Elliptic Curve Cryptography: faster and lighter encryption protocol for cloud computing environment

Jagadish Thiruvayipati.
Department of Electrical Engineering Cleveland State University
Cleveland, Ohio

Abstract— Cloud computing environment is not fully secure. Several promising approaches are introduced to resolve the security issues of cloud computing. However, there are lots of vulnerabilities, which need to be solved. On the other side, the cloud applications are being constantly increased that creates typical situation for the client to keep track of their confidential data stored on the cloud. To secure the cloud- computing environment, we introduced novel elliptic curve cryptography (ECC) based on smaller key encryption method. In ECC security rests on the discrete logarithm problem over the points on the elliptic curves. Elliptic curve discrete logarithmic problem takes full exponential time whereas RSA takes sub- exponential time. By this we can use smaller parameters in ECC. The same level of security afforded by an RSA-based system with a large modulus can be achieved with a much smaller elliptic curve group.

Keywords—Encryption, ECC, Key generation, Cloud computing.

1. INTRODUCTION

Cloud Computing has become a revolutionary trend and globally the organizations prefer to have cloud computing services, keeping in view the kind of significant benefits which the organizations could envisage from the cloud computing environment. There are various aspects of cloud computing like the flexibility in terms of managing the software, infrastructure and the platforms for applications development, Scalability aspects are the key to the success of implementing the cloud computing solutions [1].

It is very essential for the organizations to focus on improving their technical infrastructure facilities and the cloud computing solutions are offering the organizations considerable leverage in terms of facilitating the organizations with the flexibility of enhancing their technical infrastructure according to the need basis and also in terms of providing better accessibility to the organizations in terms of data storage and data systems management. [2]

There are various facets of cloud computing environment, which holds the key to success of an organization. The predominant aspects of cloud computing environment are about using right kind of cloud computing platforms, the data security and the integrated systems development and the adaptation of cloud computing solutions as an effective choice by the organizations. [3]

Significant developments are taking place in the cloud computing environment, majorly in terms of data solutions, Storage solutions, and to the extent that even the ERP solutions are currently offered in the cloud computing environment, which enables the organizations to have seamless integration of their enterprise systems and the real-time data management for the organizations. [4]

Data Security solutions are very important for organizations and in the cloud computing environment if the service providers are not ensuring adequate security solutions, then there could be significant issues for the organizational data. It is imperative that there is certain kind of risk associated on depending upon the third-party services for cloud computing solutions. [5]

Cloud Computing has become an imperative solution for organizations in terms of effective solutions for the technological solutions. There are various facets of cloud computing solutions like the SaaS (Software as a Service), IaaS (Infrastructure as a Service) and PaaS (Platform as a Service), which extends the support to the organization in terms of addressing the challenges of infrastructure management. [6]

With the advent support of the IaaS, the organizations can have the feasibility of scaling up or scaling down the hosting space or the server space requirements, depending upon the need at any given time. This will support the organization in terms of investments on the technical infrastructure solutions. Also by adapting the cloud computing services for the organizational requirements, the scope for flexible access is possible, and the staff of the organization can access the systems network of the organization from any location. [7]

In this research paper, the focus is on how to resolve the security issues faced by organizations and ensuring the integrity of the data. Ensuring the integrity of data involves testing of different encryption techniques and to find the best encryption technique for organizations that can solve the security threats faced by users.

The remainder of the paper is organized as section2 gives the related work, section 3 presents proposed plan and focuses on ECC and RSA key generation process in cloud computing.

II. RELATED WORK

Cloud computing applications are growing very fast as the requirement of data storage servers are increasing. The cloud is a next generation platform that provides dynamic resource pools, virtualization, and high availability. There are many works done for ensuring the security in cloud servers using different type of algorithms such as RSA, AES, DES and ECC. [8] Elliptic curve Cryptography is one of the many encryption techniques used in the organizations for providing secured data for the user. The works related to encryption in cloud computing ended ECC showing up a lot of advantages. [10]

In their research study, the researchers emphasize about various trends that are emerging in the cloud computing environment and have depicted the information on how the offline cloud computing, data synchronization are providing seamless support to the organizations, in terms of effectively managing their technical infrastructure, facilitating their staff in terms of accessing the enterprise applications of the organizations. [11]

