International Journal of Engineering Research-Online
A Peer Reviewed International Journal
Articles available online http://www.ijoer.in

Vol.2., Issue.1., 2014

RESEARCH ARTICLE

ISSN:2321-7758

# STUDY AND ASSESSMENT OF FULLY HOMOMORPHIC ENCRYPTION SCHEMES OVER CLOUD COMPUTING

**VAKITI PULLAMRAJU**
M.Tech Scholar, G Pullaiah College of Engineering & Technology,Kurnool, Andhra Pradesh, India

Article Received: 18/02/2014 | Article Revised on: 04/03/2014 | Article Accepted on:8/03/2014

**VAKITI PULLAMRAJU**

**ABSTRACT**

The safety and privacy issues associated with cloud computing have become the main problem for businesses to move towards cloud computing. various technologies have been used to fair these issue including various access control methods and cryptography techniques. likely that fully homomorphic encryption has for cloud computing is its facility to perform computations on encrypted data without previous decryption. Since Gentry's implementation of the first fully homomorphic scheme in 2009 many interesting variants and schemes have been proposed and developed to improve the performance, reduce the complexity and the cost of the scheme. Two interesting schemes are reviewed and discussed in this paper. The first scheme discussed in this paper is "Fully Homomorphic Encryption over Integers", by Marten Van Dijk et al. The second scheme is "Fully Homomorphic Encryption without Bootstrapping" by Brakerski et al. The scheme analysis is mainly focused on the security, performance and complexity factors of the mentioned schemes.

Keywords: Fully Homomorphic Encryption, homomorphic scheme, Gentry's implementation

## INTRODUCTION

In recent years, distributed systems and especially cloud computing are developing at a high speed. The economic benefits achieved through resource sharing and the greater degree of flexibility in scaling resources has pushed the cloud into mainstream computing.

However, the cloud inherits most information security problems from traditional computing platforms. In addition, the distributed nature of the cloud enables many new types of attacks. There are several major problems that the cloud faces:

- The cloud may be untrusted. The cloud service provider (CSP) is not necessarily trusted. For example, a malicious Google employee may be able to setup back doors and bypass all the protection over the Google cloud services. In addition, some machines in the cloud may be mismanaged, making them vulnerable to attacks. Even further, some machines may belong to attackers.
- Implementation bugs can be exploited. Even if the CSPs are trusted and they provide isolation mechanisms such as sandboxing and virtualization. Bugs in the system of which more are discovered every day, may be exploited to circumvent any protection, e.g. [1]. As an example, in [2] the authors show that an attacker could take control of the VMware and Xen virtualization software when moving a virtual machine from one physical computer to another.
- Side channel attacks can bypass protection. Even if the system is fully secure and the code is executing in a trusted environment, the side channel attacks may still compromise the security. For example, an attacker using the cold boot attack [3] is able to retrieve sensitive data from the unrepressed DRAM after using a cold reboot to restart the machine. An attacker using the

  Branch prediction attacks [4] can gather information about the encryption keys by simply monitoring the CPU time. These attacks typically require physical access to the machines, which is not an easy task traditionally. However, in cloud computing settings, it is possible that your code will be executed in a machine belongs to the attacker. In such cases, the attacker will be able to gain physical access to the machine easily.

## 1. Back ground

### 1.1 Homomorphic Encryption Schemes:

The development of homomorphic encryption provides yet another clear-cut approach to build SFE protocols. Informally, a homomorphic encryption scheme allows computation directly on encrypted data. It is clear that a SFE protocol can also be build quite straightforward using HE. Alice can now encrypt the input x and send the ciphertexts to Bob. Bob will compute f(x) directly on the ciphertext and send back the encrypted result that only Alice can decrypt. In this way, Bob will not be able to learn anything about x as long as the security of the homomorphic encryption scheme holds. Homomorphic properties of standard public key encryption schemes,e.g. RSA and ElGamal encryption, were recognized early on [7]. However they were largely viewed as a weakness rather than an asset. Applications where data is static typically require non-malleable encryption. However, the community has grown to trust the security of these schemes and, recently, the work of Gentry and others

demonstrate that, when carefully employed, such homomorphic properties can be quite valuable. Indeed, a number of recent specific applications such as data aggregation in distributed networks [8, 9], electronic voting [10], biometrics [11] and privacy 3

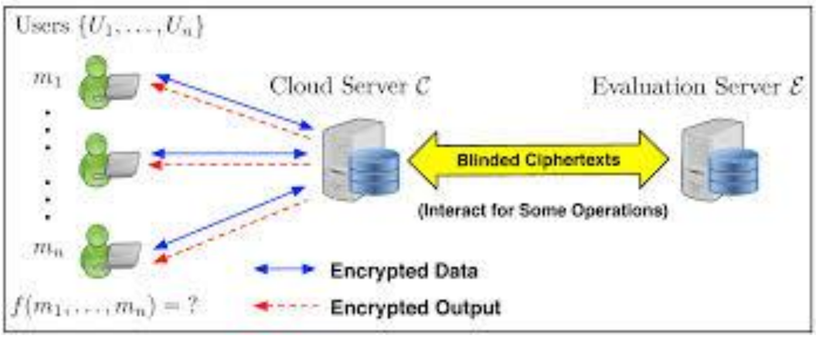

Fig 1. Homomorphic encryption scheme

Limitation of Partial HE Schemes: Clearly, partial HE schemes are useful in certain applications. In addition, the efficiency of some partial HE schemes is high enough for practical applications. E.g. the Paillier scheme can perform evaluations in milliseconds level. However, the drawbacks of this family of schemes are also clear.

