VOL. 10, NO. 16, SEPTEMBER 2015 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

HYBRID ELLIPTIC CURVE CRYPTOGRAPHY USING ANT COLONY BASED AUTHENTICATION SYSTEM FOR CLOUD COMPUTING

HemanthChakravarthy M. and Kannan E.
Department of Computer Science and Engineering,Vel Tech University, Chennai, India
E-Mail:chekri_5@yahoo.co.in

ABSTRACT

In order to meet the recent requirements and environment, the cloud computing become most important resourcefor both industry and personal usage. Hence, cloud computing is the most rapidly growing technology of the past few years. This rapid growth of cloud computing leads to severe security concerns, because security has a critical issue in the cloud computing, as the user / the provider is the third party and many users are sharing a same cloud. One important design issue in the cloud security framework is space complexity of the security model should be very less in order to meet the mobile customers, because many of the users accessing the cloud by the modern hand held devices. Therefore, smaller sizes of security keys are much preferred for encryption algorithm. Hence, this paper proposes elliptical curve cryptography based security mechanism and ant colony optimization based secured key management technique. The proposed system provides better space complexity than existing RSA and CRT, and the ACO improves optimality.

Keywords: cloud computing, security, RSA algorithm, elliptic curve cryptography, ant colony optimization.

INTRODUCTION

Cloud computing is a kind of high performance computing, which includes distributed computing, grid computing and cloud computing. Grid computing (Abrishami*et al*, 2012) is a paradigm of resource sharing which offers wide and collective distributed computing. For the past few years, the cloud computing is one among top 10 growing technology, which proves a significant impact on IT in the future. Due to this absolute growth of cloud computing, security becomes critical issue. The security of the cloud computing differ from network security, because the user / the provider is the third party to one another and also many users are sharing a same cloud. Also the cloud are accessed by many users through their mobile and handheld devices, therefore the proposed cryptography should occupy lesser memory space. Therefore, smaller sizes of security keys are much preferred for encryption algorithm.

In the recent public key cryptography, factors decomposition problems based on large numbers are commonly used, for example, RSA. With the development of computer hardware and high-performance computing technology, RSA has encountered some difficulties. In the situations, the cryptography based on elliptic curve discrete logarithm problem appears, whose public key is short, network bandwidth is little and ability to resist to attack is strong.

Till date, RSA become most widely used and supported public key encryption scheme. There are new research proposals are in line to improve RSA, such as Chinese Remainder Theorem and Elliptic Curve Cryptography (ECC). ECC is an alternative to traditional public key cryptographic systems. Even though, RSA (called by its inventors, Rivest-Shamir-Adleman) was the most prominent cryptographic scheme, it is being replaced by ECC in many systems (Jarvinen*et al,* 2008). This is due to the fact that ECC gives higher security with shorter bit length than RSA.

In Elliptic curve based algorithms elliptic curve point multiplication is the most computationally intensive operation. Therefore implementing point multiplication using hardware makes ECC (Athavale*et al*, 2009) more attractive for high performance servers and small devices.
RSA is a well-known and most widely used asymmetric algorithm for the past few decades. Recent days, RSA with 2,048-bit key are used in modern computers which is required 8 times higher computations/processing than 1,024-bit RSA keys. Hence, it is not recommended for hand-held products like personal digital assistant and personal communication devices like cellular phones. Further, few researchers improved the performance of RSA using Chinese Remainder Theorem. Even though, it is still a question for those approaching the clouds using their hand held systems in terms of processing time, memory and bandwidth (Bai Qing-Hai*et al,* 2012). Therefore, this paper proposes Hybrid Elliptic Curve Cryptography (HECC).

REVIEW ON CLOUD SECURITY

Implementation of cloud become more rapid in the recent years, such as Wu and Huang *et al* (2011), Mehdi *et al* (2011), Kumar *et al* (2012) and Ponnuramu*et al* (2012).Defining a framework is an initial process of the security model of cloud computing. Zehua Zhang *et al* (2009) proposed Mobile Agent Based Open Cloud Computing Federation (MABOCCF) mechanism, which combines the mobile agent and cloud computing to provide a realization for the open cloud computing federation. MABOCCF offers multiple heterogeneous cloud computing platforms and realizes portability and interoperability. A cloud security management framework is proposed by Almorsy (2011). This framework is based on aligning the FISMA standard to fit with the cloud computing model. This framework is based on improving collaboration between cloud providers and service providers, which defined on top of a number of security standards.

