Survey On Data Security In Cloud Computing

Mr.Pragnesh G. Patel
C.S.E. Department,
Government Engineering College,
Sector-28, Gandhinagar

Prof. Sanjay M. Shah
C.S.E. Department,
Government Engineering College,
Sector-28, Gandhinagar

Abstract

Cloud Computing is a technique used to unite the power of various resources over network in a more efficient and scalable way to the end user. Cloud computing is one of the rapidly growing field of IT among the many business activities of large organization. It provides resources in the form of services as per usage base model. With the formation of cloud (pool) of various shared resources such as platform, application, infrastructure, storage over the network, it gives user on demand access to the resources on the scalable and dynamic way to avoid large upfront cost of investment. With the advantages, to access the services there is need to share resources and data including crucial information over the network, which lead to the hackers for various security issues. In this paper, we explore cloud computing along with its advantages with various security threats and existing authentication and encryption methods to provide security.

Keywords: Cloud Computing, Security, Encryption, Authentication, Data security in Cloud Computing

I. Introduction

A. What is Cloud Computing

Cloud computing is a paradigm in which tasks are assigned to a combination of connections, software and services accessed over a network. Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications and services) which create cloud. Cloud computing is a disruptive technology that has the potential to enhance collaboration, agility, scaling, and availability, and provides the opportunities for cost reduction through optimized and efficient computing. The cloud model envisages a world where components can be rapidly orchestrated, provisioned, implemented and decommissioned, and scaled up or down to provide an on-demand utility-like model of allocation and consumption[1]. Google Docs is also one of the best example of Cloud computing. We access the document using internet on the PC which does not have any supported software for that document. Cloud computing means create a "cloud" of resources and let user will select them according to their needs or requirement or we can give power of super computer to the user on the usage per model. Usage per model means user will pay as per their use of resources. Cloud Computing is also described as "on-demand computing" because the user can access as per their requirement and demand. Cloud computing can also be defined as it is a new service, which are the collection of technologies and a means of supporting the use of large scale Internet services for the remote applications with good quality of service (QoS) levels[2].

A. Characteristics of cloud computing [3]

Cloud computing exhibit five essential characteristics defied by NIST (National Institute of Standards and Technology)

1) On-demand self-service: A consumer can unilaterally provision computing capabilities.

2) Broad network access: Capabilities are available over the network and accessed through standard mechanisms that promote use by heterogeneous thin or thick client platforms.

IJERT

3) Resource pooling: The provider's computing resources are pooled to serve multiple consumers, with different physical and virtual resources dynamically assigned and reassigned according to consumer demand.

4) Rapid elasticity: Capabilities can be rapidly and elastically provisioned, in some cases automatically, to quickly scale out and rapidly released to quickly scale in.

5) Measured service: Cloud systems automatically control and optimize resources use by leveraging a metering capability at some level of abstraction appropriate to the type of service.

C. Types of Cloud Computing [4]

1) Software as a Service (SaaS): It provides capabilities to use various software applications running on a cloud infrastructure. The software applications are accessible throw client interface like web browser. The best example of this is Google Docs, which you can use for creating and storing text documents, presentations, spreadsheets etc..The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, storage with the possible exception of limited user specific application configuration settings.

2) Platform as a Service (PaaS): It provides capabilities to deploy consumer-created or acquired applications created using programming languages and tools supported by the provider onto the cloud infrastructure. For example, we can create web based application like e-bay on the cloud platform. The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, or storage, but has control over the deployed applications and possibly application hosting environment configurations.

3) Infrastructure as a Service (IaaS): It provides capability to provision processing, storage, networks, and other fundamental computing resources where the consumer is able to deploy and run arbitrary software, which can include operating systems and applications. For example, it provides the infrastructure for hosting the website & pay as per use. It provides various online data storage for storing large data. The consumer does not manage or control the underlying cloud infrastructure but has control over operating systems, storage, deployed applications, and possibly limited control of select networking components (e.g., host firewalls).

