**INSTRUCTIONS TO ACCELERATE DECENTLY HOMOMORPHIC ENCRYPTION FOR CLOUD EXAMINING**

**ABSTRACT:**

In a Semi-autonomic cloud auditing architecture we weaved in privacy enhancing mechanisms [15] by applying the public key version of the Somewhathomomorphic encryption (SHE) scheme from [4]. It turns out that the performance of the SHE can be significantly improved by carefully deriving relevant crypto parameters from the concrete cloud auditing use cases for which the scheme serves as a privacy enhancing approach. We provide a generic algorithm for finding good SHE parameters with respect to a given use case scenario by analyzing and taking into consideration security, correctness and performance of the scheme. Also, to show the relevance of our proposed algorithms we apply it to two predominant cloud auditing use cases. Index Terms—Somewhat Homomorphic Encryption, Software Acceleration, Cloud Auditing.

|  |  |
| --- | --- |
| **EXISTING SYSTEM** | **PROPOSED SYSTEM** |
| **EXISTING CONCEPT:-**   * In the field of computation on encrypted data ideally one would like to perform both, additively and multiplicatively arithmetic operations on encrypted data in its most flexible way. This is what FHE [7] achieves. * However, for most application fields FHE comes with a non-acceptable performance degradation, both with respect to runtime and with respect to ciphertext versus cleartext data size | **PROPOSED CONCEPT:-**   * We provide a generic algorithm for finding good SHE parameters with respect to a given use case scenario by analyzing and taking into consideration security, correctness and performance of the scheme. * We are investigating the usage of a SHE scheme [4] which we want to carefully adjust to concrete privacy enhancing cloud auditing use cases (UC). |
| **EXISTING ALGORITHM:-**   * Fully homomorphic encryption(FHE) | **PROPOSED ALGORITHM:-**   * Somewhat homomorphic encryption (SHE) |
| **ALGORITHM DEFINITION:-**   * We construct a simple fully homomorphic encryption scheme, using only elementary modular arithmetic. | **ALGORITHM DEFINITION:-**   * SHE parameters with respect to a given use case scenario by analyzing and taking into consideration security, correctness and performance of the scheme. Also, to show the relevance of our proposed algorithms we apply it to two predominant cloud auditing use cases. |
| **DRAWBACKS:-**   * Most application fields FHE comes with a non-acceptable performance degradation | **ADVANTAGES:-**   * High security, correctness and performance |

|  |  |
| --- | --- |
| **PROPOSED SYSTEM** | **FUTURE ENHANCEMENT** |
| **PROPOSED CONCEPT:-**   * We provide a generic algorithm for finding good SHE parameters with respect to a given use case scenario by analyzing and taking into consideration security, correctness and performance of the scheme. * We are investigating the usage of a SHE scheme [4] which we want to carefully adjust to concrete privacy enhancing cloud auditing use cases (UC). | **FUTURE CONCEPT :** -   * Leveled Fully Homomorphic Encryption (LFHE) scheme which is based on the building blocks of the SHE scheme may be a more advanced choice. |
| **PROPOSED ALGORITHM:-**   * Somewhat homomorphic encryption (SHE) | **FUTURE TECHNIQUE:**-   * Homomorphic Encryption (LFHE) |
| **ALGORITHM DEFINITION:-**   * SHE parameters with respect to a given use case scenario by analyzing and taking into consideration security, correctness and performance of the scheme. Also, to show the relevance of our proposed algorithms we apply it to two predominant cloud auditing use cases. | **TECHNIQUE DEFINITION:-**   * A leveled FHE scheme that can evaluate depth-L arithmetic circuits (composed of fan-in 2 gates) using O(λ2) per-gate computation, which is independent of L. Security is based on RLWE for quasipolynomial factors. This construction uses bootstrapping as an optimization. |
| **ADVANTAGES:-**   * High security, correctness and performance | **EXTRAVAGANCE:-**   * Advanced performance |

**SYSTEM REQUIREMENTS**

**HARDWARE**

PROCESSOR : DUALCORE 2 DUO.

RAM : 4GB DD RAM

HARD DISK : 250 GB

**SOFTWARE**

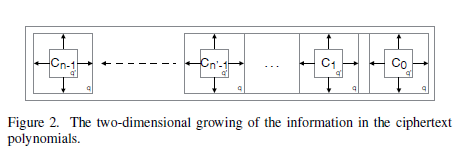
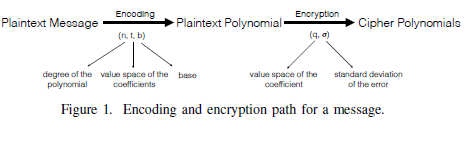
FRONT END : JAVA (J2EE, SERVLETS, JSP)

BACK END : MY SQL

OPERATING SYSTEM : WINDOWS 07

IDE : ECLIPSE

**SYSTEM ARCHITECHER:**

****