Enhanced Homomorphic Re-Encryption using Laplacian for Preserving the Privacy in Big Data Analytics

V.Shoba, R.Parameswari

Abstract: ***Big data offers various services like storing sensitive, private data and maintaining the data. Big data users may upload encrypted data rather than raw data for preserving data. Processing and analyzing the encrypted data is the primary target for attackers and hackers. Homomorphic Re-Encryption to supports access control, processed cipher-text on encrypted data and ensure data confidentiality. However, the limitation of Homomorphic Re-Encryption is the single-user system, which means it allows the party that owns a homomorphic decryption key to decrypt processed cipher-texts. Original Homomorphic Re-Encryption cannot support multiple users to access the processed cipher texts flexibly. In this paper, propose a Privacy-Preserving Big Data Processing system which support of a Homomorphic Re-Encryption using laplacian phase that extends partially from a single-group user system by offering cipher text re-encryption that allows accessing processed cipher-texts. Through the cooperation of a Data Provider, to increase the flexibility and security of our system, However apply multiple Services to take in charge of the data from their users and design computing operations over cipher-texts belonging to multiple Service. The analysis completed on proves that our Preserving the Privacy of Big Data Processing method's to performance in terms of security is good on some datasets, inefficiency this also ensures the security and user privacy.***

Key words: ***Big data, Homomorphic Re-Encryption, Paillier, Laplacian, Security, Privacy Preserving.***

I. INTRODUCTION

In this proposed system, the sensitive data must be secured and preserved to store, retrieve and transfer the data. In order to provide security, conventional encryption techniques can be applied. Encrypt each data before storing and on the receiver side. Due to the rapid increase in computing data storage with data mining, the efficiency, and scalability of the data must be retained. Due to its efficiency and scalability, this analytics can be utilized by businesses, government, and individuals by orders of magnitude. Data privacy protection is still a paramount issue in big data applications, as the data owner has no physical control of its data according to the Big Data Security Alliance [1]. To highlight this fact, for example, the sensed body wireless data may contain certain sensitive information like employee identity and location, in addition to non-sensitive values like Name, age, blood group, sex, mail id, phone number, etc.. That can be obtained through cross queries request to recover sensitive data. Thus the scheme called homomorphic encryption computed encrypted data without decrypting the data has been first designed in this state of the art in 1978 by [2]. To address this challenge, in 2009, an improvement is done by Craig Gentry [3] which seems to be an effective solution to address the data privacy protection issue and it widely attracted the IT field and research communities in the data security field like a hot topic. [4] Big Data provides many services as per the demands of the user by reassigning the resources. It reduces the heavy load of users for managing information systems. In Big Data, for storage and processing, users outsource their data to various platforms. Such a platform is usually beyond the control of the users. This gives rise to the necessity to protect the security and privacy of data in Big Data. On the other hand, with the massive development of Big Data, the huge volume of data, [5] are produced and transmitted over the network. Data process, analytics, and mining become important to help a party gain profits not only from their own individual data but also from the data provided by other third parties. Due to security issues, many parties may be unwilling to contract their personal data to the Big Data for analysis. However, some resource-restricted parties have to depend on the Big Data to complete a complex computation Time, in particular, big data analytics and mining. Therefore, preserving the privacy for data processing scheme in Big Data[6], which can adjust to different environments. A challenging solution to preserve data privacy in the Big Data is to store and process data in an encrypted form. Whatever encryption highly complicates data processing and introduces more different challenges and problems. Homomorphic Encryption was proposed to support both computations on encrypted data and assure data confidentiality. However, a drawback of Homomorphic Encryption is it is a single-user system, which means it only allows a single party with an equivalent secret key to access processed cipher texts [7] and cannot support multiparty access. However, in many situations, data are observed and collected all the time for potential use without knowing a concrete aggregator or a data access requester. For example, hospitals and medical research can benefit greatly from the statistics of patients and more than one party could be interested in requesting encrypted processing results after data collection and process. Original Homomorphic Encryption cannot support multiple users to access the processed cipher texts.