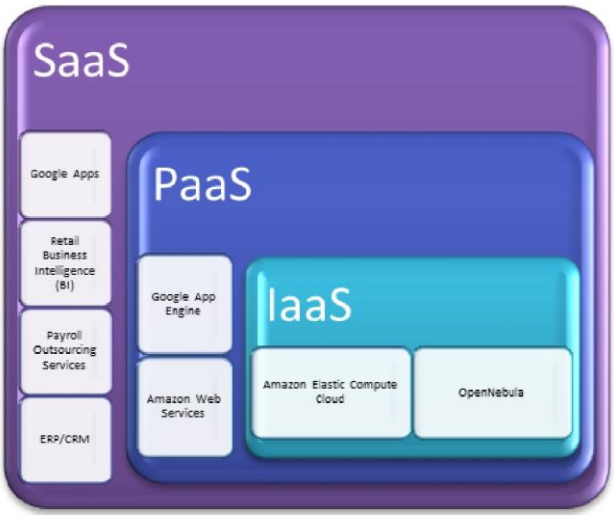

Figure 1 Cloud Service [5]

D. Cloud Deployment Model [1]

1) Public Cloud: The cloud infrastructure is made available to the general public or a large industry group and is owned by an organization selling cloud services.

2) Private Cloud: The cloud infrastructure is operated solely for a single organization. It may be managed by the organization or by a third party and may be located on-premise or off-premise.

3) Community Cloud: The cloud infrastructure is shared by several organizations and supports a specific community that has shared concerns (e.g., mission, security requirements, policy, or compliance considerations). It may be managed by the organizations or by a third party and may be located on-premise or off-premise.

4) Hybrid Cloud: The cloud infrastructure is a composition of two or more clouds (private, community, or public) that remain unique entities but are bound together by standardized or proprietary technology that enables data and application portability (e.g., cloud bursting for load-balancing between clouds).

E. Advantages and disadvantages of cloud computing [6]

Advantages:-

- Lower upfront costs and reduced infrastructure costs.

IJERT

- Easy to grow your applications.
- Scale up or down at short notice.
- Only pay for what you use.
- Everything managed under SLAs.
- Overall environmental benefit (lower carbon emissions) of many users efficiently sharing large systems.

Disadvantages:-

- Higher ongoing operating costs.
- Greater dependency on service providers.
- Risk of being locked into proprietary or vendor-recommended systems. How easily can you migrate to another system or service provider if you need to?
- What happens if your supplier suddenly decides to stop supporting a product or system you've come to depend on?
- Potential privacy and security risks of putting valuable data on someone else's system in an unknown location?
- Dependency on a reliable Internet connection

II. Security Issues of Cloud Computing

Top security threats given by Cloud Security Alliance to Cloud Computing are as follow [7].

1) Abuse and Nefarious Use of Cloud Computing: Abuse and nefarious use of cloud computing is the top threat identified by the CSA. Attackers can infiltrate a public cloud, for example, and find a way to upload malware to thousands of computers and use the power of the cloud infrastructure to attack other machines.

Suggested remedies by the CSA to diminish this threat:

- Stricter initial registration and validation processes.
- Enhanced credit card fraud monitoring and coordination.
- Comprehensive introspection of customer network traffic.
- Monitoring public blacklists for one's own network Blocks

2)Insecure Application Programming Interfaces: As software interfaces or APIs are what customers use to interact with cloud services, those must have extremely secure authentication, access control, encryption and activity monitoring mechanisms - especially when third parties start to build on them.

Suggested remedies by CSA to diminish this threat:

- Analyze the security model of cloud provider interfaces.
- Ensure strong authentication and access controls are implemented in concert with encrypted transmission.
- Understand the dependency chain associated with the API.

3) Malicious Insiders: The malicious insider threat is one that gains in importance as many providers still don't reveal how they hire people, how they grant them access to assets or how they monitor them. Transparency is, in this case, vital to a secure cloud offering, along with compliance reporting and breach notification.