One of the closely related work proposed the concept of providing secured cloud application by ECC architecture using Sql server 2005 and JAVA application programming software. They carried out the implementation using Trusted Platform Mobile (TPM), which provides a trust for building computing base [9]. Studies proved that ECC algorithm was one among the best algorithms compared to other algorithms, which provide a higher level of security using less number of bits. But considering the security levels, ECC cannot be able to provide high level security than RSA. [12] However, except a little no successful attacks have been evident on this family of curves due to the design of the elliptic curve. [13]

III. PROPOSED PLAN

*A. **Proposed** ECC protocol for cloud computing:*

Elliptic curve cryptography (ECC) is an approach to public-key cryptography based on the algebraic structure of elliptic curves over finite fields. [14] ECC requires smaller keys compared to non-ECC cryptography to provide equivalent security.

Asymmetric cryptography or public-key cryptography is cryptography in which a pair of keys is used to encrypt and decrypt a message so that it arrives securely. Initially, a network user receives a public and private key pair from a certificate authority.

Elliptical curve cryptography uses elliptic curve with key domain parameters D = (Ls, Ri, Pr, Pu, M(x), I1, I2)

q: prime power, that is q = p or q = 2m, where p is a prime

FR: field representation of the method used for representing field elements ∈ Fq

a, b: field elements, they specify the equation of the elliptic curve E over Fq, $y2 = x3 + ax + b$

G: A base point represented by G= (A1, B1) on E (Fq)

n: Order of point G, that is n is the smallest positive integer such that nG = O

h: cofactor, and is equal to the ratio #E(Fq)/n, where

#E(Fq) is the curve order

B. Key Generation:

Jack's public and private keys are associated with a particular set of elliptic key domain parameters (Ls, Ri, Pr, Pu, M(x), I1, I2). Mike generates the public and private keys as follows

1. Select a random number d, d ∈ [1, n – 1]

2. Compare Pu = dG.

3. Jack's public key is Pu and private key is Pr.

It should be noted that the public key generated needs to be validated to ensure that it satisfies the arithmetic requirement of elliptic curve public key. A public key Q = (A0, B0) associated with the domain parameters (Ls, Ri, Pr, Pu, M(x), I1, I2) is validated using the following procedure

1. Check that Pu ≠ O

2. Check that A0 and B0 are properly represented elements of Fq

3. Check if Q lies on the elliptic curve defined by a and b.

4. Check that nQ = O

Algorithm 1: Key Generation

Step 1: Initialization (Ls: local storage, f: file, Ri: random integer, Pr: private key (jack's key), I: interval, I1: prime integer 1, I2: prime integer 2, Delta h: hash, M(x): message, d: divisor, Pu: public key, Pc: a point on curve)
Step 2: Input (f, Pr)
Step 3: Output (M(x))
Step 4: set f subset of
Ls
Step 5: select Ri belongs to I[1, n-1]
Step 6: compute RiPr = (A1, B1) and I1 = A1
mod n
// A1 is considered as integer between 1 and d-1
Step 7: If I=0 then
Step 8: set f subset of Ls
Step 9: compute Ri-1 mod n;
Step 10: compute I2= (1/Ri)(delta h(M(x))+I1P2r)
Step 11: end if
Step 12: if I2=0 then
Step 13: set f subset of Ls
Step 14: compute M(x) = (I1,I2)
Step 15: end if .

Key generation is done in the algorithm1 was using the process of ECC. After initialization with domain parameters of Ri, Pr, I1, I2 and message M(x). By taking f in the range of local storage and selecting Ri in the range of 1 and n-1 calculate the product of random integer and private key of jack as a point (A1, B1) where A1 is considered as an integer between 1 and d-1. For i=0 compute Ri-1 mod n and also calculate prime integer 2 i.e., I2 using the mathematical notation I2= (1/Ri) (delta h(M(x))+I1P2r).
For I2=0 and f in the range of local storage compute message M(x) as (I1, I2). This is how the key was generated i.e., (I1, I2) by using the Algorithm1.

