Fully Homomorphic Encryption Scheme for Secure Computation

Alisa Gazizullina
Innopolis University, Russia
a.gazizullina@innopolis.ru

Sergey Krendelev
Department of Information Technology, Novosibirsk State University, Novosibirsk, Russia

INTRODUCTION

Cloud Computing requires data stored in third-party servers to be decrypted to allow manipulations over it, thus causing risks of data leakages. Fully Homomorphic Encryption schemes help to address this issue by making computation applicable over ciphertexts. Most of the existent solutions, while providing good data protection require huge computational resources and produce big keys and ciphertexts. In this paper, we propose a new compact FHE scheme with keys and output data well suited for practical use.

KEYWORDS

fully homomorphic encryption, secure computation, modular arithmetic

RELATED WORK

Computational models used in FHE [4] Algorithms can be either based on binary arithmetic or on modular arithmetic.
Papers of Gentry and his followers introduced a scheme based on binary logic [2]. It is difficult to construct an efficient implementation for it. In [7] authors introduce modular arithmetic to FHE algorithm. However, that approach turned out to be non-persistent. The major drawback of the similar approach described in [3] is the exponential growth of data caused by multiplication.
Most of the proposed schemes, while securing computation over data, suffer from not being efficient for a practical use. That defines our goal to develop a new efficient in terms of the memory requirements and complexity of computations FHE scheme.
The proposed algorithm is based on modular arithmetic in the meaning of computer model presented in [5] and thus avoids an increase in data size. Also, it is efficiently implementable on digital machines, as we introduced the notion of multiplication tables for computations over ciphertexts.

PROPOSED CRYPTOSYSTEM

Basics FHE supports arbitrary computation over ciphertexts with no need to decrypt and perform computations over original data. Thus, FHE concerns an encryption algorithm E and a decryption algorithm D, such that $C_1 = E(X_1)$, $C_2 = E(X_2)$ and $D(f(C_1, C_2)) = f(X_1, X_2)$, where C_1 and C_2 are ciphertexts, X_1 and X_2 are plaintexts, f - arbitrary function.
Key Generation The main components of the secret key are modulus M, vector m of k relatively prime moduli, the set of vectors $s_i, \forall i = 1, \cdots, k$, permutation matrix P_C, the number r of vectors of randomly chosen elements. Secret vectors s_i are chosen arbitrarily and are required to be inevitable in $\mathbb{Z}_M$, $j = 1, \cdots, l$.
Encryption The inputs are original message X, set of multiplication rules. X is represented as a vector $(x_1,\cdots,x_l)$, s.t. $X = \sum_{i=1}^{l} x_i mod M$. Then the secret key vectors $(\vec{s_1}, \cdots, \vec{s_k})$ is applied to compute a ciphertext $C = (\vec{c_1}, \cdots, \vec{c_k})$ as $\vec{c_i} = \vec{s_i} \cdot \vec{X}\,(mod\,(m_i))$, for i=1,...,k.
Decryption To restore the ciphertext from permutation we apply P_C matrix first: $C = P_C C$. Then, apply the inverses ${s_i}^{-1}$ of the secret vectors s_i for decryption and use Chinese remainder theorem [1] to find X, that is solve the system of equations of type $\vec{X} = (\,c_i \cdot {s_i}^{-1}\,)\,mod\,m_i$.
Multiplication The multiplication of two ciphertexts leads to the increase of the result's size about l times. To solve this problem we first introduce a set of vectors $\{\zeta_{ij}, \zeta^f\}$ with bases $\{{\zeta_{ij}}^b\}$ to represent entries of ciphertext c_{ij} and c^{rj} as products $c_{ij} \cdot \zeta_{ij}$. Then the public key sent to the server γ, is estimated as $\gamma_{iajb} = ({\zeta_{ia}}^b \cdot S^{-1}(mod m_i)) \cdot ({\zeta_{jb}}^b \cdot S^{-1}(mod m_j))$.

CONCLUSION

In this research, we propose the FHE scheme which is well suited for the efficient implementation on the computer. The use of modular arithmetic prevents overflow involving legitimate computation range. Multiplication tables address the problem of exponential data growth and allow to work with rational numbers, thus increasing the strength of the encryption scheme. Domingo-Ferrer's HE scheme [3] is turned out to be the special case of the scheme proposed in this paper. Both schemes involve random splitting of the original number into small secret values $\in \mathbb{Z}_M$. However, instead of choosing a single modulus m and a vector $\vec{s}$ of invertible values as a secret key, our scheme uses the secret vector of k moduli m_i and a set of k secret vectors s_j (invertible in $\mathbb{Z}_{m_i}$). Thus, our scheme is more secure as it requires a number to be represented as a matrix of values in the rings with the different bases m_i. Our scheme generalizes Domingo-Ferrers' scheme to multivariable functions and extends it to encompass the application of multiplication operations over encrypted data without the growth of the result vector's length.
As a future work, we are planning to integrate RSA public-key cryptosystem with our FHE scheme to enhance security features in RSA for the cloud-based applications. Also, we intent to work in the direction of adapting our algorithm for genomic data encryption, taking into considerations results of [6].

REFERENCES

[1] Sushil Jajodia (eds.) Anne Canteaut Prof. (auth.), Henk C. A. van Tilborg. 2011. *Encyclopedia of Cryptography and Security* (2 ed.). Springer US. http://gen.lib.rus.ec/book/index.php?md5=da0f15e098ac4bb5f05191efbffd57ef
[2] Gentry C. 2009. *Fully homomorphic encryption using ideal lattices.* Vol. 9. 169–178 pages.
[3] J. Domingo-Ferrer. 2002. *A Provably Secure Additive and Multiplicative Privacy Homomorphism.* Vol. 2443. Springer-Verlag, London. 471–483 pages.
[4] Craig Gentry. 2009. *A fully homomorphic encryption scheme.* crypto.stanford.edu/craig.
[5] D. I. Yuditskii. I. Y. Akushskii. 1968. *Machine arithmetic residual classes.* Vol. 36. Soviet Radio, Moscow. 440 pages.
[6] Kristin Lauter, Adriana López-Alt, and Michael Naehrig. 2015. *Private Computation on Encrypted Genomic Data.* Springer International Publishing, Cham. 3–27 pages.
[7] Dertouzos M. L. Rivest R.L., Adleman L. 1978. *On data banks and privacy homomorphisms.* Vol. 4. 169–180 pages.