Suggested remedies by CSA to diminish this threat:

- Enforce strict supply chain management and conduct a comprehensive supplier assessment.
- Specify human resource requirements as part of legal contracts.
- Require transparency into overall information security and management practices, as well as compliance reporting.
- Determine security breach notification processes.

4) Shared Technology Vulnerabilities: Sharing infrastructure is a way of life for IaaS providers. Unfortunately, the components on which this infrastructure is based were not designed for that. To ensure that customers don't thread on each other's "territory", monitoring and strong compartmentalization is required.

Suggested remedies by CSA to diminish this threat:

- Implement security best practices for installation/configuration.
- Monitor environment for unauthorized changes/activity.
- Promote strong authentication and access control for administrative access and operations.
- Enforce service level agreements for patching and vulnerability remediation.
- Conduct vulnerability scanning and configuration audits.

5) Data Loss/Leakage: Be it by deletion without a backup, by loss of the encoding key or by unauthorized access, data is always in danger of being lost or stolen. This is one of the top concerns for businesses, because they not only stand to lose their reputation, but are also obligated by law to keep it safe.

Suggested remedies by CSA to diminish this threat:

- Implement strong API access control.
- Encrypt and protect integrity of data in transit.
- Analyze data protection at both design and run time.

IJERT

- Implement strong key generation, storage and management, and destruction practices.
- Contractually demand providers to wipe persistent media before it is released into the pool.
- Contractually specify provider backup and retention Strategies

6) Account, Service & Traffic Hijacking: Account service and traffic hijacking is another issue that cloud users need to be aware of. These threats range from man-in-the-middle attacks, to phishing and spam campaigns, to denial-ofservice attacks.

Suggested remedies by CSA to diminish this threat:
- Prohibit the sharing of account credentials between users and services.
- Leverage strong two-factor authentication techniques where possible.
- Employ proactive monitoring to detect unauthorized activity.
- Understand cloud provider security policies and SLAs.

7) Unknown Risk Profile: Security should always in the upper portion of the priority list. Code updates, security practices, vulnerability profiles, intrusion attempts – all things that should always be kept in mind.

Suggested remedies by CSA to diminish this threat:
- Disclosure of applicable logs and data.
- Partial/full disclosure of infrastructure details (*e.g.*, patch levels, firewalls, etc.).
- Monitoring and alerting on necessary information.

III Security Provided by Cloud Computing

We have seen the security threats to the cloud computing. Among the above stated threats, the major threat of cloud computing is on the data while it is in the transit from cloud to the customer over the network or while it is accessed from the data centre. So Cloud service provider provides security using various Authentication and Encryption scheme. Here, we present some of the authentication and encryption techniques.

1) Identity-Based Authentication [8]:

Identity Based Authentication is a method of authenticating the user by its identity. Each user is given unique private key as its identity. This scheme is using the concepts of Identity-Based Hierarchical Model for cloud computing (IBHMCC).

Identity-Based Hierarchical Model for Cloud Computing (IBHMCC):

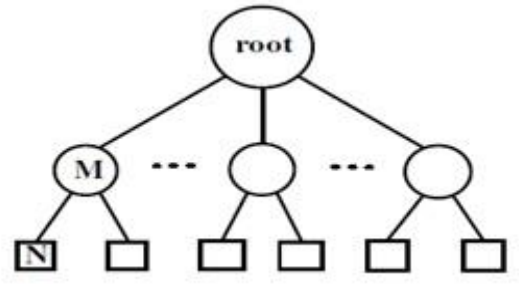

Figure 2 IBHM for cloud computing

As shown in Figure 2, IBHM for cloud computing (IBHMCC) is composed of three levels. The top level (level-0) is root Private Key Generator (PKG). The level-1 is sub-PKGs. Each node in level-1 corresponds to a data-centre (such as a Cloud Storage Service Provider) in the cloud computing. The bottom level (level-2) are users in the cloud computing.