The size of higher bits security key will increase the security of the algorithm but it is expensive in terms of computational requirements. For example, increasing from a 1,024-bit RSA key to a 2,048-bit key requires 8 times of the computations/processing. This is not recommended for hand-held products like personal digital assistant and communication devices, because it simply not having the processing capability to use RSA keys of 3,072 bits and higher (Brohi*et al*, 2014).As early stated, the RSA is most widely used for a long time and it is well understood algorithm. In the other end, the hackers will crack the RSA which used smaller keys. Hence, the size of key is increased from its initial version of 256 bits to 512-bits.

The performance of RSA is improved with CRT. Even it is still a question for those approaching cloud using their hand held systems. Therefore, this paper proposes hybrid ECC. The ECC requires comparatively much less processing time, and provides high security than RSA. For example, a 256-bit ECC key is as secure as a 3,072-bit RSA key. Similarly, the 521-bit ECC keys used in BlackBerry wireless handheld devices are equivalent to RSA keys with 15,000+ bits.

ECC is initially introduced in 1985 by Neal Koblitz from the University of Washington and Victor Miller from IBM. In 2006, Sun Microsystems started to support ECC in its Solaris operating system, and Microsoft followed suit beginning in 2007 with its Vista operating system. ECC is an approach to public-key cryptography based on the algebraic structure of elliptic curves over finite fields. Elliptic curve arithmetic reduces the modular exponentiation operation to multiplication operation within a group. Thus, this scheme aims to extract the promising features of private credentials with the efficiency of ECC (Wang You-Bo *et al*, 2007).

ECs are mathematical NP-hard problems, which are proofed to be intractable in term of complexity (Al-Saidi*et al*, 2011). Cryptography has efficiently utilized the strength EC in developing several cryptosystems such as key agreement protocols, digital signatures and others. Elliptic Curve Cryptography (ECC) usage is with smaller key to give high security and high speed in a low bandwidth. ECC is considered as the best method for upcoming applications. Elliptic curve point multiplication, which is the operation used in every elliptic curve cryptosystem, is hierarchical in nature, and parallelism can be utilized in different hierarchy levels as shown in many publications.

HYBRID ACO BASED ECC SECURITY FRAMEWORK

In the proposed approach, Hybrid ECC is used for encryption and the ACO is used for optimal key management, which represented in Figure-1.

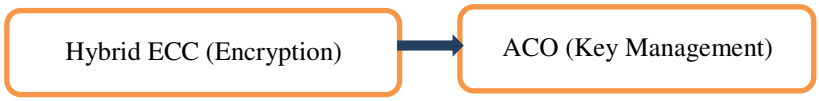

Figure-1. Functionality of proposed works.

Initially the ECC is applied for encryption in the following manner.

- The field of domain in ECC is defined by ‘p’ in the prime case and the pair of ‘m’ and ‘f’ in the binary case. The elliptic curve is defined by the constants ‘a’ and ‘b’ used in its defining equation.
- For current cryptographic purposes, an elliptic curve is a plane curveover a finite field (rather than the real numbers) which consists of the points satisfying the equation (1).

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

along with a distinguished point at infinity, denoted ∞.

- The cyclic subgroup is defined by its generatorG. Several discrete logarithm-based protocols have been adapted to elliptic curves, replacing the group $(Z_p)^x$with an elliptic curve, which may be any one of the following five methods:

  a) the elliptic curve Diffie-Hellman (ECDH) key agreement scheme is based on the Diffie-Hellman scheme,
  b) the Elliptic Curve Integrated Encryption Scheme (ECIES), also known as Elliptic Curve Augmented Encryption Scheme or simply the Elliptic Curve Encryption Scheme,
  c) the Elliptic Curve Digital Signature Algorithm (ECDSA) is based on the Digital Signature Algorithm,
  d) theECMQV key agreement scheme is based on the MQV key agreement scheme.
  e) TheECQV implicit certificate scheme.

- For cryptographic application the order of G, that is the smallest non-negative number n such that nG=∞, which is normally prime. Since n is the size of a subgroup of $E(F_p)$ it follows from Lagrange's theoremthat the number h is an integer. The h is expressed in equation (2).

$$h = \frac{|E(F_p)|}{n} \quad (2)$$

where, h is called the cofactor, must be small (h≤4) and, preferably, h=1. Let us summarize: in the prime case the domain parameters are (p, a, b, G, n, h) and in the binary case they are (m, f, a, b, G, n, h).

