International Electronic Journal of Pure and Applied Mathematics

Volume 8 No. 2 2014, 41-52
ISSN: 1314-0744
url: http://www.e.ijpam.eu
doi: http://dx.doi.org/10.12732/iejpam.v8i2.1

A FRAMEWORK FOR SECURE DATA EXCHANGE IN MOBILE CLOUD COMPUTING

Ardi Benusi[1] §, Dolantina Hyka[2]
[1,2]Faculty of Natural Sciences
University of Tirana
ALBANIA

Abstract: Cloud computing optimizes usage of IT resources such as CPU, storage and network. Network access technologies like 2G, 3G, Wi-Fi, Wi-Max etc. have made possible emerging of a new derivate of Cloud Computing known as Mobile Cloud Computing (MCC) where the related processing of data happens in the cloud and can be accessed through mobile devices. In this paper are addressed issues of a secured data service for file exchange between two end users, one the owner of the file and other the authorized receiver. The cloud mobile is used for storing encrypted files without disclosing any information to the cloud provider. The proxy re-encryption and identity based encryption schemes are used based on bilinear mapping. The security scheme is considered strong based on the hardness of Decisional Bilinear Di-ffie-Hellman assumption and its related inversion.

AMS Subject Classification: 97R30
Key Words: cloud computing, mobile cloud computing, cryptography, proxy Re-encryption, bilinear maps, algorithmic, identity based encryption, Diffie-Hellman, ElGamal cryptosystem.

1. Introduction

Internet has been a driving force towards different technologies that have been developed. One of the most argued and discussed among all of these is Cloud Computing.

Received: April 25, 2014

§Correspondence author

Cloud computing is seen as a trend in the present day scenario with almost all the organizations trying to adapt to it. The advantages of using cloud computing are: i) reduced hardware and maintenance cost, ii) accessibility around the globe, and iii) flexibility and the highly automated process wherein the customer need not worry about software up-gradation which tends to be a daily matter [5, 6]. With the emergence of high-end network access technologies like 2G, 3G, Wi-Fi, Wi-Max etc., a new derivative of cloud computing has emerged. Popularly referred as "Mobile Cloud Computing (MCC)", it has gained popularity among mobile users. Research predicts that the number of mobile cloud computing subscribers is expected to grow from 42.8 million (1.1% of total mobile users) in 2008 to 998 million (19% of total mobile users) in 2014 [3, 10]. MMC can be defined as a composition of mobile technology and cloud computing infrastructure where data and the related processing will happen in the cloud only with an exception that they can be accessed through a mobile device [16].

The growth of mobile cloud computing subscribers is still below expectations according to survey conducted by the International Data Corporation. Most IT Executives and CEOs are not interested in adopting such services due to the risks associated with security and privacy when moving to a cloud service [19]. Once the data has been moved to a cloud provider, control over it has been lost. The user cannot tell where the data resides physically and cannot be fully confident the data is handled with care in a secure manner. Furthermore, when the data has been moved to or created in the cloud, there are concerns about who really owns the data.

According to the different types of services offered, cloud computing can be considered to consist of three layers:

Software as a Service (SaaS):
It is the most popular way of getting into the cloud. This type of Cloud Computing model, delivers applications through a browser using a multiuser architecture. Google Docs, Gmail, Zoho, Google Sites, salesforce.com are some known SaaS examples. SaaS is also commonly used for enterprise resource planning and human resource applications. Access is provided online via e web browser and data is stored in the cloud.

Infrastructure as a Service (IaaS):
This is a capability provided to the consumer by which, it can provision processing, storage, networks and other fundamental computing resources where the consumers can deploy and run the software (i.e. operating systems, applications). Known examples are Amazon S3 storage. IaaS is also used for private networks.

Platform as a Service (PaaS):
PaaS is closely related to SaaS, but delivers a platform from which to work rather than an application to work with. These Service providers offer application programming interfaces (API-s) that enable developers to exploit functionality over the Internet [18].

