International Journal of Computer Sciences and Engineering Open Access
Survey Paper Volume-2, Issue-4 E-ISSN: 2347-2693

A Survey on Security in Cloud Using Homographic and Disk Encryption Methods

Varun K. Reddy[1*] and Jagadeeshwar E. Rao[2]

[1*,2]*Dept. of Computer Science & School of IT, Jawaharlal Nehru Technological University, India*

www.ijcseonline.org

Received: 02/03/2014 Revised: 17/03/2014 Accepted: 19/04/2014 Published: 30/04/2014

***Abstract*—** Offering strong data protection to cloud users while enabling rich applications is a challenging task. The cryptographic security solutions existing and applied over internet cannot directly be applied to the cloud environment. This paper presents a couple of security solutions. Several homogaphic and disk encryption methods and their limitations are discussed. The efforts to combine the benefits of both the techniques give rise to further more possibilities. The author hopes that all the possibilities are researched and a single platform layer security suit will be made for cloud data protection.

***Index Term*—** Cloud Security; Homographic Encryption; Disk Encryption; Combine Benefits; Security Suit

I. INTRODUCTION

Cloud computing promises rapid scaling, cost effective, easy maintenance and service availability anytime and anywhere. A survey conducted by Microsoft found that "59 percent of public and 85 percent of business leaders are excited about the possibilities of cloud computing. But more than 91 percent of are worried about *security, privacy and availability* of their data that would rest in the cloud."Until now, the cloud offered very little platform level support for user data protection beyond data encryption. Currently, users rely primarily on legal agreements and implied economic harm as a proxy for application trustworthiness.

Major challenge in designing a platform layer solution is allowing rapid development and maintenance. Developers do not want to lose their users in attempt to solve their security problems! To ensure a practical solution, we consider goals relating to data protection as well as ease of development and maintenance.

- *Integrity*: Faithful storage of user's private data and guarantee that it shall not be corrupted.
- *Privacy*: Unauthorized person shall not have access to the user's private data.
- *Access transparency*: The logs of accesses to data indicating who or what performed each access should be easily available.
- *Ease of verification*:. Users should easily be able to verify on what platform or application the code is running. They may also wish to verify whether their privacy policies have been strictly enforced by the cloud or not.
- *Rich computation*: The platform should allow computations on sensitive user data, and should be able to run those computations efficiently.
- *Development and maintenance support*: Any developer probably faces a huge list of challenges: frequent software upgrades or patches, find bugs and fix, demand for high performance by the users and continuous change of usage patterns. A credible ad efficient data protection approach must deal with these matters, which are very often overlooked in the literature on the topic.

This paper initially presents a survey on various homographic and disk encryption techniques, their limitations and comparison of both. Further key management issues in cloud are discussed. Also the auditing schemes to further improve the security in cloud are discussed. Finally, a technique- "Data protection as a service" security suit is discussed which combines benefits of both the homographic encryption and disk encryption.

II. LITERATURE REVIEW

Literature survey is an important step in software development process. Before developing any tool it is required to determine the cost, time factor and strength of the company. Once these things are checked, next task is to determine which OS or language can be used to developing the tool. Once the programmers begin building the tool, they are going to need a lot of external assistance. This support can be obtained from websites or senior programmers. Before building the system the below understanding are required for developing the proposed system.

2.1. Cloud Computing

Cloud Computing is a general term used to describe a new class of network based computing that takes place over the Internet. It is basically a step on Utility Computing and a collection/group of networked hardware, software and Internet infrastructure (called a platform). A cloud is remotely hosted and is Ubiquitous.

Cloud computing is one such technology that uses both the internet and remote central servers to maintain applications and data. Cloud computing enables consumers to use applications without any overhead installations and access their private files from any computer device with internet

Corresponding Author: *Varun K. Reddy,*
Dept. of Computer, Jawaharlal Nehru Technological University. India

connectivity. This technology offers much more efficient computing by centralizing memory, processing, bandwidth and storage. Cloud is generally built on clusters of computer servers and Open Source software combined with in-home applications and some system software.

Consider a simple example of cloud computing as in fig.1, wherein 2 enterprises are connected to cloud. Both the enterprises are connected through internal LAN, which then connects to a router and further connected to internet through a gateway. The cloud provider is present at a remote location away from the enterprises. Now the enterprises use cloud services such as storage and other infrastructure. The enterprises act as thin clients and they use light weight machines. The cloud is where the actual infrastructure, hardware and software lie. The enterprises are just a mere user interfaces. The consumer gets to use the software alone and enjoy the benefits. The analogy is, 'If you need milk, would you buy a cow?'

