/ decryption seems almost similar in each repetition. This implies that the message values does not influence the encryption/decryption time and thus that timing attacks using encryption timing is of no use. Therefore, the content of the message does not affect encryption/decryption time.

It was noticed that ElGamal Cryptosystem based on elliptic curve seems to offer better performance compared to ElGamal based on integer group, regarding encryption and decryption

speeds. ElGamal Cryptosystem based on elliptic curve is good option for clients connect to cloud based application with small session time. As the growth in computing power happens, the requirement of strong key size will also grow. Cloud based application uses lot of thin and dumb client which has very less battery power, they might not be able handle such huge computations. In such scenario, ElGamal based on elliptic curve will come more useful.

The results show that ECDSA outperforms DSA in both signature and verifications. ECSDA provides usable authentication to access services and data for both enterprises and clients. ECSDA is fast, less memory and key size is less compared to DSA based on integer group. Signature generation and verification time are almost same for ECDSA.

Cryptography based on ECC provided a robust and secured model for development and development of secured cloud applications. In future, it can use different key size along with elliptic curve with different parameter. Different encryption algorithm can be used. Client side tool using elliptical curve cryptography for thin and dump devices can be introduced. A secure random number cryptographically should be included while generating private keys in digital signature.

10. REFERENCES

[1] Alowolodu O.D, Alese B.K, Adetunmbi A.O., Adewale O.S, Ogundele O.S., " Elliptic Curve Cryptography for Securing Cloud Computing Applications", International Journal of Computer Applications (0975 – 8887), Volume 66– No.23, March 2013.

[2] Vipul Patel, Rahul Kumar, and Ankur Raj, "Improving Security and Integrity of Data Storage in Cloud Computing By Using Homomorphic Authentication technique", International Journal of Innovations in Engineering and Technology (IJIET), Vol. 3 Issue 2 December 2013

[3] R.Balaji, N. Karthick Gowtham, "Elliptic Curve Cryptography in Cloud Architectures With Lower Latency", Proceedings of International Conference On Global Innovations In Computing Technology (ICGICT'14), 2014

[4] Mohamed Al Morsy, John Grundy and Ingo Müller, "An Analysis of The Cloud Computing Security Problem", In Proceedings of APSEC 2010 Cloud Workshop, Sydney, Australia, 30th Nov 2010

[5] Xiao-Yong l11, Li-Tao Zhou2,Yong Shi1, Yu Guo, "A trusted computing environment model in cloud architecture ", Proceedings of the Ninth International Conference on Machine Learning and Cybernetics, Qingdao, 11-14 July 2010

[6] Neha Tirthani, Ganesan R, " Data Security in Cloud Architecture Based on Diffie Hellman and Elliptical Curve Cryptography", IACR Cryptology, 2014

[7] Aarti P Pimpalkar, Prof. H.A. Hingoliwala, " A Secure Cloud Storage System with Secure Data Forwarding", International Journal of Scientific & Engineering Research, Volume 4, Issue 6, June-2013

[8] Mr. Pragnesh G. Patel, Mr. S.M.Shah, "Data Security in Cloud Computing using Elliptical Curve Cryptography", International Research Journal of Computer Science Engineering and Applications, Vol 2 Issue 7 July 2013

[9] Prof Swarnalata Bollavarapu, Bharat Gupta, "Data Security in Cloud Computing", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 4, Issue 3, March 2014

[10] Ms. Ritu Patidar, 2 Mrs. Rupali Bhartiya, "Implementation of Modified RSA Cryptosystem Based on Offline Storage and Prime Number", IJCAT International Journal of Computing and Technology, Volume 1, Issue 2, March 2014

[11] Abhuday Tripathi, Parul Yadav, "Enhancing Security of Cloud Computing using Elliptic Curve Cryptography", International Journal of Computer Applications (0975 – 8887), Volume 57– No.1, November 2012

[12] Nagesh M.Wankhade, Kiran A. Sahare, Prof. Vaishali G. Bhujade, "SECURE CLOUD SIMULATION USING TRIPLE DES ", International Journal of Research in Advent Technology, Volume 2, Issue 1, January 2014

[13] S.Sathish, D.Sumathi, P.Sivaprakash, " Security Services using ECDSA in Cloud Computing", Volume 4, Issue 5, May 2014

[14] Martin Leslie. Elliptic curve cryptography. (An ECC research project), 2006.

[15] Padma Bh, D.Chandravathi , P.Prapoorna Roja, "Encoding And Decoding of a Message in the Implementation of Elliptic Curve Cryptography using Koblitz's Method ", International Journal on Computer Science and Engineering Vol. 02, No. 05, 2010, 1904-1907

[16] Veerraju Gampala, Srilakshmi Inuganti, Satish Muppidi, "Data Security in Cloud Computing with Elliptic Curve Cryptography", International Journal of Soft Computing and Engineering (IJSCE), Volume-2, Issue-3, July 2012