Mobile devices being battery powered, have limited processing power, low stor-

International Electronic Journal of Pure and Applied Mathematics – IEJPAM, Volume 8, No. 2 (2014)

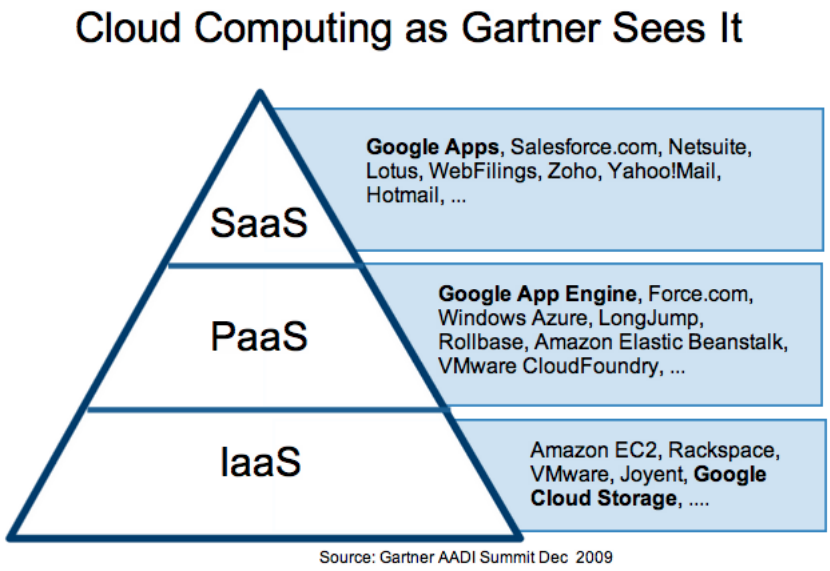

Figure 1: Cloud Computing Service Levels

age, less security, unpredictable Internet connectivity, and less energy. The aforementioned limitations of mobile devices are always obstacles for computationally intensive and storage demanding applications on a mobile. To increase the capability, capacity and battery time of the mobile devices, computationally intensive and storage demanding jobs should be moved to cloud [1, 2].

Figure 2: Mobile Cloud Computing Scenario

International Electronic Journal of Pure and Applied Mathematics – IEJPAM, Volume 8, No. 2 (2014)

2. Identity-Based Encryption

Identity-based Encryption (IBE) takes a breakthrough approach to the problem of encryption key management. IBE can use any arbitrary string as a public key, enabling data to be protected without the need for certificates. Protection is provided by a key server that controls the dynamic generation of private decryption keys that correspond to public identities and the key servers base root key material. By separating authentication and authorization from private key generation through the key server, permissions to generate keys can be controlled dynamically on a granular policy driven basis, facilitating granular control over access to information in real time. The stateless nature of IBE also dramatically simplifies operation and scaling. Key servers can be distributed independently and geographically and key requests load balanced across them without the need to synchronize data, thus enabling high scale without growing complexity and to enable distributed and federated key management across the world easily and quickly.Identity-based encryption (IBE) is a public-key encryption technology that allows a public key to be calculated from an identity, and the corresponding private key to be calculated from the public key. Calculation of both the public and private keys in an IBE-based system can occur as needed, resulting in just-in-time key material. This contrasts with other public-key systems, in which keys are generated randomly and distributed prior to secure communication. The ability to calculate a recipient's public key, in particular, eliminates the need for the sender and receiver in an IBE-based messaging system to interact with each other, either directly or through a proxy such as a directory server, before sending secure messages[11]. Compared with typical public-key cryptography, this greatly reduces the complexity of the encryption process for both users and administrators. An added advantage is that a message recipient doesn't need advance preparation or specialized software to read the communication [4].As simple identifiers for generating public keys, email addresses, mobile phone numbers, IMEI-s are used.In IBE the process of encrypting messages can be initiated by the sender who can calculate the recipient's public key using a unique identifier of the recipient like his/her email address or phone number. A trusted server called the private key generator use a cryptographic algorithm and calculates the corresponding private key from the public key. The recipients generate their own private key directly from the server, without worrying about the distribution of public keys.

