SECURE HOMOMORPHIC ENCRYPTION WITH DYNAMIC KEY MANAGEMENT AND ADAPTIVE CASE MANAGEMENT

[1]RAHUL AV, [2]PRIYANKA S, [3]POOJASHRI P

Department of Computer Science Engineering, Chennai Institute of Technology Chennai, India
E-mail: [1]rahul20021002@gmail.com, [2]priyankasekarjaya@gmail.com, [3]poojashri12004@gmail.com

Abstract - In a period of data-driven products, securing confidential information is our utmost priority.This study explores the integration of isomorphic encryption with adaptive case management (ACM) and dynamic key management, aiming to improve the secure processing of sensitive data in a dynamic business environment while preserving adaptability on. The research method requires careful implementation of a bespoke homomorphic encryption scheme in the ACM system, combined with dynamic key management strategies for enhanced security posture To show what is useful, we use a specific application, such as healthcare records management, serves as a real-world scenario. Preliminary findings show promising results across all primary outcomes, including assessments of performance, safety complexity, and user experience. This paper provides valuable insights into the interface between cryptographic technology and business process management. By combining isomorphic encryption with ACM and dynamic key management, we propose a new paradigm for secure and customizable data processing in a dynamic business environment. Our findings not only address current challenges, but also pave the way for future developments in the secure use of critical information. As businesses continue to grapple with evolving privacy trends and regulatory issues, the proposed framework provides a practical solution for organizations looking to balance security, flexibility and efficiency between This study serves as a catalyst for further research, encourage the integration of Advanced cryptography techniques in dynamic business processes to meet the growing demand for secure and flexible information management.

Keywords - Cryptographic Techniques, Computational Flexibility, Adaptive Framework, Information Privacy, Dynamic Key Management, Adaptive Case Management (ACM),Performance Metrics, Key Generation, Keyrotation, Throughput, Privacy Solutions

I. INTRODUCTION

In today's dynamic information technology environment, the inevitable challenge of seamlessly blending security and rigor within complex business processes has been a priority for associations. This exploration explores the intricate interplay between three vital rudiments homomorphic encryption, Adaptive Case Management(ACM), and dynamic vital operation, presenting a new approach to review secure data processing in the ever- evolving ultramodern business terrain. Homomorphic encryption, a revolutionary cryptographic fashion, serves as a lamp for enhanced data sequestration by enabling calculations on paraphrased data without decryption, converting the conventional trade- off between data security and computational harshness. Completing homomorphic encryption, ACM emerges as a foundation in clustering security and harshness. Unlike rigid, rule- rested approaches, ACM provides a flexible and knowledge worker- centric frame suitable of orchestrating complex business processes characterized by fluidity, query, and the need for mortal intervention. This adaptive approach aligns seamlessly with the dynamic nature of ultramodern associations, empowering them to navigate through cases and processes that warrant predefined structures. At the core of this integration, dynamic vital operation serves as the linchpin, introducing responsive mechanisms for the generation, gyration, and distribution of encryption keys. This not only fortifies the security structure but also enhances the harshness of the entire system, allowing for impeccable updates and acclimatizations without dismembering ongoing processes. The provocation for this exploration extends beyond immediate challenges; it anticipates unborn conditions. The fusion of homographic encryption, adaptive Case Management and Dynamic key management plays a crucial role in future proofing data processing systems. In an period marked by data breaches, heightened sequestration enterprises, and an ever-changing nonsupervisory terrain, associations are seeking innovative results that transcend conventional approaches. Conventional approaches constantly struggle to accommodate the fluid and changeable nature of contemporary business processes, leading to lodgment in either security or harshness. By synthesizing homomorphic encryption, ACM, and dynamic vital operation, this exploration seeks to offer a holistic and forward- looking approach to data processing, transcending these limitations. The primary ideal is to construct a secure and adaptable system suitable of managing sensitive data within dynamic business processes. Through scrupulous disquisition of the individual principles and complications of homomorphic encryption, ACM, and dynamic vital operation, the exploration aims to unravel the complications of each element and understand how they can be uniquely integrated into a unified system. The practical connection, effectiveness, and advantages of this intertwined system will be demonstrated through a focused use case, similar as healthcare records operation, showcasing its distinctive real- world impact. This

exploration aspires to be a guiding compass for associations seeking to marshal in a new period of uniquely secure, adaptive, and sequestration-conserving data processing within the intricate shade of dynamic business surroundings.