In IBHMCC, each node has a unique name. The name is the node's registered distinguished name (*DN)* when the node joins the cloud storage service. For example, in the fig.2, *DN* of the root node is 0 *DN*, *DN* of node M is *M DN* and *DN* of node N is *N DN*.
We define the identity of node is the *DN* string from the root node to the current node itself. For example, the identity of entity N is *IDN = DN*0 || *DNM* || *DNN* where "||" denotes string concatenation.

The deployment of IBHMCC needs two modules: Root PKG setup and Lower level setup.

Root PKG setup: In this phase, it generates a group of prime number and selects any one which is used for finding Q-value. It selects cryptography hash function and calculates the Q-value of the root.

Lower-level setup: In this phase, it computes the public key of node, say X for level-1 and calculates the secret key for node X using the secret point for node X, which is known to node X and its parent node. Then it calculates the Q-value for node X. The secret point and secret key keeps secret while public key and Q-value is available to all nodes. This process is repeated to find value for all nodes in the level-1 and also for the level-2.

Identity-Based Encryption: IBE is based on the Root PKG setup and Lower-level setup algorithms. It is composed by two parts: Encryption and Decryption.

IJERT

Encryption: It performs the encryption using the identity of the node, hash function and public key of the node.

Decryption: It performs the decryption using the secret key, secret point and Q-value.

Identity-Based Signature: IBS is also based on Root PKG setup and Lower-level setup algorithms. It incorporates two algorithms: signature and verification.

Signature: It digitally sign the message send by the sender using the identity and secret point of the node.

Verification: It verifies the signature.

2) Elliptic Curves Cryptography [9]:

An elliptic curve over a field K is a nonsingular cubic curve in two variables, f(x,y) =0 with a rational point (which may be a point at infinity).

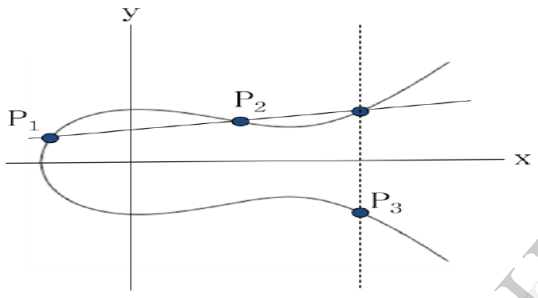

Figure 3 Elliptical Curve Cryptography

Consider elliptic curve E: $y^2 = x^3 - x + 1$. If P1 and P2 are on E, we can define addition P3 = P1 + P2.

Elliptic curve cryptography (ECC) is a public-key cryptosystem. Every user has a public and a private key. Public key is used for encryption/signature verification. Private Key is used for decryption/signature generation. Elliptic curves are used as an extension to other current cryptosystems. That is Elliptic Curve Diffie-Hellman Key Exchange and Elliptic Curve Digital Signature Algorithm.

It works as follow for the two Clouds A and B:

Key generation: It takes any two random number as the private key for A and B and generates the public key for A and B. This key will be used for the subsequent process.

Signature Generation: When cloud A wants to send some message to cloud B ,A has to digitally sign that message so that cloud B verify that it comes from cloud A only not from any attacker. These algorithms takes private key of A, message m send by cloud A and generates the signature and send it to B.

Encryption algorithm: Suppose A wants to send to B an encrypted message. So A takes plaintext message *M*, and encodes it onto a point, *PM*, from the elliptic group. A chooses another random integer, *k* from the interval [1, *p*-1] where *p*=prime. The cipher text is a pair of points given by *PC* = [(*kB*), (*PM* + *kPB)*]. Cloud A send ciphertext *PC* to cloud B.

Decryption algorithm: Cloud B will take the cipher text *PC* in the form of point on the elliptical curve and decrypt that into the form of message.

Signature Verification: When cloud B receives the message from A, cloud B authenticate A's signature using the public key *PA* of cloud A for ensuring that whether the incoming message is from A or from attacker.