The above diagram illustrates how user A would send a message to user B based on identity based encryption.

1. User Alice (A) send an encrypted message to user Bob (B) using his email address as public key.

2. User B needs to authenticate to key-server in order to get his private key. B contacts key-server and server contacts a directory service or any other authenticated service and challenges the user to provide his credentials to verify

International Electronic Journal of Pure and Applied Mathematics – IEJPAM, Volume 8, No. 2 (2014)

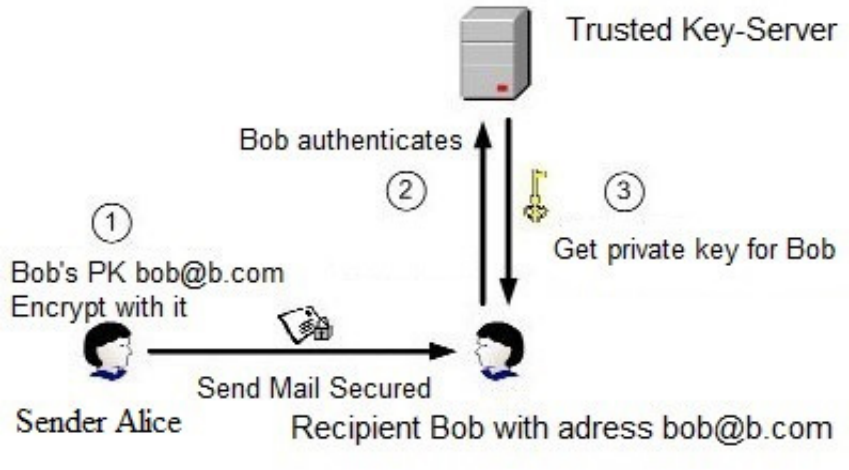

Figure 3: How IBE works

user identity. Also other policy elements may be applied to the conversation. For example recipient is not an active user anymore and the message is bounced back to sender.

3. Key-Server returns to B his private key and the message is decrypted. The key is cached in user's device and future encryption with B's public key can be decrypted even if user is offline. B does not need any special software before A can send him a secured message.

The only information that the key-server stores permanently is a secret master key, a large random number that is only known to the server. Upon the IBE installation software by the user, the master key is used by server to generate a common set of public-key parameters that are given to users installing the software and the recipient's secret keys.

3. Proxy re-encryption

The goal of this newly-devised cryptographic primitive is to securely re-encrypt a ciphertext from one secret key to another without the intervention of a third trusted party. An everyday example is the case of encrypted emails that need to be forwarded from user Alice (A) to user Bob (B). User A receives emails encrypted with her public key and since she possesses the private key, only her can decrypt mails arrived at her inbox. When she lives on vacation, she wants to delegate her job to user B by forwarding every mail to him, but she doesn't want to give him his

International Electronic Journal of Pure and Applied Mathematics – IEJPAM, Volume 8, No. 2 (2014)

secret key. The naive way would be that the mail server decrypts the message for Alice using her private key stored on server and re-encrypt it using Bob's public key and then Bob could decrypt using his own secret key. This naive approach is simple and the whole work is done by mail server which has a copy of the message and a copy of Alice's private key. The mail server can accidentally or purposely read or change the message of Alice and so it cannot be trusted. More efficient approaches are possible as suggested in [15].

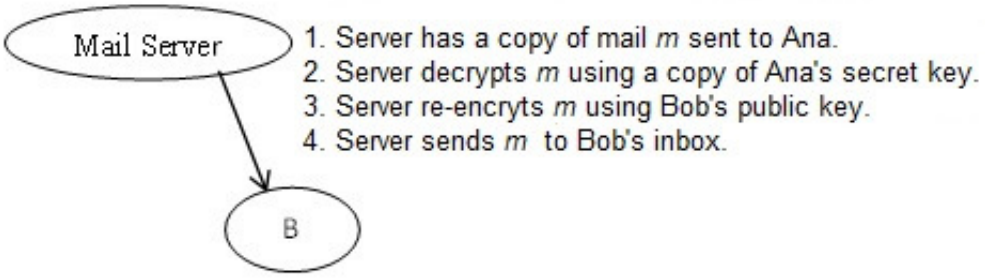

Figure 4: Simple but naive implementation of re-encryption

In 1998 Blaze, Bleumer and Strauss suggested the atomic re-encryption known as BBS approach [7]. The notion of a "re-encryption" key $RE_{A\to B}$ is introduced that allows re-encryption from one secret key to another without ever knowing the plain text. The scheme is based on ElGamal cryptosystem.