Algorithm 2: Signature Validation Process

Step 1: Initialization
(Ls: local storage, f: file, Ri: random integer, Pr: private key (jack's key), I: interval, I1: prime integer 1, I2: prime integer 2, Delta h: hash, M(x): message, d: divisor, Pu: public key, Pc: a point on curve)

Step 2: Input
Step 3: Output
Step 4: apply Pu belongs to X
Step 5: verify that I1, I2 belongs to [1, n-1]
Step 6: compute (1/I2) mod n and delta h[M(x)]
Step 7: compute w = (1/I2) mod n and delta h[M(x)]
Step 8: compute beta1 = delta h[M(x)]w mod n and beta2 = I1w mod n
Step 9: compute beta1Pc+beta2Pu =(A0, B0);
r = A0 mod n
Step 10: accept if and only if r=I1

This algorithm was used for validation of signature by the cloud for an user. Taking public key as X verify both prime integer 1 and prime integer 2 are in the range from 1 to n-1. In the next step compute (1/I2) mod n and delta h[M(x)].

For calculating the word, we use the notation w = (1/I2) mod n and delta h[M(x)]. After finding the point on elliptic curve, take the sum of beta1 Pc and beta2 Pu as a point (A0, B0). After A0 is generated, compute r = A0 mod n and this generated signature r matches the generated prime integer then the server authorizes the user to read the message or else the sever denies the request by user.

Here, in the implementation, a file is selected from a local storage and goes through the following above process.