Revised Manuscript Received on December 5, 2019.
***** Correspondence Author
V.Shoba*, Research Scholar, Department of Computer Science, Vels Institute of Science Technology & Advanced Studies, Chennai 600 117, India. Email: sho13velfam@gmail.com
Dr.R.Parameswari, Associate Professor, Department of Computer Science, Vels Institute of Science Technology & Advanced Studies, Chennai 600 117, India. Email:dr.r.parameswari16@gmail.com

In order to achieve secure data and privacy-preserving process at the Big Data have been designed with various techniques. However, none of them can overcome the problems. a scheme [8] based on Paillier's cryptosystem [7] was proposed to achieve computations over cipher texts outsourced by a number of users with their own keys. However, it can only support multiplication and addition. It Contains Enhanced Homomorphic Re-Encryption Based Paillier Schemes, extends HRES from single-user encryption to multi-user Re-encryption. To support flexible access to cipher text processing results, we design a new cryptographic primitive in Enhanced Homomorphic Re-Encryption Based Pallier Schemes. It employs two service providers to manage encrypted data and flexibly support Group access control on cipher text processing results with two-level Encryption, thus successfully achieves re-encryption over the data encrypted with homomorphism. Only authorized users can access the cipher text processing result in a secure way.

In this paper, It will propose the contribution by introducing our Enhanced Homomorphic Re Encryption Based Paillier and Differential privacy preserving using Laplacian operation scheme that first addresses the existing data privacy weaknesses by efficiently carry out operations over cipher texts without passing the private key and public key as the model of any key's parameters [9] to untrussed server. And secondly, we show how effectively our scheme retrieves cipher texts at the reduce phase by an efficient and secure retrieval algorithm. Then in the rest of this paper, that will introduce the scheme and cryptographic primitives used in this paper in section 2. A brief discussion on preceding related work is done in section 3. Problem formulation and section 4 will the proposed scheme for Experiment Configuration. Ultimately conclude this paper in section 5.

II. RELATED WORK

The rapid development of Big Data [10] more and more users decide to outsource their data to the Big Data for warehousing and further outgrowth. [5] The homomorphic encryption phase is semantically secure, like RSA, also have attacks on one way. However, the risk of the data being revealed or disclosed force is vital to improve the protection and privacy of user data. In the novel, It can find many kinds of research with respect to preserving the privacy of data process, which is concisely reviewed [9] was understood as the FHE, which was supported on a common homomorphism method with the encryption function [11] Gentry's greater performance produced the restriction of this approach was that the time complexity was too high to be applicable and the timing measurements completed by other scholars also verify its inefficiency. It was reported that encrypting a bit plain-text Big Data constructs a huge amount of bits for cipher-texts, could extend millions of bits in some situations [15]. The estimated workload Big Data rapidly enlarge as the amount of evaluation trading operations was combined with the number of variables active in the calculation[13], [14]. In addition, our tensor-supported FHE was separate from the lattice-based homomorphic approach [12]developed a tensor-based FHE that applied a data holder-centric mechanism and empower all data to be encrypted in the primary Big Data while supporting accurate manipulations. [16] Reconsider the Paillier's cryptosystem and keep privacy-preserving data accumulate by separating its decryption key into two abilities and sharing them with action and social media. But this scheme cannot support multiparty Group access to encrypted data processing results. [17] intend an efficient outsourcing multiparty computation framework under Random keys based on additive homomorphic encryption [18]. However, this scheme can only support addition and multiplication, but no other operations. Based on the same cryptosystem intend as an appropriate splitting the secret keys of PHE into the separate portion, [19] design multiple calculations over outsourced data under different keys. However, homogenetic to the performance in the system can only maintain the small number of input data.

III. PROBLEM FORMULATION

A. System Model

In this work, To focus on Re-encrypted data processing under Big Data. As you know, in the normal Big Data environment, the Data provider can access the data that is in the Big Data at any time, this poses serious privacy concerns. To protect data in terms of confidentiality and privacy from unauthorized Multi-users, we propose a practical scheme for encrypted data processing under Big Data.