* Generation of keys:
  $< g >= \mathbb{G}$ of prime order q.
  $SK_a = a \in \mathbb{Z}_q^*$, $SK_b = b \in \mathbb{Z}_q^*$ where a, b are randomly selected.
  $PK_a = g^a, PK_b = g^b$
  $RE_{A\to B} = b \cdot a^{-1}(mod, q)$

* Encryption:
  Plain message $m \in \mathbb{G}$, random $k \in \mathbb{Z}_q^*$
  Encrypted message for Alice is $C_a = (g^k \cdot m,\ \mathrm{g}^{ak})$

* Decryption:
  $m = \frac{g^k \cdot m}{(g^{ak})^{a^{-1}}}$, private key a is used by Alice to decrypt m

* Re-encryption:
  $C_a = (g^k \cdot m,\ g^{ak})$
  $C^b = (g^k \cdot m,\ (g^{ak})^{RE_{A\to B}}) = (g^k \cdot m,\ (g^{ak})^{b\cdot a^{-1}}) = (g^k \cdot m,\ g^{bk})$, which is the encrypted message using Bob's public key.

International Electronic Journal of Pure and Applied Mathematics – IEJPAM, Volume 8, No. 2 (2014)

At the end of the Re-encryption phase, user A has re-encrypted m using B's public key and now the mail server may forward the encrypted message to Bob so only he can decrypt using his private key b. The ciphertext is secured based on ElGamal cryptosystem. Also by not knowing a and$b$, the server cannot distinguish $RE_{A\to B}$ from any element of $\mathbb{Z}_q$. However there are some things to note:

1. Server may calculate the inverse of $RE_{A\to B}(mod\, q)$, allowing the server to re-encrypt all the messages from Bob to Alice. This may flood Alice's inbox with unwanted messages.

2. If the server conspires with Alice against Bob, it is easy for both the server and Alice to learn Bob's secret key. Likewise Bob and proxy server may collaborate to learn Alice's secret key.

3. By looking at $RE_{A\to B}$, Bob must share his secret key, so both users must rely on a trusted party or compute some security computation with server. No pre-sharing of secret keys is desirable. Only the use of Alice's private key and Bob's public key are required for re-encryption. Bob's private key is not required.

A related attempt to achieve a better re-encryption scheme was proposed by Dodis and Ivan using only standard public key infrastructure, but the system required a pre-sharing phase. Even exchanging key securely is available in cryptography, the feature is not desirable since Alice and Bob may have no prior relationship and bidirectional communication is not possible [14].

4. Improved proxy re-encryption with bilinear maps

Let $\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_3$ be cyclic groups of the same order q

Definition 1. A bilinear map from $\mathbb{G}_1 \times \mathbb{G}_2$ to $\mathbb{G}3$ is a function $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_3$ such that for all $g_1 \in \mathbb{G}_1$, $g_2 \in \mathbb{G}_2$, a, $b \in \mathbb{Z}_q$,

$$e(g_1^a,\, g_2^b) = e(g_1,\, \mathrm{g}_2)^{ab}$$

In the proxy re-encryption scheme $\mathbb{G}_1 = \mathbb{G}_2$, $\mathbb{G}_1 =< g >$, $e : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ is computed efficiently and $< e(g, g) >= \mathbb{G}_2$. The last one makes e non-degenerative [12].
The decisional Diffie-Hellman Problem (DDH) in bilinear map is computationally not difficult since e is easy compute. The cryptosystems using bilinear maps are based on the decisional bilinear Diffie-Hellman (DBDH) problem and its inversion (k- DBDHI).
DBDH Problem: Given $(g, g^a, g^b, g^c (g \in \mathbb{G}_1), U \in \mathbb{G}_2)$, is

$$U = e(g, g)^{abc}?$$

International Electronic Journal of Pure and Applied Mathematics – IEJPAM, Volume 8, No. 2 (2014)