- select a random curve and use a general point-counting algorithm, for example, Schoof's algorithm or Schoof–Elkies–Atkinalgorithm,select a random curve from a family which allows easy calculation of the number of points (e.g., Koblitz curves), orselect the number of points and generate a curve with this number of points using complex multiplication technique. This can be contrasted with finite-field cryptography (e.g., DSA) which requires 3072-bit public keys and 256-bit private keys, and integer factorization cryptography (e.g., RSA) which requires a 3072-bit value of n, from which the private key should as large but the public key may be smaller to accommodate efficient encryption, especially where smaller processors are concerned.

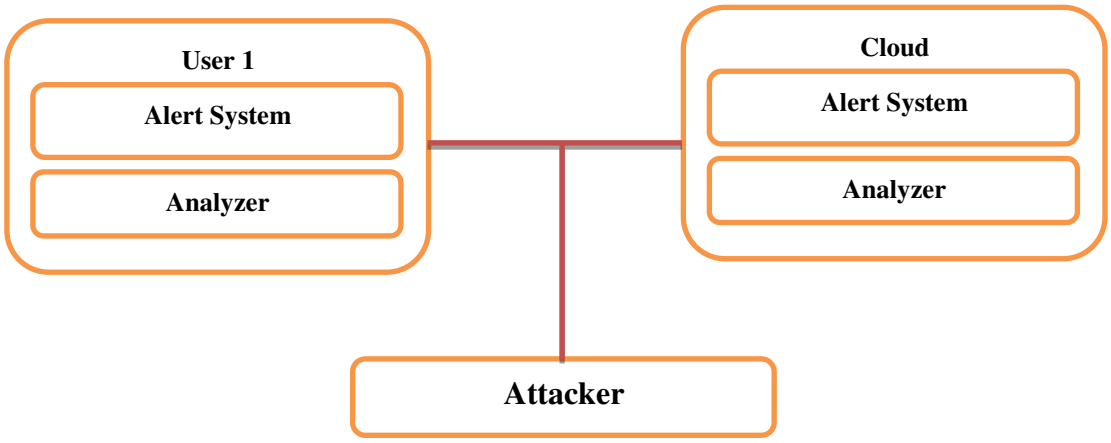

Figure-2. System design of proposed mobile agent model.

After this ECC encryption, ACO is used as Identification Agent (IA) and Target Agent (TA). In the initialization of network phase, ACO flooded in the network as IA to identify all authenticated members in order to process handshake. In the later stage, the ACO is used as TA for authenticating member and preventing non-member.

Hence, there are four components in the proposed system:

- Member: A member is an entity who belongs to the group. U ∈ G means that U belongs to the group G.
- Non-member: A non-member is an entity who does not belong to the group. U∉ G means that U does not belong to the group G.
- ACO-IA is responsible for adding users into his group.

- ACO-TA is responsible for revealing users as well as checking whether handshake players belong to his own group.

The implementation of this attractive scenario is explained hereunder:

a. **Setup:** The common parameter generation algorithm. Given a security parameter k, Setup outputs the public parameters (param) that are common to all groups.
b. **KeyGen:** The group public/secret key generation algorithm. KeyGen is run by ACO-IA and ACO-TA. Given param, KeyGen outputs agroup public key gpk, a secret key of ACO-IAisk and a secret key of ACO-TAtsk.
c. **Add:** The member addition algorithm. Add is executed by a non-member A and ACO-IA. Given param, gpk and isk, Add outputs a membership certificate (certA), a secret key (skA), and ID of A (IDA).
d. **Handshake:** The authentication protocol executed between two players A and B, based on the public input param. The group public keys (gpkA and gpkB), certificates (certA, certB) and secret keys (skA, skB) of A and B are input to Handshake. The output of the algorithm is either rej or acc. A Handshake ←→ B means the situation in which A and B executes Handshake.
e. **Group Trace:** A handshake player's group trace algorithm. Given gpk, tsk and a transcript TA, B, Group Trace outputs yes if A, B ∈ G; otherwise, Group Trace outputs no.
f. **Request Reveal:** The handshake player tracing algorithm. Given gpk, tsk, certA, skA, a transcript TA, B and internal information that are used in Handshake by a player A, Request Reveal outputs the member B.