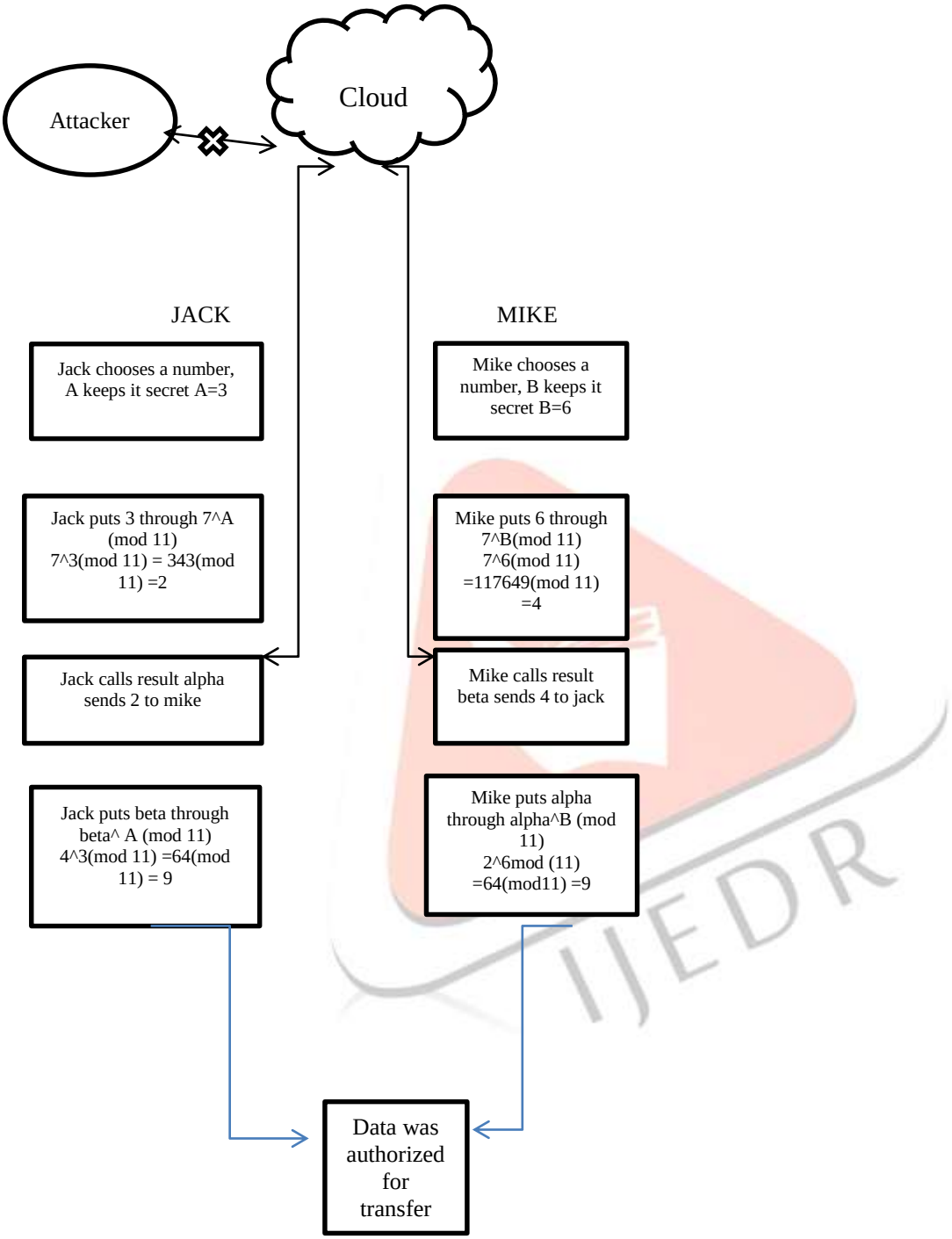

Fig1: secure data transmission between jack and mike in cloud

Jack and Mike are two users in the cloud where Jack wants to send a message to Mike. The message needs to be transferred through cloud. Here we are using Elliptic curve cryptography method for authorizing the data to be transferred from Jack to Mike. In ECC process both the users Jack and Mike needs to choose a private key and will keep it as a secret key. Jack and Mike needs to compute a value using their private key and public key (7 in this case) by the formulae shown in fig1.
The result will be exchanged between the users through cloud using the key sent by jack; mike computes the value using the mathematical equation as shown in fig1. In the same way jack computes his value using the key sent by mike.
An Attacker tries to get the private keys of jack and mike by hacking the cloud server but will not be able to get the final computed key of the users. So jack and mike will be having the unique generated key that is used for transferring the data between them. For retrieving the message sent by jack, mike needs to use the same unique key. If in case mike was not able to use the same computed key the data will not be accessed to mike.
Thus the process of elliptical curve cryptography was able to secure the data transfer between two users through cloud without any possibility of attack by the attacker.

IV. PROCEDURE

Elliptic curves are used to construct the public key cryptography system.

- The private key d is randomly selected from [1, n-1], where n is integer.
- Then the public key Q is computed by dP, where P, Q are points on the elliptic curve.
- Like the conventional cryptosystems, once the key pair (d, Q) is generated, a variety of cryptosystems such as signature, encryption/decryption, key management system can be set up.
- Computing dP is denoted as scalar multiplication.
- It is not only used for the computation of the public key but also for the signature, encryption, and key agreement in the ECC system.

An elliptic curve E over R (real numbers) is defined by following equation

$$E: y^2 + a1xy + a3y = x^3 + a2x^2 + a4x + a6$$

Where a1, a2, a3, a4, a5 ∈K and$\Delta \ddagger 0$. Δ is the discriminant of E and is defined as follows:

$$\Delta = -d2^2 * d8 - 8d4^3 - 27d6^2 + 9d2 * d4 * d6$$

$$d2 = a1^2 + 4a2$$
$$d4 = 2a4 + a1a3$$
$$d6 = a3^2 + 4a6$$
$$d8 = a1^2 * a6 + 4a2 * a6 - a1 * a3 * a4 + a2 * a3^2 - a4^2$$

A. Key pair generation:

Initialization ((P, Q is a point on the curve, Public key is L, Private Key is d)

Randomly select d∈ [1, n-1].
Compute L=dp,

- L is the discrete logarithm of d to the base P
- The main operation is point multiplication
- Multiplication of scalar d * p to achieve another point L
- Point multiplication is achieved by point addition and point doubling.