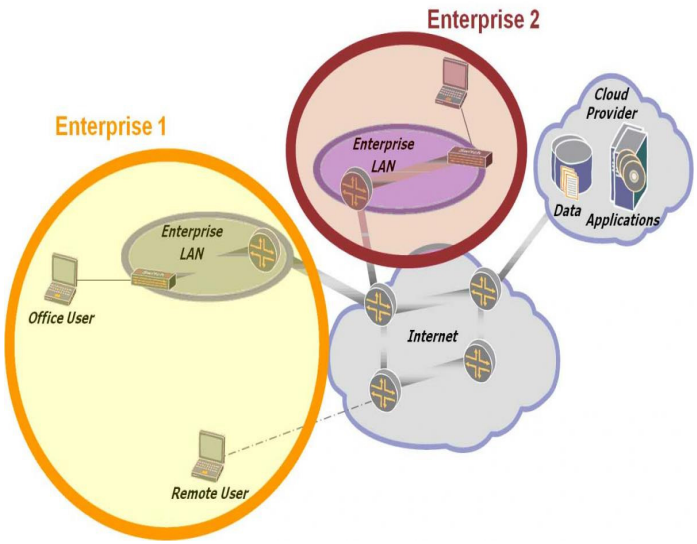

Fig. 1 Example of a cloud environment.

III. RELATED WORK

Several surveys on data security solutions in cloud computing have been conducted and are presented. Each of them has some special advantages yet posses a few drawbacks.

3.1. Fully homographic encryption (FHE) security for cloud

How can one be sure that even if the data-centers of the Cloud Computing provider were attacked, user data won't be stolen or reused? And how can user's data remain confidential and invisible even to the Cloud provider?

Homomorphic Encryption systems are used to perform tasks on encrypted data without the knowledge of the private key i.e. without decryption; the client is the sole holder of the secret key. The basic concept is to encrypt the data before sending it to the Cloud provider. But, then the user will have to decrypt the data each time he to work on it. The user will need to give the secret key to the server to decrypt the data before executing the calculations required, and this might affect the confidentiality of data stored in the Cloud. The Homomorphic Encryption method is one which allows performing operations on encrypted data without decrypting them.

In 1978 Ronald Rivest, Leonard Adleman and Michael Dertouzos suggested the concept of Homomorphic encryption for the first time. Then encryption system of Shafi Goldwasser and Silvio Micali was proposed in 1982, made it a provable security encryption scheme which reached a remarkable level of safety. This scheme was an additive Homomorphic encryption, but the drawback is that it can encrypt only a single bit. Using the same concept Pascal Paillier in 1999, also proposed a provable security encryption system and that was also an additive Homomorphic encryption. Few years hence, Eu-Jin Goh, Dan Boneh and Kobi Nissim in 2005, invented a provable security encryption, with which enable unlimited number of additions to be performed, but only one single multiplication.

An encryption is said to be homomorphic, if from Enc (a) and Enc (b) it is possible to compute Enc (f (a, b)), where f can be an addition, multiplication or ex-or: +, ×, ⊕ and without applying the private key. Homomorphic encryption is distinguished based on the operations that allow assessing the raw data. The additive Homomorphic encryption (only additions of the raw data) is the Pailler and Goldwasser-Micalli cryptosystems, and the multiplicative Homomorphic encryption (only products on raw data) is the RSA and El Gamal cryptosystems.

- *A Homomorphic encryption system is additive, if:*

  $Enc(p \oplus q) = Enc(p) \otimes Enc(q)$

  $Enc(\Sigma xi) = \Pi Enc(xi)$

Suppose we have two ciphers C1 et C2 such that, if the product of encryption of them is same as encryption of cipher formed by combining both, then it is additive homographic encryption.

$C1 = gm1.\ r1n \bmod n2$

$C2 = gm2.\ r2n \bmod n2$

$C1.C\ 2 = gm1.\ r1n\ .\ gm2.\ r2n \bmod n2 = gm1+ m2$

$(r1r2)n \bmod n2$

An application of an additive Homomorphic encryption is electronic voting system in each which vote is encrypted but only the "sum" is decrypted.

- *Multiplicative Homomorphic Encryption*

  A Homomorphic encryption system is multiplicative:
  Enc (p⊗q) = Enc(p) ⊗ Enc(q)

  Enc (Πxi) = Π Enc (xi)

Suppose we have two ciphers C1 et C2 such that, if the product of encryption of them is same as encryption of cipher formed by combining both, then it is additive homographic encryption.