II. RELATED WORKS

[2] Manish M. Potey, Chandrashekhar A. Dhote, and Deepak H. Sharma pioneer a revolutionary solution for cloud computing challenges, focusing on robust data security. Traditional cloud storage raises concerns, prompting users to devise encryption methods. However, decryption for processing remains a drawback. Guided by Potey, Dhote, and Sharma, this research introduces a groundbreaking technique—utilizing fully homomorphic encryption to securely store data in DynamoDB within AWS. Computation on encrypted data occurs in the public cloud, ensuring user data is never stored in plaintext. This innovation, under their guidance, mitigates critical security and privacy issues in cloud computing, providing a paradigm shift.

[3] Hariss, Khalil, Hassan Noura, and Abed Ellatif Samhat lead pioneering advancements in practical Homomorphic Encryption (HE) algorithms crucial for modern applications like Cloud Computing. Current solutions face limitations in computation overhead or vulnerability to attacks. Collaboratively, they introduce the Matrix Operation for Randomization and Encryption (MORE) approach, addressing these challenges. The algorithm, meticulously detailed and rigorously evaluated, proves effective against strong attacks without compromising system performance. This research marks a significant leap forward in achieving practical and secure Homomorphic Encryption for real-world applications, ensuring data privacy in shared storage and processing resources like Cloud Computing.

[7] Sun, Xiaoqiang, et al advance machine learning classification with a privacy focus, proposing an enhanced fully homomorphic encryption (FHE) scheme for trend prediction in big data analysis. Built on Helevi's FHE library, their scheme employs relinearization and modulus switching to optimize performance, reducing ciphertext size and decryption noise. By eliminating unnecessary techniques, the researchers enhance efficiency. The scheme's versatility is demonstrated through homomorphic comparison protocols, private hyperplane decision-based, and private Naïve Bayes classification. Efficient homomorphic comparison protocol interactions are achieved. Experimental simulations affirm the FHE scheme's efficacy in private machine learning, particularly in decision tree classification, showcasing substantial contributions to privacy-preserving machine learning.

[11] Amidst rising security concerns in big data and cloud computing, Mohamed Alloghani addresses the need for extended data encryption. Focusing on public cloud services, where ensuring confidentiality is complex, Alloghani explores homomorphic encryption as a promising solution for secure data manipulation. Employing the PRISMA checklist and Cochrane's Quality Assessment, this paper systematically reviews related research, revealing a predominant emphasis on security and the endorsement of homomorphic encryption. However, thematic analysis exposes additional concerns, with 38% of articles, including Alloghani's, falling short on checklist criteria. The study underscores the potential of homomorphic encryption in tackling security challenges in the realm of big data and cloud computing.

[14] Alaya, Bechir, Lamri Laouamer, and Nihel Msilini spearhead a study fortifying system security, with a focus on homomorphic encryption. This research delves into established cryptosystems enriched with supplementary techniques to enhance performance and privacy ratios. Emphasizing homomorphic encryption's adaptability, the survey explores its manifold advantages and exceptional performance across diverse fields. Alaya and team's impactful contributions shape the research landscape, providing a comprehensive comparison of adopted techniques. Their work advances the understanding of innovative encryption methodologies and their applications in various domains, contributing significantly to the fortification of system security.

[15] Chillotti, Ilaria, et al., lead a groundbreaking effort introducing a swift fully homomorphic encryption scheme over the torus (TFHE), refining GSW-based FHE and its ring variants. Focusing on bootstrapped binary gates, they achieve a significant reduction in running time to 13 ms single core, using a smaller bootstrapping key (16MB instead of 1 GB). Introducing methods for manipulating packed data in leveled homomorphic mode optimizes ciphertext expansion and function evaluation. The paper presents novel contributions, including homomorphic counter TBSR and efficient leveled evaluation of weighted automata. A circuit bootstrapping approach swiftly converts LWE ciphertexts, enhancing arithmetic function efficiency. This work signifies notable advancements in fully homomorphic encryption with practical LWE-based schemes and concrete parameter sets.