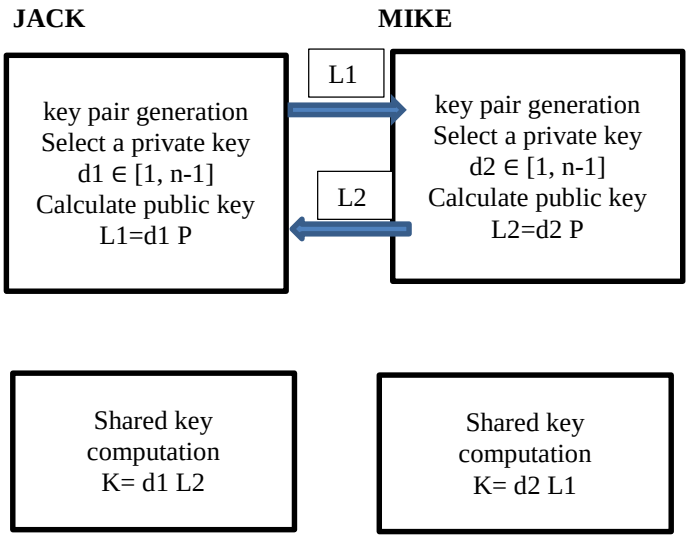

Fig2: key generation

Result key K= d1 L2= d1 d2 P= d2 L1

Jack and Mike are two users in the cloud and will be selecting their own private keys d1 and d2 in the range from 1 to n-1. Using their private keys both the users will be computing their public key using a point on the curve p as L=d*p. The users should exchange their computed key with each other i.e. jack will sent L1 to mike and will receive L2 from mike. Now using the shared keys, both the users will compute a key using their private keys d1,d2 and shared computed keys L1 and L2 as k= d*L. Here if both the final computation keys matches with each other, then the data will be authorized for transfer or else will deny the authorization.

B. Signature validation**:**
For Mike to authenticate Jack's signature, mike must have jack's public key L
Initialization (P1: prime integer1, P2: prime integer2,

1. Verify that P1 and P2 are integers in [1, n − 1]. If not, the signature is invalid
2. Calculate e = HASH (m)
3. Calculate w = P2 −1 (mod n)
4. Calculate u1 = e*w (mod n) & u2 = P1*w (mod n)
5. Calculate (x1, y1) = u1*G + u2*Q
6. The signature is valid if x1 = P1 (mod n)

Thus only if x1 = P1 (mod n), signature will be validated for authorization

Here, we used elliptical curve cryptography protocol for key generation and signature validation as the ECC algorithm which can provide the equal levels of security in comparison with RSA and AES algorithms with less number of bits.
We used java on windows platform and used eclipse software to implement it. We used excel for simulating the graph.

V. SIMULATION AND RESULTS
Here, we compute the bit length of the generated key using ECC and RSA algorithm. The goal of computing the bit length is to select secure and light weight algorithm for secure communication. To confirm the performance efficiency of RSA and ECC, we implemented both algorithms using MATLAB. The complete experimental specification is given in table1.

TABLE1: Experimental Specifications

Parameters	Specifications
Operating system	Windows 8
Processor	Intel® core i3™
RAM	4 Gb
System type	64 bit operating system
Eclipse IDE	Luna service release(4.4.1)
Matlab	Free mat version 4.2

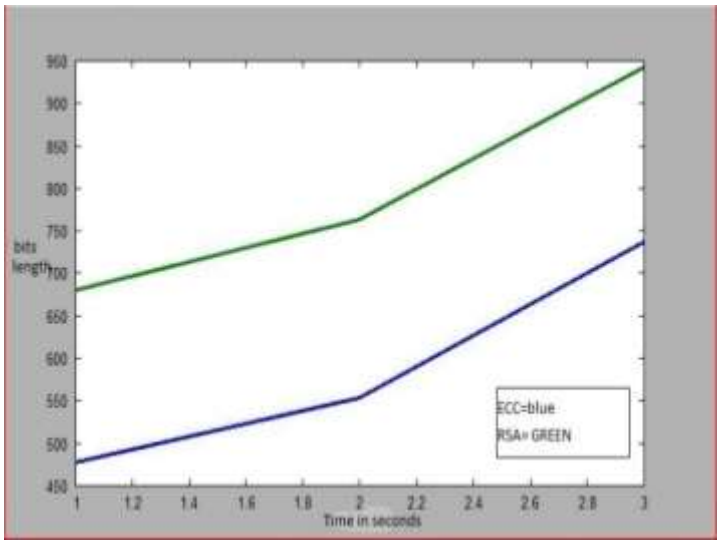

Fig3: bit generation comparison between ECC and RSA

Figure 3 compares the level of security provided by both ECC and RSA with respect to the bit length of generated key. But it is observed that ECC uses less number of bits compared to RSA providing same level of security for the data in cloud.