The proposed mobile agent based secured model is shown in Figure-2. In each node, two types of systems are defined, such that Alert system and analyzer. The analyzer consists of mobile agent which is defined and used as program model to collect information regarding security information. The analyzer receives the security key and verifies the authentication. The alert system broadcast the alert messages to the authenticated neighbours when it identifies the intruder. This alert message also used for verification if the identified attacker may be authenticated user of other authenticated nodes of the concern node.

When an authenticated node of a group receives the message from unknown node, it initiates the mobile agent to collect security information of the unknown node. The MD5 hash function H is used to create message digest H(M) in the authenticated node. The authenticated node generates the following digital signature, if the unknown node is an authenticated node of the group.

$$d_{sign}=(H(M))^{d} \bmod n \quad (3)$$

The authenticated node is encrypting message by using its digital signature. Encrypting the message digest H(M) with its private key d where, $n = p * q$, p and q are random prime numbers with $p \neq q$. The source node forwards d_{sign} with data M, (d_{sign}, M) to its neighbouring node through the path it takes to reach sink.

A neighbouring node on reception of (d_{sign}, M) and the path in the data packet, verifies the digital signature by comparing decrypted value of $d^{e}_{sign} \bmod n$ with message digest H(M).The $d^{e}_{sign} \bmod n$ is key (e, n) using the formula, decrypted using sender's public

$$d^{e}_{sign} \bmod n = ((H(M))^{d} \bmod n)^{e} \bmod n \quad (4)$$

$$= (H(M))^{ea} \bmod n \quad (5)$$

By applying Little Fermat's Theorem to above Equation, it can be shown that

$$d^{e}_{sign} \bmod n = H(M) \quad (6)$$

If the generated H(M) by the receiver and the decrypted H(M) of digital signature d^{sign} is equal, then the receiver accepts the data; otherwise rejects the data and informs the sender that the data is altered through by generating route error packet.

This process is repeated in every hop of the node disjoint path between source and destination. The proposed public key crypto system provides authentication, integrity and non-repudiation in the ad hoc network.

RESULTS AND DISCUSSION

The proposed work is implemented using Network Simulator 2 (NS2), NS2 is the well-known discrete event networking simulation tool. The simulation parameter is shown in Table-1.

The performance of RSA, ECC and the proposed HECC are compared in terms of its execution time, memory requirements in bytes and energy consumption. Figure-3 shown the comparison of execution time and Figure-4 represents the performance comparison of memory requirements.Figure-4 represents the performance energy consumption.

Table-1. Simulation parameters.

Parameters	Values
Simulation area	200 × 200 m2
Propagation	Two ray ground
MAC type	802.11
Antenna	Omni Antenna
Queue	Drop Tail/Priority
Queue Limit	50
No of Nodes	10 to 500
Packet Type	CBR
Packet Size	220 Bits

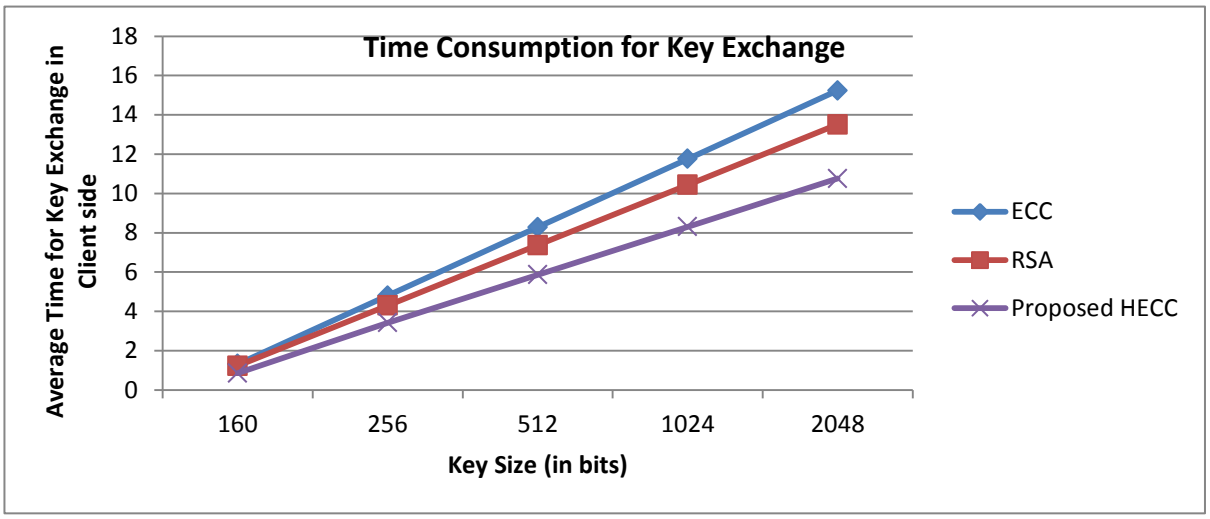

Figure-3. Performance comparison of execution time.