[19] Wood, Alexander, Kayvan Najarian, and Delaram Kahrobaei lead pioneering research at the crossroads of machine learning, privacy preservation, and bioinformatics, leveraging fully homomorphic encryption (FHE). Addressing ethical and privacy challenges in medical and genomic data sharing, their work explores FHE's potential for secure evaluation over encrypted data. Successfully implementing machine learning models like deep learning and decision trees, they showcase FHE's efficacy in privacy-preserving applications for medical data, including encrypted data classification and model

training. The survey comprehensively outlines FHE's concepts, history, open-source implementations, and its pivotal role in privacy-preserving techniques in machine learning and bioinformatics.

III. PROPOSED WORK

In the realm of escalating data challenges, our proposed system aims to redefine data processing paradigms by seamlessly integrating three core components: Homomorphic Encryption, Adaptive Case Management (ACM), and Dynamic Key Management. Homomorphic Encryption, a cornerstone of our approach, enables computations on encrypted data without the need for decryption, ensuring end-to-end data confidentiality. Complementing this, ACM injects adaptability into workflows, empowering knowledge workers to make informed decisions in response to dynamic scenarios. Adding another layer of resilience, Dynamic Key Management offers flexibility by dynamically generating and updating encryption keys during runtime.

This amalgamation aspires to establish a secure and adaptive framework, marking a paradigm shift in contemporary data processing. The subsequent sections will intricately explore each component, delineating their distinct roles, potential benefits, and highlighting the seamless integration that addresses the multifaceted challenges inherent in the rapidly evolving landscape of data processing.

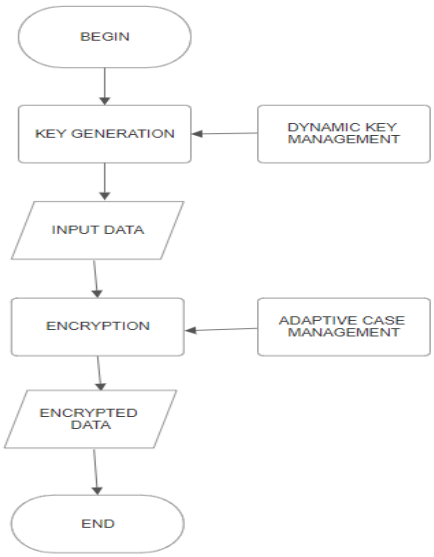

Figure 1, System Architecture

3.1 Homomorphic Encryption:

Homomorphic encryption represents a breakthrough in the field of cryptography by allowing computations on encrypted data, preserving privacy in various applications. It involves the encryption of data with a public key, and mathematical operations can be conducted on the encrypted data directly. The encrypted result retains the same mathematical relationship as if the operations were performed on the unencrypted data. Only the holder of the corresponding private key can decrypt the final result. This innovation has transformative implications for secure data processing, especially in cloud computing scenarios where sensitive information can be processed without exposing the raw data. As advancements continue, homomorphic encryption remains at the forefront of privacy-centric technologies, ensuring secure and confidential data handling in an increasingly interconnected and data-driven world.$r_k=\{ s\circ \Pi r_i, s_1 , CRT(j_{lk} s_k)\}$

where r and s are two large prime numbers

$$a = \sum_{i=1}^{k} c_i d_i + \sum_{j} z_j \ mod \ z_0$$

$$c_i = (a \ mod \ r_i) \ mod \ s_i$$

Where
c: public key
d: private key
r: plain text
s: cipher text
z:error

To implement homomorphic encryption, start by generating a pair of cryptographic keys—a public key for encryption and a private key for decryption. Ensure that the data to be encrypted meets the specific requirements of the chosen homomorphic encryption scheme. Utilize the public key to encrypt the prepared data, transforming it into an encrypted form while preserving its mathematical properties. Transmit the encrypted data to a server or processing unit capable of handling homomorphic operations, or perform these operations locally if applicable. Conduct mathematical operations, such as addition or multiplication, directly on the encrypted data, ensuring compatibility with the chosen homomorphic encryption scheme. Obtain the result of these homomorphic operations, which remains encrypted at this stage. If necessary, decrypt the final result using the private key; however, note that not all applications require decryption, as subsequent computations can be performed on the encrypted data. Finally, obtain the final output of the computations in a format suitable for the intended application or analysis. This process showcases the versatility of homomorphic encryption in performing secure computations on encrypted data without compromising its confidentiality.