So is it possible to distinguish$e(g,g)^{abc}$ from a random element $U \in \mathbb{G}_2$ of the group?
k- DBDHI Problem: Given $(g, g^t, g^{t^2}, \cdots g^{t^k} (g \in \mathbb{G}_1), U \in \mathbb{G}_2)$, is

$$U = e(g,g)^{t^{-1}}?$$

So is it possible to distinguish $e(g,g)^{t^{-1}}$ from a random element $U \in \mathbb{G}_2$ of the group?
The algorithm for improving proxy re-encryption based on bilinear maps is the following:

* Generation of keys:
  $< g >= \mathbb{G}_1$ of prime order q.
  $SK_a = a \in \mathbb{Z}_q^*$, $SK_b = b \in \mathbb{Z}_q^*$ where a, b are randomly selected.
  $PK_a = g^a, PK_b = g^b$
  $U = e(g,g)$
  $RE_{A\to B} = (g^b)^{a^{-1}} = g^{\frac{b}{a}}$

* Encryption:
  Plain message $m \in \mathbb{G}_2$, random $k \in \mathbb{Z}_q^*$
  Encrypted message for Alice is $C_a = (U^k \cdot m, g^{ak})$

* Decryption: $m = \frac{U^k \cdot m}{e(g^{ak}, g^{a^{-1}})} = \frac{U^k \cdot m}{U^k}$, private key a is used by Alice to decrypt m

* Re-encryption:
  $C_a = (U^k \cdot m,\ \mathrm{g}^{ak})$
  $C_b = (U^k \cdot m, e(g^{ak}, RE_{A\to B})) = (U^k \cdot m, e(g^{ak}, g^{\frac{b}{a}})) = (U^k \cdot m, U^{kb})$
  The decryption now can be made only by Bob, using his sekret key b:
  $m = \frac{U^k \cdot m}{(U^{kb})^{b^{-1}}}$

Some nice features are improved by this algorithm:

1. Alice uses only her private key and Bob's public key to re-encrypt message. No need to pre-share secret keys.

2. The server cannot re-encrypt Bob's messages for Alice, because no extraction of $g^{a \cdot b^{-1}}$ from $g^{b \cdot a^{-1}}$ can be made, assuming the DDH problem is hard.

3. The server cannot read the message it is encrypting for Bob without extracting the secret keys on its own.

4. Even with the collaboration of user A, the server cannot extract secret key b from $RE_{A\to B}$.

5. DBDHI assumption together with ElGamal scheme retain the security of this scheme.

International Electronic Journal of Pure and Applied Mathematics – IEJPAM, Volume 8, No. 2 (2014)

5. Secure exchange of data in the cloud

Using the combination of Identity-Base encryption and proxy Re-encryption on bilinear maps a secure data service can be achieved that not only provides the data privacy, but also a fine access control with minimum cost of updating access policy like in the cases of medical records or documents stored in a cloud mobile platform. In the case of files being stored on the cloud, three components will be considered:

1. File owner (A), is the owner of files that are uploaded to the cloud. File owner might be a company, an individual of an organization or an entire department with policies for granting access privileges to documents.

2. File sharer (B) is the person, company, institution, department to whom is granted the access to download a readable copy of the documents from the cloud storage.

3. Cloud Service is a cloud mobile provider with the main duty to provide storage as a service to the clients requesting documents sharing services. The cloud provider is also responsible for a lot of things like authentication of users to the cloud, managing, operating and allocating the cloud resources efficiently. The mobile cloud can be considered as a trusted third party when generating the master key for identity-base encryption, but a different third provider may be considered. The integrity of the files stored on the mobile cloud is not considered here. A secure scheme may be found at [8], where Reed-Solomon codes together with homomorphic tokens are used in ensuring integrity of the files from any corruption due to server failure and/or any Byzantine failures, during insert, delete or update operations.