C1 = m1e mod n

C2 = m2e mod n

C1.C2 = m1em2e mod n = (m1m2) e mod n

RSA cryptosystem realize the properties of the multiplicative Homomorphic encryption, but it still lacks in security, because if we assume that two ciphers C1, C2 which corresponds to the messages m1, m2 respectively, such that:

C1 = m1e mod n

C2 = m2e mod n

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

FHE provides a good security solution, but still needs little advancement and not only that, it is also slow in performance. A goggle search would take 1 trillion more time using FHE.

3.2. Fully disk encryption (FDE) for cloud

(FDE) means to apply encryption transparently to the entire hard disk in order to enforce data security in scenarios where a disk, or a whole machine, is physically lost or stolen.

Full disk encryption can be classified into software-based and hardware-based solutions. Examples of software-based are BitLocker and True-Crypt . These where present for end-users for over a decade, hardware-based solutions have made their breakthrough only recently with the advent of solid-state drives (SSDs).These disks have an inbuilt encryption logic inside the disk drive controller, so that encryption keys are never present in the PC's main memory or CPU. Hence, such systems are usually called self-encrypting disks or self-encrypting drives (SEDs). Apart from this undisputed advantages of this method, such as maintainability, a significant gain in performance and OS transparency. Software-based solutions are commonly believed to be less secure than SED's.

Particularly, we look on SSD models that provide inbuilt encryption depending on ATA security authentication. Since the1990s, TCG Storage Security provides another authentication method. Advanced technology attachment (ATA) was made as a standard interface for plugging mass storage devices. The ATA security feature (a.k.a. HDD Password, Drive Lock, or Security Lock) was standardized in ATA-3 (1997). Today, serial ATA (SATA), standardized in ATA-7(2003), is the prevalent bus for connecting SSDs. ATA drives have the capability to get locked and remain inaccessible until a correct password is entered, but a drive lock does not necessarily involve encryption.

ATA security defines two kinds of passwords, user and master, as well as two security levels, high and maximum. On high security level, the user and master password can be used interchangeable for unlocking the drive. On maximum level, the master password can only be used to securely erase the drive and to reset the user password, but not to read data. In other words, the master password enables a company or vendor to reset disks in the case of password loss. Both user and master passwords are defined to be 32-byte long.

SSD-vendors shipped their products with inbuilt encryption facilities, so called self-encrypting disks (SEDs). AES standard are used by these to encrypt user data. There is a unique encryption key for each of the SED, which is generated from entropy sources inside the drive. This key is called the media encryption key (MEK), also known as data encryption key (DEK). The MEK is used to encrypt the actual user data and is encrypted by means of a key encryption key (KEK). KEK's are derived from user passwords. And the disks are powered up-locked until the correct password is entered. MEK must be encrypted newly each time with the KEK when a password is changed, this avoids re-encryption. And also setting of new password, or resetting of a password does not involve lengthy encryption or decryption procedures either.

CG Storage Security: Opal, Another infrastructure that SEDs can be build on is the standardized Opal Security Subsystem Class (Opal SSC) from the Trusted Computing Group (TCG) . Opal SSC is a set of specifications that SED vendors can comply with to integrate their hardware into a trusted platform host. Some SED manufacturers provide OPAL compliance for their devices, others not.

Hot Plug Attacks: In this variant of the cold boot attack, RAM chips are removed from running PCs and then re-plugged into another PC in order to extract their contents. But as it is pointless to re-plug RAM chips in the context of hardware-based FDE, we considered re-plugging the disk

itself instead. This idea turned out to be an effective though simple attack against all SEDs. We call this attack hot plug attack because it requires the disk to be running and unlocked before seizure. PC systems that are switched off completely, such that the disk is off and locked cannot be attacked by hot plug attacks.

3.3. Key management issues in cloud

Referring to the paper, "Cryptographic Key Management" by Dr Keith Martin, in which he discussed about key management issues and techniques.

The security of the system mainly depends on the security of the keys. Irrespective of the algorithm used a security system without strong management procedures, there is no security.

In case of FDE, the disk is encrypted with data kept inside of it. The data itself is not encrypted. So, the keys are kept with the cloud itself. Data in the physical disk is safe as long as it is in the disk, but when it is transported it is done in the clear form. Thus, there exists a threat of leakage. Moreover the user must trust the cloud completely with his data, while the cloud provider may hire a third party which might turn out to be risky.