3.2 Adaptive Case Management:

Adaptive Case Management (ACM) is an innovative approach to handling knowledge-intensive work that

thrives in dynamic and unstructured environments. It diverges from traditional, rigid workflows by embracing flexibility and adaptability. At its core, ACM revolves around the concept of a "case," representing a unique piece of work. Unlike predefined processes, ACM allows for the dynamic evolution of workflows based on real-time considerations and the expertise of knowledge workers.

In the ACM framework, decision-making is a collaborative and adaptive process. Knowledge workers, equipped with valuable insights, have the autonomy to make informed decisions and adjust processes as needed. The emphasis is on achieving goals rather than adhering strictly to predefined steps, fostering creativity and adaptability.

Collaboration and communication are integral to ACM, with systems providing tools for real-time interactions, discussions, and document sharing among team members. Information integration is another key feature, enabling a holistic view of a case by integrating structured data, unstructured documents, and external information.

$$\mathrm{ACM(k)} = \mathrm{C}\left(\mathrm{D_p(k)}\right)$$

k: Encrypted process data (homomorphically encrypted.

C: Decision logic function operating on encrypted data.

D_p : Homomorphic encryption function with key p

ACM: Adaptive Case Management process.

Commence by initializing a new case, providing details on the context, objectives, and initial information, marking the beginning of a knowledge-intensive task or project. Develop a digital representation of the case, encompassing crucial details like documents, data, and any pertinent information linked to the case. Identify and allocate knowledgeable individuals to the case, entrusting them with decision-making responsibilities to propel the case forward. Facilitate the dynamic creation and adjustment of workflows, allowing them to evolve based on the changing nature of the case and decisions made by knowledge workers.Foster collaboration among knowledge workers by providing tools for real-time communication, document sharing, and discussions to enhance collective decision-making.Grant knowledge workers the authority to make informed decisions throughout the case's lifecycle, with decision-making adapting to the evolving context of the case.

3.3 Dynamic Key Management:

Dynamic Key Management (DKM) is a sophisticated cryptographic approach ensuring adaptability and security in encryption key management. Encryption keys are pivotal in protecting sensitive information, and DKM addresses evolving security threats and privacy regulations. It involves continuous key generation, distribution, and rotation during runtime, allowing cryptographic systems to dynamically respond to emerging threats.

pe_i : the secret key at time i

qe_i : the corresponding public key at time i

KG: key generation algorithm

KU: key update algorithm

$$(pe_0, qe_0) = KG(\ \ )$$

$$(pe_{i+1}, qe_{i+1}) = KU(pe_i, qe_i)$$

Dynamically generate encryption keys based on predefined policies and security requirements.Initialize the system with the generated keys to establish a secure foundation. Implement a routine key rotation process to replace old keys with new ones.Set up policies to dictate when key rotation should occur, ensuring timely updates. Develop and implement policies that govern key management procedures.Define criteria for key updates, rotations, and other key-related operations. Ensure that key updates are seamlessly integrated into ongoing cryptographic operations.Minimize downtime or interruptions during the key update process.Establish secure mechanisms for distributing updated keys to relevant parties. Use secure communication channels and protocols to protect the integrity of key distribution. Implement versioning mechanisms to track different iterations of encryption keys. Maintain comprehensive audit logs that record key updates, rotations, and associated events.