3) Public Key Cryptography with Matrices [10]:

The Public Key Cryptography with Matrices is a three-stage secured algorithm and it has a constant complexity (fixed number of multiplications) irrespective of the key size given over the ring of integers. The working of each stage is as follows:

Stage-1 Shuffling of the Data: This stage involves the shuffling of the original data for which the linear congruential method is used and then the data is arranged in the form of a matrix of some dimension n x n. Suppose L be the length of the message to be encrypted. We consider here two arrays as follows:

1) index[1…….L] is an array containing all the indices of the message.

2) hash[1……..M] is the array containing some magic numbers such that when we apply the linear congruential method to the array index[1…L] then the output of the index[1…L] array does not contain any repeated indices and original message is shuffled or rearranged on the basis of the array index [1….L]

Stage-2 Traversing the Data Matrix: This stage involves reading out of the data from the data matrix of order NxN. We read data using one of the following traversing techniques: Spiral Traversal, Reversed Spiral Traversal, Helical Traversal, Sine Waveform Traversal, and Reverse Helical.

IJERT

Stage-3 Generation of private key and encryption and decryption: This stage deals with generating the system of non-homogeneous linear equations from which we generate the private keys and this key is used for encryption and decryption of the data. This stage is again subdivided into three stages.

In first stage, it generates system of non-homogeneous linear equations from which we generate the private keys and public key.

In second stage, both communicating party calculate sequence using their private key and exchange it with each other and both calculates the matrix based on the sequence and generates shared private key which is mutually agreed upon by them as key agreement.

In third stage, encryption is performed using the shared private key and send it to the other party which decrypt and find the necessary message.

4) Fully Homomorphic Encryption [11]:

One of the use of Cloud computing is to store personal or professional information which are stored outside the concerned structure (i.e. outside the company).To provide the security to this data in cloud means secure the treatments (calculations) and storage (databases hosted by the Cloud provider). So we encrypt data before sending it to the cloud and we allow cloud provider to perform the operations on encrypted data without decrypting them using the cryptosystems based on Homomorphic Encryption.

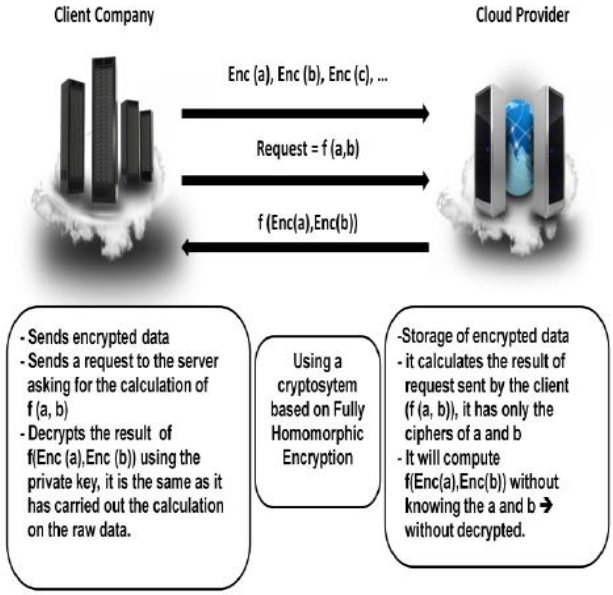

Fig. 4 Homomorphic Encryption applied to the Cloud Computing

Homomorphic Encryption systems are used to perform operations on encrypted data without knowing the private key (without decryption), the client is the only holder of the secret key. When we decrypt the result of any operation, it is the same as if we had carried out the calculation on the raw data. Figure 4 shows the exchange of messages in Homomorphic Encryption.