The main problem of the partial HE schemes is the range of the circuits that they support. Most partial HE schemes only support one type of operation, e.g. additions for Paillier and multiplications for RSA. This draws a heavy restriction on the circuits that the HE schemes can evaluate homomorphically. Some partial HE schemes supports more than one operation, however, restrictions still exist. The Boneh-Goh-Nissim scheme support one level of multiplications via bilinear maps. This feature enables it evaluating 2-DNF formulas which cannot be evaluated using single-operation partial HE schemes. However, only one level multiplication is supported. The Boneh-Goh-Nissim scheme cannot handle more complicated circuits.

Somewhat Homomorphic Encryption Scheme: FHE schemes without “refreshing” the noise can also be employed as partial HE schemes. These schemes usually support a large number of additions and limited levels of multiplications. HE schemes with this property are usually referred to as Somewhat Homomorphic Encryption Schemes (SWHE). Although this type of partial HE schemes can support much more complicated circuits than the single-operation ones, it is still heavily restricted since the limitation on levels of multiplications will eventually be reached. Some partial HE schemes have additional bottlenecks that prevent them from being employed for practical applications. For example, the Boneh-Goh-Nissim scheme requires a small message size to achieve tractable decryption efficiency, which imposes extra limitation to the scheme.

## 1.2 GENTRY’S Fully Homomorphic Encryption Schemes

The idea of fully homomorphic encryption was raised by Rivest, Adleman and Dertouzos [7], shortly after the invention of RSA [8]. A fully homomorphic encryption scheme consists of following four algorithms:

- KeyGen (λ) - Generates the encryption keys. It takes the security parameter λ as an input, and generates the secret key sk and the public key pk.
- Enc (pk, m) – Encrypts the plaintext m with the public key pk to create ciphertext c.
- Dec (sk,c) – Decrypts the ciphertext c using the secret key sk to retrieve the plaintext m.
- Eval (pk,C,c1,c2 … ct) - Uses a Boolean circuit C to outputs a ciphertext of f(m) such that Decrypt (sk, m) = f(m).

Gentry’s construction consists of three main elements: a somewhat homomorphic encryption scheme that can evaluate low degree polynomials, a technique to “squash the decryption circuit”to get a “bootstrappable” scheme and finally a method of transferring “bootstrapping” the scheme into fully homomorphic encryption scheme. The significant point in this process is to obtain a scheme that can evaluate high degree polynomials while the decryption procedure can still be expressed as low degree polynomial. Once the scheme can evaluate its own decryption function plus an additional operation then it is called “bootstrappable” scheme and can be converted into a fully homomorphic scheme.

Although Gentry scheme proved the possibility of implementing fully homomorphic encryption, the scheme complexity, efficiency and performance needed to be improved. For example Gentry has estimated that building a circuit to execute an encrypted Google search with encrypted keywords would multiply the current computing time by 1 trillion. Nevertheless the scheme has inspired many researchers to propose many variants to Gentry’s scheme to improve the performance and reduce the complexity and ciphertext size. Two interesting approaches were discussed in the following sections.

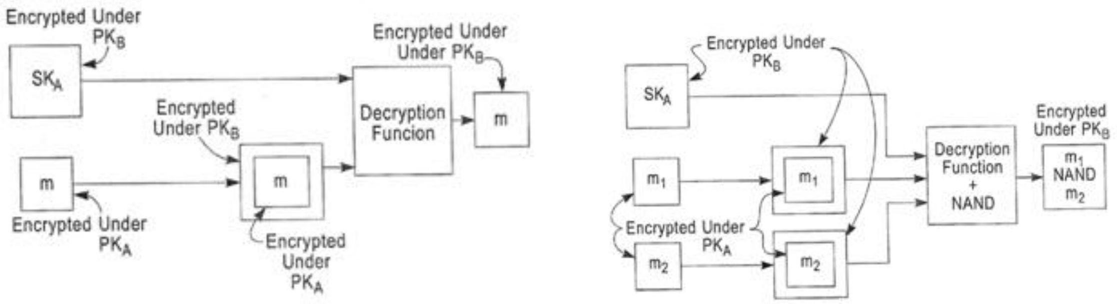

Fig 2 & 3. FHE scheme

## 3. ANALYSIS AND COMPARISON

In this section Dijk's and Brakerski's schemes are compared and analyzed for the security, efficiency and complexity factors.