IV. PERFORMANCE ANALYSIS

4.1. THROUGHPUT:

Throughput stands as a pivotal metric in the evaluation of cryptographic system performance, gauging the effectiveness of cryptographic operations over a given time span. The formula defining throughput is the ratio of total cryptographic operations to the corresponding time duration, expressed as:

$$\text{Throughput} = \frac{\text{Total Cryptographic Operations}}{\text{Time Taken}}$$

This analytical approach proves crucial in the identification of potential bottlenecks within the system, originating from constraints such as processing limitations, memory constraints, or network capabilities. Continuous monitoring of throughput trends offers valuable insights, aiding in the identification of optimization opportunities. This, in turn, guides enhancements in algorithms, system configurations, or hardware to bolster overall system efficiency. Furthermore, the analysis of throughput provides a comprehensive understanding of resource utilization during cryptographic operations. This knowledge facilitates the judicious allocation of resources, ensuring optimal system performance. This

method of analysis offers a holistic perspective on the system's performance characteristics without the need for specific subheadings.

4.2 EXECUTION TIME:

Performance analysis for execution time in cryptographic systems is crucial for evaluating the efficiency of cryptographic operations. The primary metric for this analysis is the Execution Time Overhead (ETO), calculated as the difference between the time taken for cryptographic operations and the baseline time without encryption:
Execution Time Overhead (ETO) = Time taken for Cryptographic Operations - Baseline Time Without Encryption

The formula quantifies the additional time introduced by cryptographic processes, providing insights into the impact on system performance. Minimizing ETO is essential for ensuring that cryptographic operations do not significantly impede the overall execution time of applications. Factors influencing ETO include the choice of encryption algorithms, key lengths, and the complexity of cryptographic operations. Efficient key management practices and algorithm selection contribute to mitigating execution time overhead.

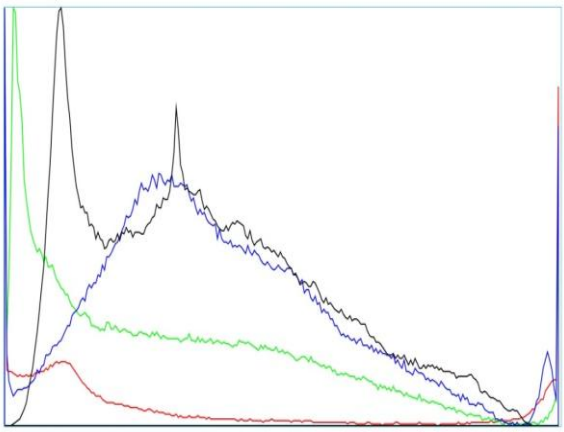

Figure 2 , Comparison graph on execution time

From 2 the figure , four distinct homomorphic encryption schemes were evaluated based on their execution times for cryptographic operations. The efficiency of each scheme was assessed, with lower execution times indicating enhanced performance. The evaluation considered key factors such as encryption and decryption speed, computational complexity, noise management, and the supported set of operations. The graph presents a visual representation of the comparative performance, highlighting the strengths and weaknesses of each homomorphic encryption scheme. Notably, schemes exhibiting lower execution times demonstrate superior efficiency, making them favorable choices for scenarios where quick computations on encrypted data are paramount.

V. CONCLUSION AND FUTURE DIRECTION

The fusion of homomorphic encryption, adaptive case management (ACM), and dynamic key management (DKM) forms a robust framework for safeguarding sensitive data in a dynamic ecosystem. Homomorphic encryption ensures data confidentiality, ACM adds flexibility to business processes, and DKM enables secure key management. This amalgamation establishes a foundation for secure, privacy-preserving computations.
This approach tackles challenges related to data privacy and secure collaboration. Integrating homomorphic encryption into ACM enables computations on encrypted data, facilitating confidential data analytics. DKM adds flexibility by dynamically managing encryption keys to meet evolving security requirements without sacrificing performance.Looking ahead, quantum-safe homomorphic encryption is crucial to resist quantum attacks. Enhancements in ACM may involve integrating machine learning and AI for automated decision-making, improving case management efficiency. Exploring real-time DKM strategies allows instantaneous key updates for heightened security without compromising performance.Addressing scalability challenges in homomorphic encryption is essential for large-scale systems. Establishing industry standards for homomorphic encryption, ACM, and DKM promotes interoperability, fostering widespread adoption. Additionally, user-friendly interfaces and tools simplify integration, making these technologies accessible across industries.In conclusion, the convergence of homomorphic encryption, ACM, and DKM holds immense potential for secure, private, and efficient data management. Continuous research in these areas will contribute to advancing privacy-preserving technologies, seamlessly integrating them into diverse applications and industries.