In the secure exchange network model a semi-trusted proxy hosted on the cloud transforms the file block encrypted with owner's public key to an encrypted block using sharer's public key. Both identity-based encryption and proxy re-encryption are used in this scheme. Identity of a user is converted to a binary string $ID \in \{0,1\}^*$ and as public identities, email address, phone numbers or any other public information are considered. The bilinear map $e : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ is chosen carefully to be non-degenerative and computed efficiently on cyclic groups $\mathbb{G}_1$ and $\mathbb{G}_2$ with prime order q. Hash function $H : \{0,1\}^* \to \mathbb{G}_1$ is used to convert the identity of a user to an element of $\mathbb{G}_1$. The scheme works as follows:

1. Master Key (MK) and system parameters are selected ($\mathbb{G}_1$, $\mathbb{G}_2$, g, e). MK is owned only by semi-trusted proxy hosted on the cloud and is used to generate secret key for authenticated user on the cloud.

2. Mobile users have to register in the system to obtain a secret key SK based on mobile users' identity using MK and H function. A mobile program may

International Electronic Journal of Pure and Applied Mathematics – IEJPAM, Volume 8, No. 2 (2014)

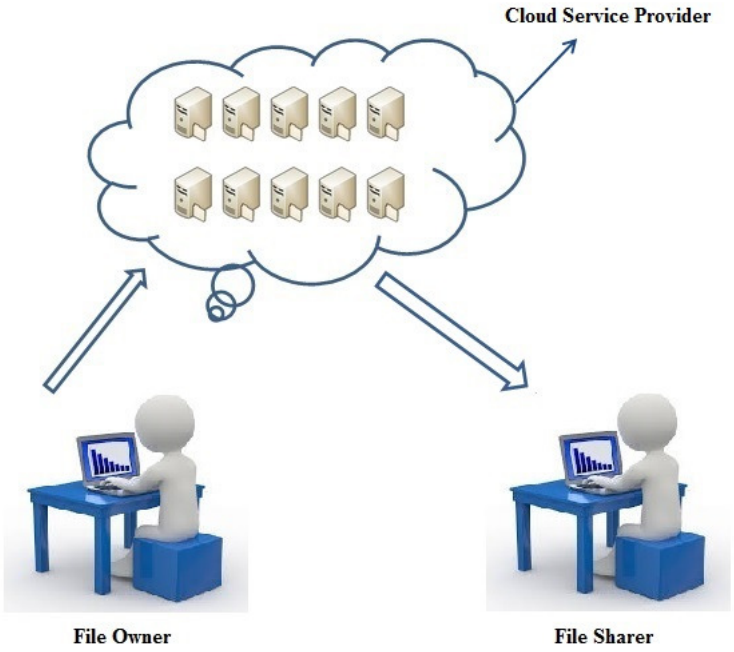

Figure 5: Exchange network model

be downloaded from each user and during installation phase the secret key SK_{ID} is generated. Identity-based encryption scheme is used to generate and exchange private and public keys.

3. Owner of file divides the file in $n$chunks and presents each piece as an element $m_i \in \mathbb{G}_2$. Using owner's identity ID_{owner} as public key, the message is encrypted and then the encrypted piece $EF(m_i)$ is uploaded on the cloud server:
$$EF_{owner}(m_i) = (m_i \cdot e(g,g)^k, g^{k \cdot SK_{IDowner}}),$$
where $k \in \mathbb{Z}_q^*$ is randomly selected.

4. After uploading the file on the cloud, the owner generates the re-encryption keys and using proxy re-encryption based on bilinear maps transforms the block file $EF_{owner}(m_i)$ encrypted with its own public key into a new block file $EF_{sharer}(m_i)$, encrypted with sharer's public key.