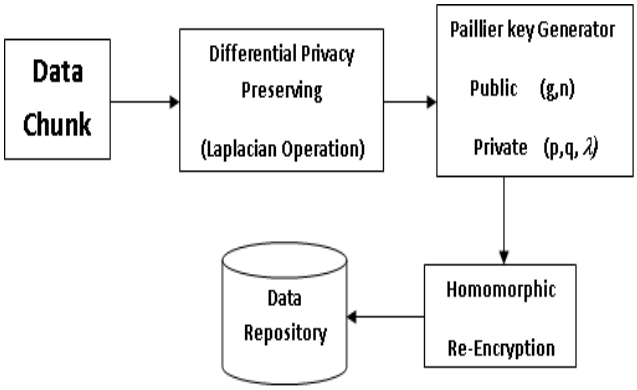

Fig.1 Enhanced Homomorphic Re Encryption with Laplacian Operation

Data Chunk in the Data Provider and Multi-User entities mainly store the encrypted data for homomorphic Re Encryption computation service. We can use private and public keys for encrypted data. We can think of it as storage with computing capabilities of Fig1.

B.Design Goals

To achieve an efficient security system which fulfils the fore mentioned scenario and privacy requirements, our scheme should consider the security and performance overhead, which mainly consider the computation time. To achieve the preserving system is the privacy of the Big Data system, it must assure the correctness of the standard prototype. The main objective of this system is scalability and to act faster computation time. In the already existing work, the Encryption time takes Micro seconds, but in the proposed work to more secure and encryption time is achieved in better Micro Seconds for a large scale of data.

C. Laplacian Algorithm

1. Def Pre-processed data in laplacian (data)
2. Apply the mark using 3X3 matrix mask=[[0,1,0],[1,-4,1],[0,1,0]]
3. print(mask[0])
4. for i in range(len(data)-3)
5. for j in range(len(mask))
6. for k in range(len(mask[0]))
7. data[i+j][k]=int(data[i+j][k])-mask[j][k]
8. print(data)

D. Enhanced Homomorphic Re Encryption Based Paillier Algorithm

1. Parameters
2. prime numbers p, q
   $n = pq$
3. $\lambda = \text{lcm}(p-1, q-1)$ (λ is a Carmichael's Function)
4. g, with $g \in Z^*_{n^2}$ and the order of g is a multiple of n (since g=(1+n) this is a best choice).
5. Public key
   n, g
6. Private key
   p, q, λ
7. Encryption plaintext
   $m < n$
8. select a random $r < n$ such that $r \in Z^*_n$
9. cipher text
   $c = g^m r^n \bmod n^2$
10. For every data in D repeat the same encryption once
11. Decryption cipher text
    $c < n^2$
12. plaintext
    $m = \dfrac{L(c^\lambda \bmod n^2)}{L(g^\lambda \bmod n^2)} * \bmod n$
13. For every data in D repeat the same decryption once Following the notation in [8], L(u) = u−1 n , for u = 1 mod n. This function is only used on input values u that actually satisfy u = 1 mod n.

IV. THE PROPOSED SCHEME

A. Experiment Configuration

The experimental setup configuration with two major objectives of the evaluation, which were useful for measuring the correctness of the data retrievals and assessing the execution time while processing various sizes of data has been addressed. The evaluation was completed in our laboratory environment. The simulator of the homomorphic Re-encryptions was running on Python. The implemented our Enhanced Homomorphic Re Encryption approach and handled different input data sizes to assess two major objectives, the correctness and execution time. That presented partial results that were collected from Different groups of records in data set to demonstrate the correctness performance of our approach. The data sets were based on different sizes of input data, which were 1000, 2000, 3000, 4000 and 5000 till 10000 records. However, the correctness was measured by calculating the results differences between the quality from plain texts and that of cipher texts.

The experimental results experience that our access was noise-free and also charged the performance time of expanding polynomials for encrypting merge trading operations results. The evaluations were supported on multiple groups of Records measurement while various input data sizes were applied. The comparisons necessity to Encryption Time for a different group of data set. Due to the limited space, that illustrated a few experimental results based on our experimental configuration Next section displayed.