### A. Fully Homomorphic Encryption over Integers Scheme

Security: The security of this scheme is based on the hardness of the approximate-gcd problem. With appropriate selection of parameters the scheme has proved to resist different types of attacks to recover the secret key including brute-force attack with at least $2\lambda$ time. However it has been proven that the scheme can be attacked to recover the plaintext from ciphertext using lattice reduction algorithm [13]. The parameters settings which have been used in the attack were considered to be appropriate by the Dijk's scheme.

Performance: The noise factor grows large when addition and multiplication operations are performed, it doubles on addition and squares on multiplication. Multiplying ciphertexts generates a noise factor with a size equals ~2dn where n is the number of operations. When the noise factor grows above q/2 the cipher text the original message cannot be rectified. In other hand, in order to improve the security of the scheme the cipher text was selected to have a large value n6, this value increase with multiplication which also results on degraded efficiency.

Complexity: Reducing the Gentry's scheme complexity was the main purpose of developing this scheme. The complexity of the somewhat scheme was reduced by using additions and multiplications over integers instead of ideal lattices. Perhaps one of the significance of this scheme is that it proved out that different mathematical approaches and theories can be applied to construct a fully homomorphic encryption scheme using Gentry's blueprint.

### B. Fully Homomorphic Encryption without Bootstrapping Scheme

Security: The security of this scheme is based on the hardness of lattice problems with quasi-polynomial approximation factors. The achieved level of security has not improved from original FHE scheme as it remains $2\lambda$ time against known lattice attacks. Since the scheme is relatively new, it is probably still too early to confirm its security strength against different types of attacks with great confidence.

Efficiency: Brakerski has developed a novel noise management technique that controlled the noise level so that it increases linearly with multiplication instead of exponentially. Theoretically, this scheme beats previous bootstrapping-based FHE schemes performance-wise. The scheme also allowed for L-level arithmetic circuit to be evaluated with $\tilde{O}(\lambda.L3)$ per-gate computation or instead of $\Omega(\lambda4)$ which is a large polynomial in the security parameter. The removal of the bootstrapping technique has also resulted on real cost reduction as the cost of bootstrapping in only $\theta(\lambda)$ time was $\Omega(\lambda 4)$. This allowed for evaluating deeper circuits at a lower cost. Applying batching and bootstrapping as optimization techniques can achieve a better per-gate computation of $\tilde{O}(\lambda 2)$ independent of number of levels.

Complexity: When compared to FHE over Integers, Brakerski's scheme uses more complex mathematical algorithms and notations as a result of using Ring LWE instead of working with integers. However the removal of bootstrapping technique has reduced decryption function and calculations.

## 4. CONCLUSION

Although Dijk's scheme has succeeded to reduce the complexity of Gentry's original scheme, his scheme has inherited the efficiency limitations of the original scheme in term of noise, length of cipher text and encryption keys, as well the time needed for encryption, decryption and evaluation functions. In the other hand Brakerski's scheme introduced new novel technique for noise management which allowed for evaluating deeper circuits at the same cost as before, this technique is used in later schemes to improve FHE schemes performance. Both schemes have inspired many researchers to search for new mathematical approaches and techniques to improve the performance and efficiency while meeting the security requirements. At the moment the available schemes provide a great potential for cloud computing but they still have lots of scope for improvement and enhancement before they can be ready for practical use in the cloud computing.

## REFERENCES

[1] M. Dijk, C. Gentry, S. Halevi, and V. Vaikuntanathan, —Fully homomorphic encryption over the integers, in EUROCRYPT, 2010, pp. 24–43.

[2] N. Howgrave-Graham, —Approximate integer common divisors, in CaLC, 2001, pp. 51–66.

[3] Z. Brakerski, C. Gentry, and V. Vaikuntanathan, —Fully homomorphic encryption without bootstrapping,Cryptology ePrint Archive, Report 2011/277, 2011.

[4] Vadim Lyubashevsky, Chris Peikert, and Oded Regev, —On ideal lattices and learning with errors over rings, in EUROCRYPT, volume 6110 of Lecture Notes in Computer Science, pages 1–23, 2010.

[5] Oded Regev, —On lattices, learning with errors, random linear codes, and cryptography,in Harold N. Gabow and Ronald Fagin, editors, STOC, pages 84–93. ACM, 2005.

[6] Gu Chunsheng, —Attack on Fully Homomorphic Encryption over the Integers, Cryptology ePrint Archive, Report 2012/157, 2012.

[7] National Institute of Standards and Technology - Computer Security Resource Center - www.csrc.nist.gov.

[8] R. Rivest, L. Adleman, and M. Dertouzos, —On data banks and privacy homomorphisms,in Foundations of Secure Computation. Academic Press, 1978, pp. 169–177.

[9] T. E. Gamal, —A public key cryptosystem and a signature scheme based on discrete logarithms, in CRYPTO, 1984, pp. 10–18.

[10] Gu Chunsheng, —Attack on Fully Homomorphic Encryption over the Integers, Cryptology ePrint Archive, Report 2012/157, 2012.