An encryption is homomorphic, if: from Enc(a) and Enc(b) it is possible to compute Enc(f (a, b)), where f can be: +, ×, ⊕ ,Enc means Encryption, and without using the private key. Among the Homomorphic encryption we distinguish, according to the operations that allows to assess on raw data, the additive Homomorphic encryption (only additions of the raw data) is the Pailler [12] and Goldwasser-Micalli [13] cryptosystems, and the multiplicative Homomorphic encryption (only products on raw data) is the RSA [14] and El Gamal [15] cryptosystems.

Suppose we have two ciphers C1 et C2 such that:
$C1 = m1^e \bmod n$
$C2 = m2^e \bmod n$
$C1.C2 = m1^e\, m2^e \bmod n = (m1m2)^e \bmod n$

The client sends the pair (C1, C2) to the cloud server, the server will perform the calculations requested by the client and sends the encrypted result (C1 × C2) to the client. If the attacker intercepts two ciphers C1 et C2, which are encrypted with the same private key, he/she will be able to decrypt all messages exchanged between the server and the client. Because the Homomorphic encryption is multiplicative or additive, i.e. the product or addition of the ciphers equals the cipher of the product or addition respectively.

For all types of calculation on the data stored in the cloud, we must opt for the fully Homomorphic encryption which is able to execute all types of operations on encrypted data without decryption.

In 2009 Craig Gentry of IBM has proposed the first encryption system "fully homomorphic" that evaluates an arbitrary number of additions and multiplications and thus calculate any type of function on encrypted data [16].

5) Attribute Based Cryptography [17]:

Attribute Based Cryptography (ABE) is a public-key cryptography which enforces access control. In ABE both the user secret key and the ciphertext are associated with a set of attributes. A user is able to decrypt the ciphertext if and only if at least a threshold number of attributes overlap between the ciphertext and

IJERT

user secret key. ABE is intended for one-to-many encryption in which ciphertexts are not necessarily encrypted to one particular user. Goyal et al. [18] proposed a key-policy attribute-based encryption (KP-ABE) scheme and ciphertextpolicy attribute-based encryption (CP-ABE).

Key-Policy Attribute-Based Encryption:

The idea of a KP-ABE scheme is as follows: The ciphertext is associated with a set of attributes and each user secret key is embedded with an access structure which can be any monotonic tree-access structure. A user is able to decrypt a ciphertext if and only if the ciphertext attributes satisfy the access structure embedded in her secret key. A KP-ABE scheme consists of the following four algorithms.

Setup: This algorithm takes as input a security parameter *K* and returns the public key *PK* as well as a system master secret key *MK*. *PK* is used by message senders for encryption. *MK* is used to generate user secret keys and is known only to the authority.

Encryption: This algorithm takes a message *M*, the public key *PK*, and a set of attributes *Y* as input. It outputs the ciphertext *E*.

Key Generation: This algorithm takes as input an access structure *T* and the master secret key *MK*. It outputs a secret key *SK* that enables the user to decrypt a message encrypted under a set of attributes *Y* if and only if *Y* matches *T*.

Decryption: It takes as input the user's secret key *SK* for access structure *T* and the ciphertext *E*, which was encrypted under the attribute set *Y*. This algorithm outputs the message *M* if and only if the attribute set *Y* satisfies the user's access structure *T*.

Ciphertext-Policy Attribute-Based Encryption:

In CP-ABE, the ciphertext is associated with an access structure and each user secret key is embedded with a set of attributes. A KP-ABE scheme consists of the following four algorithms.
Setup: This algorithm takes as input a security parameter *K* and returns the public key *PK* as well as a system master secret key *MK*. *PK* is used by message senders for encryption. *MK* is used to generate user secret keys and is known only to the authority.

Encrypt: This algorithm takes as input the public parameter *PK*, a message *M*, and an access structure *T*. It outputs the ciphertext *CT*.

KeyGen: This algorithm takes as input a set of attributes *Y* associated with the user and the master secret key *MK*. It outputs a secret key *SK* that enables the user to decrypt a message encrypted under an access structure *T* if and only if *Y matches T*.