Whereas if the keys are owned and managed by the user. It might be difficult for the cloud to perform any local user maintenance and management operations. And moreover, the safety and the reliability of the keys with the user are to be doubted. After all the main theme of cloud computing is to overcome user maintenance.

3.4. Auditing in cloud computing

Auditing is the process of evaluating the claims against the actual facts to guarantee compliance. Auditing is done to ensure that systems are what they claim to be and there is someone to take responsibility for. A malicious program may find sideways to access the data in the cloud. Thus auditing may catch improper usage. By auditing a user knows what data is been accessed, when and by whom. Thus auditing bolsters the confidence of the user.

A public auditing scheme is one such auditing scheme which provides a complete outsourcing solution not only the data itself, but also its integrity checking. A public auditing scheme consists of four algorithms (KeyGen, VerifyProof , SigGen, GenProof). KeyGen is an algorithm to generate key which is run by the user to setup the scheme. SigGen is used by the user for the generation of verification meta-data, which may consist of digital signatures. GenProof is run by the cloud server to generate a proof of data storage, while VerifyProof is used to audit the proof.

Running a public auditing system consists of two phases, Setup and Audit: In setup phase the user initializes the public and secret parameters of the system by executing KeyGen, and preprocesses the data file F by using SigGen to generate the verification meta-data. Whereas, in audit phase the TPA challenges to the cloud server to make sure that the cloud server has retained the data file F properly at the time of the audit. The TPA then verifies the response via VerifyProof.

3.5. Comparison of FHE and FDE techniques

Until now we have seen advancements made in fully homographic encryption and fully disk encryption security solutions. Both of them have their own merits and demerits.
Data aggregation is nothing but performing various operations on data such as data mining, data statistics etc. In fully disk encryption the encryption keys are with the cloud itself, hence data aggregation is easy. Whereas in case of fully homographic encryption each data unit is encrypted with different keys, hence computation on data is not possible

Performance is the rate at which an operation is done. Fully disk encryption is a disk firmware and it uses symmetric encryption which allows it to run at disk's full bandwidth. Fully homographic encryption has slower response. A goggle search would take a 1 trillion more time using fully homographic encryption.
Ease of development is more in case of fully disk encryption. Fully disk encryption is hidden behind an abstraction of physical disk and has no impact on application development.

Maintenance deals with activities such as debugging of bugs. Quickly debugging of bugs is top priority in cloud. To determine what went wrong would be difficult with fully homographic encryption.

Privacy and invisibility are very important aspects of security. Fully homographic encryption encrypts the data unit and not the disk and hence the security is always with the data. Moreover, the data itself remains invisible to the server which encrypts it.

IV. A WAY FORWARD

We have seen that disk encryption offers better performance and ease of development, while the homographic encryption is good at providing privacy and invisibility. So, a better security solution would be to combine both the methods and their benefits. One such security suit proposed is "Data protection of a service" paradigm (DPaaS).

DpaaS security suit performs keying on sharable data units thus implementing FHE. It uses symmetric encryption and thus implementing the advantages of FDE. key management and access control is moved to a middle tier for easy maintenance and user side verifiability. In addition auditing of the user data is also implemented which further helps to bolster the users confidence.

In DPaaS a TPM chip is used to provide trust.TPM can load or execute TC which provides dynamic root of trust by isolation, key management, access control and logging. TPM provide confinement by using SDC's and SEE's. A secure data capsule is a data unit packed with security policy, for example ACL capsule. A secure execution environment prevents leakage of user data through isolation. A SEE could be a pool of VM's or a more light weight approach is to use OS process isolation. SDC's can be imported or exported to outside websites using policies. Additionally DPaaS can log all the instances where the data is exported. To confine the data the platform decrypts the SDC's data only in SEE in compliance with the policies. The platform also enforces ACL modifications, sharing or un-sharing and the creator of the data units can add or revoke other authorized users. DPaaS uses AES which requires more performance but once the data is loaded into SEE's it need not encrypt or decrypt again.

Auditing is done using logs. There are four basic types of logs- online data access logs, access control modification by authorized user logs, logs of offline data requests and logs relating to maintenance operations. DPaaS gives the flexibility of even using third party auditing. Ordinary user access is governed using ACL's whereas the administrators access is governed using administrative policies.

DPaaS offers platform verifiability. This is done by logging and auditing at platform level so that all the applications running on it can share its benefits. When in offline mode the auditor verifies the working of protection features of platform. At runtime the platform provides user with trusted computing to at least for software or an application running on it.