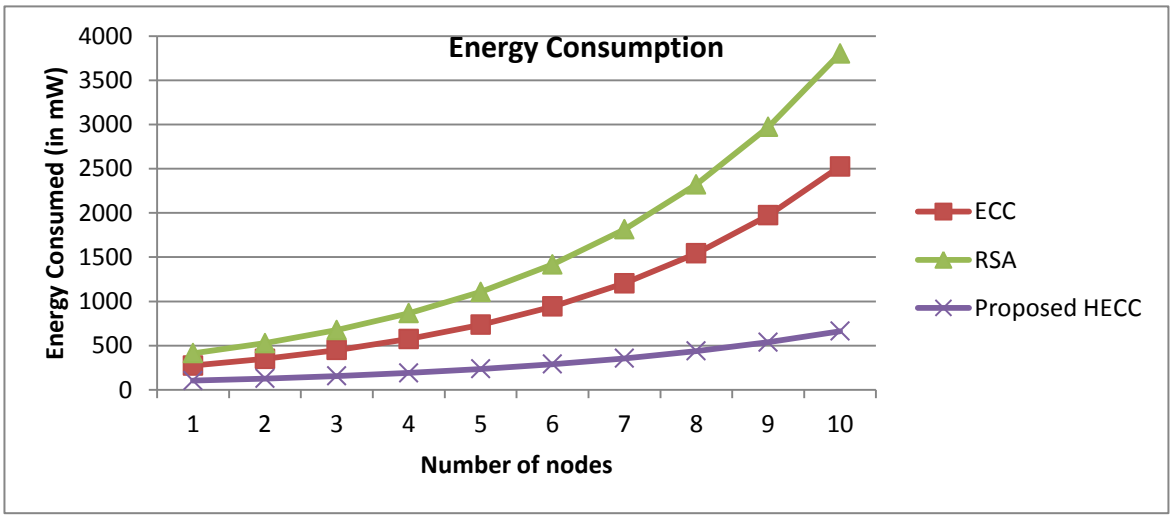

Figure-4. Performance comparisons of memory requirements.

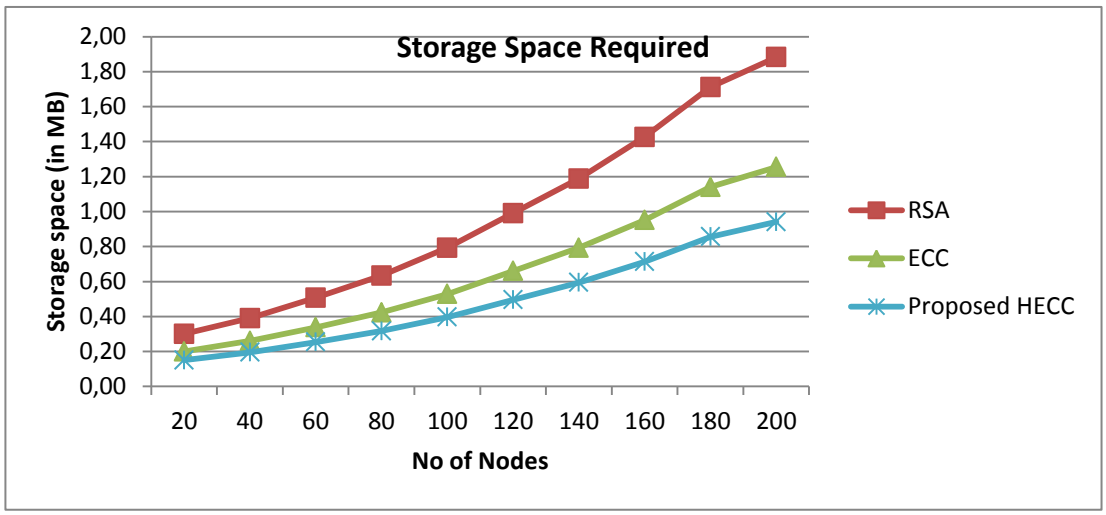

Figure-5. Performance comparisons of memory requirements.