REFERENCES

[1] Fragkiadakis, Alexandros, et al. "A practical implementation of an adaptive compressive sensing encryption scheme." 2016 IEEE 17th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM). IEEE, 2016.

[2] Mr, Manish M. Potey, Chandrashekhar A. Dhote, and Deepak H. Sharma Mr. "Homomorphic encryption for security of cloud data." Procedia Computer Science 79 (2016): 175-181.

[3] Hariss, Khalil, Hassan Noura, and Abed Ellatif Samhat. "Fully Enhanced Homomorphic Encryption algorithm of MORE approach for real world applications." Journal of Information Security and Applications 34 (2017): 233-242.

[4] Acar, Abbas, et al. "A survey on homomorphic encryption schemes: Theory and implementation." ACM Computing Surveys (Csur) 51.4 (2018): 1-35.

[5] Chen, Hao, et al. "Logistic regression over encrypted data from fully homomorphic encryption." BMC medical genomics 11 (2018): 3-12.

[6] Kim, Miran, et al. "Secure logistic regression based on homomorphic encryption: Design and evaluation." JMIR medical informatics 6.2 (2018): e8805.

[7] Sun, Xiaoqiang, et al. "Private machine learning classification based on fully homomorphic encryption." IEEE Transactions on Emerging Topics in Computing 8.2 (2018): 352-364.

[8] Wu, D. N., Q. Q. Gan, and X. M. Wang. "Verifiable public key encryption with keyword search based on homomorphic encryption in multi-user setting." IEEE Access 6 (2018): 42445-42453.

[9] Xu, Jian, et al. "Dynamic fully homomorphic encryption-based merkle tree for lightweight streaming authenticated data structures." Journal of Network and Computer Applications 107 (2018): 113-124.

[10] Yagoub, Mohammed Amine, et al. "An adaptive and efficient fully homomorphic encryption technique." Proceedings of the 2nd International Conference on Future Networks and Distributed Systems. 2018.

[11] Alloghani, Mohamed, et al. "A systematic review on the status and progress of homomorphic encryption technologies." Journal of Information Security and Applications 48 (2019): 102362.

[12] Geng, Yang. "Homomorphic encryption technology for cloud computing." Procedia Computer Science 154 (2019): 73-83.

[13] Rajan, D. Palanivel, S. John Alexis, and S. Gunasekaran. "Dynamic multi-keyword based search algorithm using modified based fully homomorphic encryption and Prim's algorithm." Cluster Computing 22 (2019): 11411-11424.

[14] Alaya, Bechir, Lamri Laouamer, and Nihel Msilini. "Homomorphic encryption systems statement: Trends and challenges." Computer Science Review 36 (2020): 100235.

[15] Chillotti, Ilaria, et al. "TFHE: fast fully homomorphic encryption over the torus." Journal of Cryptology 33.1 (2020): 34-91.

[16] Kucherov, Nikolay N., Maxim A. Deryabin, and Mikhail G. Babenko. "Homomorphic encryption methods review." 2020 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus). IEEE, 2020.

[17] Najam, Shaheryar, Mujeeb Ur Rehman, and Jameel Ahmed. "Data encryption scheme based on adaptive system." 2020 Global Conference on Wireless and Optical Technologies (GCWOT). IEEE, 2020.

[18] Tran, Julian, et al. "Implementing homomorphic encryption based secure feedback control." Control Engineering Practice 97 (2020): 104350.

[19] Wood, Alexander, Kayvan Najarian, and Delaram Kahrobaei. "Homomorphic encryption for machine learning in medicine and bioinformatics." ACM Computing Surveys (CSUR) 53.4 (2020): 1-35.

[20] Li, Baiyu, and Daniele Micciancio. "On the security of homomorphic encryption on approximate numbers." Advances in Cryptology–EUROCRYPT 2021: 40th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, October 17–21, 2021, Proceedings, Part I 40. Springer International Publishing, 2021.

★★★