Decrypt: This algorithm takes as input the ciphertext *CT* and a secret key *SK* for an attributes set *Y*. It returns the message *M* if and only if *Y* satisfies the access structure associated with the ciphertext *CT*.

IV. Conclusion

Now a day's cloud computing facing many security challenges along with regulatory compliance. At the same time, cloud users are attracted to cloud for its advantages like flexibility, elasticity and pay per usage model. Cloud users put their data in the cloud and transform from one cloud to another cloud at the risk of privacy of user data. Cloud users need strong security policies without affecting the advantages of cloud computing. There is always need to encrypt user's data so we can use any one of the encrypted techniques or some combination of encrypted techniques and make user data more secure. In future, we can develop more efficient encryption techniques which reduce the size of the key as well as reduce the time needed for encryption and decryption.

V. References

[1] Security Guidance for Critical Areas of focus in cloud computing v3.0 Prepared by Cloud Security Alliance, 2011.

[2] Sales force Customer Relationships Management (CRM) system, http://www.salesforce.com/

[3] The National Institute of Standards and Technology (NIST), Information Technology Laboratory definition of Cloud Computing by Peter Mell and Tim Grance, version 15, October 7, 2009.

[4] Security Guidance for Critical Areas of Focus in Cloud Computing V2.1 Prepared by Cloud Security Alliance December 2009.

[5] Megha Gupta, Syed Imtiyaz Hassan "Improving scope of Cloud technology under Open Source Tool " UNIASCIT, Vol 2 (1), 2012, pp. 173-178.

[6] Chris Woodford http://www.explainthatstuff.com/cloudcomputingintroduction .html June 22, 2012.

[7] Top Threats to Cloud Computing V1.0 Prepared by the Cloud Security Alliance March 2010.

[8] Hongwei Li, Yuanshun Dai, Ling Tian, and Haomiao Yang "Identity-Based Authentication for Cloud Computing" M.G. Jaatun, G. Zhao, and C. Rong (Eds.): CloudCom 2009, LNCS 5931, 2009. pp. 157–166.

[9] Veerraju Gampala, Srilakshmi Inuganti, Satish Muppidi "Data Security in Cloud Computing with Elliptic Curve Cryptography" International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-3, July 2012.

[10] Birendra Goswami, Dr.S.N.Singh "Enhancing Security in Cloud computing using Public Key Cryptography with Matrices" International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-962 Vol. 2, Issue 4, July-August 2012, pp.339-344.

[11] Maha TEBAA, Saïd EL HAJJI, Abdellatif EL GHAZI "Homomorphic Encryption Applied to the Cloud Computing Security" Proceedings of the World Congress on Engineering, London, U.K. ISBN: 978-988-19251-3-8 ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online), Vol I, July 4 - 6, 2012.

[12] Pascal Paillier Public-key cryptosystems based on composite degree residuosity classes. In 18th Annual Eurocrypt Conference (EUROCRYPT'99), Prague, Czech Republic, volume 1592, 1999.

[13] Julien Bringe and al. An Application of the Goldwasser-Micalli Cryptosystem to Biometric Authentication, Springer-Verlag, 2007.

[14] R. Rivest, A. Shamir, and L. Adleman. "A method for obtaining digital signatures and public key Cryptosystems" Communications of the ACM, 21(2) pp. 120-126, 1978. Computer Science, Springer, 1999, pp. 223-238.

[15] Taher ElGamal. "A public key cryptosystem and a signature scheme based on discrete logarithms" IEEE Transactions on Information Theory, 1985, pp. 469-472.

[16] Craig Gentry, A Fully Homomorphic Encryption Scheme, 2009.

[17] Shucheng Yu "Data Sharing on Untrusted Storage with Attribute-Based Encryption" Worcester Polytechnic Institute July 2010.

[18] V. Goyal, O. Pandey, A. Sahai, and B. Waters. "Attribute-Based Encryption for Fine-grained Access Control of Encrypted Data" In *Proc. of CCS'06*, Alexandria, Virginia, USA, 2006.

IJERT