5. The sharer queries the cloud to obtain the file re-encrypted and the cloud verifies the sharer's identity. If verification is successful, the cloud sends the re-encrypted file to user and due to ciphertext transformation: sharer can decrypt the data without the owner involvement.

International Electronic Journal of Pure and Applied Mathematics – IEJPAM, Volume 8, No. 2 (2014)

References

[1] B.P. Rimal, E. Choi, I. Lumb, A taxonomy and survey of cloud computing systems, *Fifth International Joint Conference on INC, IMS and IDC*, (2009).

[2] G. Huerta-Canepa, D. Lee, A virtual cloud computing provider for mobile devices, *Proceedings of the 1st ACM Workshop on Mobile Cloud Computing & Services Social Networks and Beyond -MCS '10*, No.6 (2010).

[3] M. Armbrust, A. Fox, R. Griffith, A.D. Joseph, R.H. Katz, A. Konwinski, G. Lee, D.A. Patterson, A. Rabkin, I. Stoica, M. Zaharia, A view of cloud computing, *Communications of the ACM*, **53**, No.4 (2010).

[4] J. Kilian, Advances in Cryptology - CRYPTO 2001, *21st Annual International Cryptology Conference, Santa Barbara, California, USA, Proceedings*, **2139**, (2001), 213-229

[5] Harold C. Lim, Shivnath Babu, Jeffrey S. Chase, Sujay S. Parekh, Automated Control in Cloud Computing: Opportunities and Challenges, *ACDC '09 Proceedings of the 1st workshop on Automated control for datacenters and clouds*, (2009), 13-18.

[6] R. Maggiani, Cloud Computing is Changing How we Communicate, *Professional Communication Conference, 2009. IPCC 2009. IEEE International*, (2009), 1-4.

[7] M. Blaze, G. Bleumer, M. Strauss, Divertible protocols and atomic proxy cryptography, *In Proceedings of Eurocrypt '98*, **1403**, (1998), 127-144.

[8] Cong Wang, Qian Wang, Kui Ren, Wenjing Lou, Ensuring data storage security in Cloud Computing, *Quality of Service, 2009. IWQoS. 17th International Workshop on*, (2009), 1-9.

[9] X. Zhang, Ensure Data Security in Cloud Storage, *International Conference on Network Computing and Information Security*, (2011).

[10] K. Krauter, R. Buyya, M. Maheswaran, A taxonomy and survey of grid resource management systems for distributed computing, *Software: Practice and Experience*, **32**, No.2 (2002), 135-164.

[11] D. Anand, V. Khemchandani, R. K. Sharma, Identity-Based Cryptography Techniques and Applications (A Review) *5th International Conference on Computational Intelligence and Communication Networks*, (2013).

[12] G. Ateniese, K. Fu, M. Green, S. Hohenberger. Improved Proxy Re-encryption Schemes with Applications to Secure Distributed Storage, *ACM Transactions on Information and System Security (TISSEC)*, **9**, No.1 (2006), 1-30.

[13] J. Katz, J.Lindel, *Introduction to Modern Cryptography*, Chapman & Hall/CRC, USA (2008).

[14] A. Ivan, Y.Dodis, Proxy Cryptography Revisited, *Proceedings of the Network and Distributed System Security Symposium, NDSS* , (2003).

[15] D. Nunez, I. Agudo, J. Lopez, Integrating OpenID with proxy re-encryption to enhance privacy in cloud-based identity services, *4th IEEE International Conference on Cloud Computing Technology and Science Proceedings*, (2012).

[16] H. Tianfield, Security Issues in Cloud Computing, *IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, (2012).

[17] C. Paar, *J. PELZ Understanding Cryptography*, Springer, USA (2010).

[18] Chan, L. Mei, Z.Zhang, Modeling and testing of cloud applications, *Services Computing Conference, IEEE Asia-Pacific*, (2009).

[19] W. Liu, Research on cloud computing security problem and strategy, *2nd International Conference on Consumer Electronics, Communications and Networks (CECNet)*, (2012).