B. Experiment Results

In this section, we exhibited partial results that were collected from our experiments following the experimental configuration enumerate in displayed partial accuracy testing results for homomorphic Re-Encryption and Enhanced Homomorphic Re Encryption for Paillier additions or multiplications with Laplacian noise Filter. Among them, represented the results obtained from input data.

Table-I: Computation Time Between Homomorphic Re Encryption and Enhanced Homomorphic Re Encryption

S.No	Datasets	Different Range of Records	Computation Time	
			Homomorphic Re Encryption (ms)	Enhanced Homomorphic Re Encryption (ms)
1	Medical	1000	29.050	26.526
2	Hospital	2000	59.772	51.196
3	Diabetes	3000	95.583	76.915
4	Credit Card	4000	129.913	104.417
5	B-Customer	5000	157.788	130.375
6	Adult	6000	199.662	158.244
7	Salt	7000	226.915	186.069
8	Email	8000	250.965	211.848

9	Healthcare	9000	309.503	243.887
10	E-Commerce	10000	325.626	271.615

C. Efficiency of Laplacian with Re encryption

The computational overhead imposed on the Group of data achieves a transformation to Laplacian with Re Encrypted data, such as Adding and multiplying data instances to a fully random orthogonal matrix. In addition, the Laplacian with homomorphic Re encryption technique is only extended out in data users. Evaluate the scalability of encryption by measuring the consuming time of encrypting huge-scale asyndetic datasets. The asyndetic data sets include 1000-10,000 instances with 1000, 2000,3000 and 5000 variables particularly, and the results in time of performing encryption are shown in Table.1 with metabolism preserving Privacy data and encryption. In the proof, encrypting 10,000 instances with 1000 Records takes only Micro Seconds to complete the Enhanced Homomorphic Re encryption operation and also for this biggest dataset, It takes no more than 271.615 Microseconds to complete the whole privacy preservation process. The consuming time increases with generally linear growth in the numbers of instances. Therefore, since the overhead imposed on the Data Chunk in Data Provider, Data user time is very low from existing time , the proposed Re- Encryption is appropriate for big data.

V.CONCLUSION

In this paper, a new approach which reduces the noise and thus much better parameters than previous Homomorphic Re Encryption schemes for the same security level has been identified. Our secure search engine application runs in a few Microseconds on small to large size of Data set. The adaptability performance was assessed by experimental evaluations from the perspectives of Time accuracy. Its finding depicted that the Enhanced Homomorphic Re-encryption approach could successfully acquire correct outputs from decrypting cipher-results of blend operations.