V. CONCLUSION

Cloud computing offers utility based computing that is, pay per usage without any maintenance overhead. It saves a lot of time as it offer services anytime, anywhere and with any device. Not just time but space and cost of equipment are also saved. Cloud computing in future aims at implementing 'Internet of Things.' IOT aims at establishing network with all the objects in the world. This makes the cloud accessible from anywhere. These entire features make cloud computing most preferred for business and personal usage. Yet many people are finding it difficult to accept it, due to security concern.

Disk encryption and homographic encryption techniques are two such methods proposed to provide security in cloud. Disk encryption offers better performance, ease of development, but security is confined to disk and not the data. Homographic encryption on the other hand offers good privacy and invisibility even to the server which is encrypting the data. In this the security lies with the sharable data unit. Though homographic encryption offers better privacy, it is slow in performance and ease of development for the programmer.

This paper discusses various homographic encryption techniques such as homographic addition, multiplication, EX-OR and their limitations. Similarly disk encryption techniques such as hardware disk encryption and software disk encryption techniques are stated.

The management of encryption keys play a crucial role in overall security, as keys are not secured means zero percent security. The two key management method are been mentioned in this paper. The keys can either be placed with the cloud or with the user based on the purpose.

Auditing is a great way to bolsters the confidence of the users. The users are always kept aware of their data and operation performed on it, such as modifications, additions, permissions, sharing etc. A public auditing scheme is one such auditing technique which consists of four algorithms (KeyGen, VerifyProof , SigGen, GenProof.

Data protection as a service is one such security suit which combines all the aspect and emerges as a single solution for security in cloud. It implements both homographic and disk encryption, so that benefits of either are obtained. Thus it is faster and more secure. key management and access control is moved to a middle tier for easy maintenance and user side verifiability. In addition third party auditing facility is also provide to the user, which further helps to bolster the users confidence.

But still many barriers are yet to be overcome, one challenge for cloud security is that the software keeps updating with patches. Then how to perform its attestation? One way would be to log the update history. Other challenge is that Software verification is expensive due to large number of users, so auditors are required to use certifications to verify them.

ACKNOWLEDGMENT

This part should be in Times New Roman, 11 This work is supported and made under the esteemed guidance of E. Jagadeeshwar rao garu, who is a lecturer in software engineering department at School of IT, Jawaharlal Nehru Technological university, Hyderabad.

REFERENCES

[1] C. Wang, Q. Wang, K. Ren, and W. Lou, "Privacy-Preserving public auditing for storage security in cloud computing," in Proc.of IEEE INFOCOM'10, March 2010.

[2] Maha TEBAA, Saïd EL HAJJI, Abdellatif EL GHAZI, "Homomorphic Encryption Applied to the Cloud Computing Security." In proc. Of the World Congress on Engineering 2012 Vol I WCE 2012, July 4 - 6, 2012, London, U.K.

[3] Dr Keith Martin, "Cryptographic Key Management," in 2006 Information Security conference, Royal Holloway, University of London, United Kingdom.

[4] Tilo Müller, Tobias Latzo, and Felix C. Freiling, "Self-Encrypting Disks pose Self-Decrypting Risks." In Annual Computer Security Applications Conference (ACSAC), Orlando, Florida USA, Dec. 2011. University of Illinois at Urbana-Champaign, ACM.

[5] Shivlal Mewada, Umesh Kumar Singh and Pradeep Sharma, "Security Enhancement in Cloud Computing (CC)", ISROSET-International Journal of Scientific Research in Computer Science and Engineering, Volume-01, Issue-01, Page No (31-37), Jan -Feb 2013

[6] C. Gentry, "Fully Homomorphic Encryption Using Ideal Lattices," Proc. 41st Ann. ACM Symp. Theory Computing (STOC 09), ACM, 2009, pp. 169-178.

[7] B. Bosen. "FDE Performance Comparison: Hardware Versus Software Full Drive Encryption. "In Trusted Strategies LLC, Jan. 2010.

[8] Dawn Song, Elaine Shi, and Ian Fischer, "Cloud Data Protection for the Masses." Published by the IEEE Computer Society, 0018-9162/12/$31.00 © 2012 IEEE.

[9] Shaheen Ayyub1, Devshree Roy, "Cloud Computing Characteristics and Security Issues." In International Journal of Computer Sciences and Engineering, Vol.-1(4), pp (18-22) Dec 2013.