VI. CONCLUSION

In this research paper we calculated the generated key length of both RSA and ECC algorithms and to compare them to select secure and light weight algorithm for secure communication in cloud. But based on the results, it is proved that ECC uses less number of bits compared to RSA providing same levels of security.

ECC which is elliptic curve cryptography was one of the best approaches for providing high level of security using less number of bits compared to any other algorithms providing high performance and less storage requirements. So considering different factors in both the algorithms, it can be concluded that organizations can prefer ECC compared to other algorithms for securing the users data with high levels of security and with many other benefits.

REFERENCES:

[1] TAROM SA, Bucharest, and S. A. Railway Informatics. "Cloud Computing Technology Trends." (2014).Wang, Xiaoli, Yuping Wang, and Yue Cui. "Energy and locality aware load balancing in cloud computing." *Integrated Computer-Aided Engineering* 20, no. 4 (2013): 361-374.

[2] Bhadauria, Rohit, Rituparna Chaki, Nabendu Chaki, and Sugata Sanyal. "Security Issues In Cloud Computing." *Acta Technica Corviniensis- Bulletin of Engineering* 7, no. 4 (2014): 159.

[3] Somani, Uma, Kanika Lakhani, and Manish Mundra. "Implementing digital signature with RSA encryption algorithm to enhance the Data Security of cloud in Cloud Computing." In *Parallel Distributed and Grid Computing (PDGC), 2010 1st International Conference on*, pp. 211-216. IEEE, 2010.

[4] Yu, Shucheng, Cong Wang, Kui Ren, and Wenjing Lou. "Achieving secure, scalable, and fine-grained data access control in cloud computing." In*INFOCOM, 2010 Proceedings IEEE*, pp. 1-9. Ieee, 2010.

[5] Kandukuri, Balachandra Reddy, V. Ramakrishna Paturi, and Atanu Rakshit. "Cloud security issues." In *Services Computing, 2009. SCC'09. IEEE International Conference on*, pp. 517-520. IEEE, 2009.

[6] Jensen, Meiko, Jörg Schwenk, Nils Gruschka, and Luigi Lo Iacono. "On technical security issues in cloud computing." In *Cloud Computing, 2009. CLOUD'09. IEEE International Conference on*, pp. 109-116. IEEE, 2009.

[7]] Sosa-Sosa, Victor Jesus, and Emigdio M. Hernandez-Ramirez. "A file storage service on a cloud computing environment for digital libraries." *Information Technology and Libraries* 31, no. 4 (2012): 34-45.

[8] Ramgovind, Sumant, Mariki M. Eloff, and Elme Smith. "The management of security in cloud computing." In *Information Security for South Africa (ISSA), 2010*, pp. 1-7. IEEE, 2010.

[9] Singla, Sanjoli, and Jasmeet Singh. "Cloud data security using authentication and encryption technique." *Global Journal of Computer Science and Technology* 13, no. 3 (2013). Gmu7yj

[10] Koblitz, Neal, Alfred Menezes, and Scott Vanstone. "The state of elliptic curve cryptography." In *Towards a quarter-century of public key cryptography*, pp. 103-123. Springer US, 2000.
[11] Lauter, Kristin. "The advantages of elliptic curve cryptography for wireless security." *IEEE Wireless communications* 11, no. 1 (2004): 62-
67.
[12] Alowolodu, O. D., B. K. Alese, A. O. Adetunmbi, O. S. Adewale, and O. S. Ogundele. "Elliptic curve cryptography for securing cloud computing applications." *International Journal of Computer Applications* 66, no. 23 (2013).
[13] Gampala, Veerraju, Srilakshmi Inuganti, and Satish Muppidi. "Data security in cloud computing with elliptic curve cryptography." *International Journal of Soft Computing and Engineering (IJSCE)* 2, no. 3 (2012): 138-141.
[14] Gampala, Veerraju, Srilakshmi Inuganti, and Satish Muppidi. "Data security in cloud computing with elliptic curve cryptography." *International Journal of Soft Computing and Engineering (IJSCE)* 2, no. 3 (2012): 138-1


IJEDR