REFERENCES

1. T. Hayashi, S. Moriai, L. Wang, Y. Aono, and L. T. Phong, "Privacy-Preserving Deep Learning via Additively Homomorphic Encryption," IEEE Transactions on Information Forensics and Security, vol. 13, no. 5, pp. 1333–1345, 2017.
2. J. Bird and L. Eleftheriou, "Homomorphic Encryption Survey Paper," pp. 1–7, 2018.
3. S. Choinyambuu, "Homomorphic Tallying with Paillier Cryptosystem," pp. 1–10, 2009.
4. C. S. Gu, "Fully homomorphic encryption from approximate ideal lattices," Ruan Jian Xue Bao/Journal of Software, vol. 26, no. 10, pp. 2696–2719, 2015.
5. D. Bogdanov, R. Talviste, and J. Willemson, "Deploying secure multi-party computation for financial data analysis," Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 7397 LNCS, no. 8124, pp. 57–64, 2012.
6. W. Ding, Z. Yan, and R. H. Deng, "Encrypted Data Processing with Homomorphic PT US CR," Information Sciences, 2017.
7. M. Barbosa, D. Catalano, and D. Fiore, "Labeled Homomorphic Encryption : Scalable and Privacy-Preserving Processing of Outsourced Data," pp. 1–28.
8. C. Guo, X. Tang, Y. Chen, P. Tian, and M. Z. Alam Bhuiyan, "An Efficient Distributed Approach on High Dimensional Data Similarity Searchable Encryption," Proceedings - 17th IEEE International Conference on Trust, Security and Privacy in Computing and Communications and 12th IEEE International Conference on Big Data Science and Engineering, Trustcom/BigDataSE 2018, pp. 1270–1275, 2018.
9. D. Chattaraj, M. Sarma, A. K. Das, N. Kumar, J. J. P. C. Rodrigues, and Y. Park, "HEAP: An Efficient and Fault-Tolerant Authentication and Key Exchange Protocol for Hadoop-Assisted Big Data Platform," IEEE Access, vol. 6, pp. 75342–75382, 2018.
10. Q. Wang, D. Zhou, and Y. Li, "Secure Outsourced Calculations with Homomorphic Encryption," Advanced Computing: An International Journal, vol. 9, no. 6, pp. 01–14, 2018.
11. M. Tebaa, S. El Hajji, and A. El Ghazi, "Homomorphic encryption method applied to cloud computing," Proceedings of the 2nd National Days of Network Security and Systems, JNS2 2012, vol. 4, no. 15, pp. 86–89, 2012.
12. L. Mohan and S. E. M, "Secure and Privacy Preserving Mail Servers using Modified Homomorphic Encryption (MHE) Scheme A Technique for Privacy Preserving Big Data Search," vol. 9, no. 3, pp. 101–110, 2018.
13. K. Gai and M. Qiu, "Blend arithmetic operations on tensor-based fully homomorphic encryption over real numbers," IEEE Transactions on Industrial Informatics, vol. 14, no. 8, pp. 3590–3598, 2018.
14. Y. Lu and M. Zhu, "Privacy preserving distributed optimization using homomorphic encryption," Automatica, vol. 96, pp. 314–325, 2018.
15. I. You, K. Yim, and L. Barolli, Advances in network-based information systems, vol. 8, no. 3. Springer International Publishing, 2012.
16. "Shai Halevi — IBM CRYPTO 2011 Wouldn ' t it be nice to be able to …," 2011.
17. G. Ateniese, K. Fu, M. Green, and S. Hohenberger, "Improved proxy re-encryption schemes with applications to secure distributed storage," ACM Transactions on Information and System Security, vol. 9, no. 1, pp. 1–30, 2006.
18. Q. Wang, D. Zhou, and Y. Li, "S ECURE O UTSOURCED C ALCULATIONS W ITH," vol. 9, no. 6, pp. 1–14, 2018.
19. F. Benhamouda, M. Joye, and B. Libert, "A New Framework for Privacy-Preserving Aggregation of Time-Series Data," ACM Transactions on Information and System Security, vol. 18, no. 3, pp. 1–21, 2016.
20. G. Wang, C. Liu, Y. Dong, K. K. R. Choo, P. Han, H. Pan, and B. Fang, "Leakage Models and Inference Attacks on Searchable Encryption for Cyber-Physical Social Systems," IEEE Access, vol. 6, pp. 21828–21839, 2018.

AUTHORS PROFILE

V.Shoba received the M.Sc. M.Phil. Degree in Computer Science from Madurai Kamaraj University, Madurai. She is currently pursuing Ph.D. degree in Vels Institute of Science, Technology and Advanced Studies, Chennai, India. Her Current Research include Big Data Analytics, Cryptography, Information Security and Privacy Preserving Big Data and dependable Computing. She has authored three papers in International Journals and Conferences in the above-mentioned areas.

Dr.R.Parameswari is working as Associate Professor in Department of Computer Science, Vels Institute of Science, Technology and Advanced Studies, Chennai. She has 13 year of teaching Experience. She has Completed PhD in Computer Science from St. Peter's University, Chennai. She is presently guiding Eight Ph.D Scholars and One M.Phil Scholar. She has produced three M.Phil Scholar s. She has published 26 papers in varies International Journals indexed in Scopus. She has presented many papers in International Conferences and attached many seminars workshops conducted by various educational Institutions. She is acting as editor and reviewer in many International Journals. Her research interest lies in the area of Cloud Computing Big Data Analytics, Internet of Things.