CONCLUSIONS

From the results, it is observed that the performance of proposed HECC having optimal results in terms of key exchange time, space complexity and energy consumption. The security of HECC is achieved even 160 bytes which is equivalent to RSA 1024 bytes. Hence, the comparison of performance of HECC on 160, 192 and 224 bytes are carried out. Similarly the comparison of RSA and ECC are compared on 1024, 2048 bytes. On 160 bytes key size, the HECC has very lesser execution time as well as very lesser memory requirements. In this lesser key size, the proposed HECC provides optimal security, hence it is concluded that the proposed HECC optimal than RSA and RSA-CRT. The hand held devices approaching cloud will optimally use the proposed HECC for smooth, faster and secured functionalities.

REFERENCES

[1] Abrishami S., Naghibzadeh M. and Epema D. 2012. Cost-driven scheduling of grid workflows using partial critical paths. IEEE Transactions on Parallel and Distributed Systems. 23(8): 1400-1414.

[2] Almorsy M.; Grundy John; Ibrahim A.S. 2011. Collaboration-Based Cloud Computing Security Management Framework. IEEE International Conference on Cloud Computing (CLOUD). pp. 364-371.

[3] AL-Saidi N.M.G., M.R.M. Said and A.M. Ahmed. 2011. Efficiency analysis for public key systems based on fractal functions. J. Comput. Sci. 7: 526-532.

[4]Athavale A.; Singh K.; Sood S. 2009. Design of a Private Credentials Scheme Based on Elliptic Curve Cryptography. First International Conference on Computational Intelligence, Communication Systems and Networks, CICSYN '09. pp. 332-335.

[5]Bai Qing-hai; Zhang Wen-bo; Jiang Peng; Lu Xu. 2012. Research on Design Principles of Elliptic Curve Public Key Cryptography and Its Implementation. International Conference on Computer Science and Service System (CSSS). pp. 1224-1227.

[6] BrohiS.N.. M.A. Bamiah, S. Chuprat and J.L.A. Manan. 2014. Design and implementation of a privacy preserved off-premises cloud storage. J. Comput. Sci. 10: 210-223.

[7]Jarvinen K., Skytta J. 2008. On Parallelization of High-Speed Processors for Elliptic Curve Cryptography. IEEE Transactions on Very Large Scale Integration (VLSI) Systems. 16(9): 1162-1175.

[8] Kumar S.K.S. and P. Balasubramanie. 2012. Dynamic scheduling for cloud reliability using transportation problem. J. Comput. Sci. 8: 1615-1626, DOI: 10.3844/jcssp.2012.1615.1626.

[9] Mehdi N.A., A. Mamat, H. Ibrahim and S.K. Subramaniam. 2011. Impatient task mapping in elastic cloud using genetic algorithm. J. Comput. Sci. 7: 877-883, DOI: 10.3844/jcssp.2011.877.883.

[10] Ponnuramu V. and L. Tamilselvan. 2012. Data integrity proof and secure computation in cloud

computing. J. Comput. Sci. 8: 1987-1995. DOI: 10.3844/jcssp.2012.1987.1995.

[11] Wang You-Bo, Dong Xiang-Jun, TianZhi-Guang. 2007. FPGA Based Design of Elliptic Curve Cryptography Coprocessor. Third International Conference on Natural Computation (ICNC 2007). pp. 185-189.

[12] Wu, C.F. and L.P. Huang. 2011. Developing the environment of information technology education using cloud computing infrastructure. Am. J. Applied Sci. 8: 864-871, DOI: 10.3844/ajassp.2011.864.871.

[13] Zehua Zhang; Xuejie Zhang. 2009. Realization of open cloud computing federation based on mobile agent. IEEE International Conference on Intelligent Computing and Intelligent Systems. 3: 642- 646.

[14] HemanthChakravarthy M. and E. Kannan. 2014. A review on secured cloud computing environment. J. Comput. Sci. 11(8): 1224-1228.

[15] HemanthChakravarthy M. and E. Kannan. 2014. Hybrid Elliptic Curve Cryptography for Secured Cloud. International Journal of Applied Engineering Research. 9(24): 29329-29337.