WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

The Effectiveness of Homomorphic Encryption in Protecting Data Privacy.

[1]Chris Gilbert, [2]Mercy Abiola Gilbert

[1]Professor [2]Instructor

[1]Department of Computer Science and Engineering/College of Engineering and Technology/William V.S. Tubman University/chrisgilbertp@gmail.com/ cabilimi@tubmanu.edu.lr

[2]Department of Guidance and Counseling/College of Education/William V.S. Tubman University /mercyabiola92@gmail.com/moke@tubmanu.edu.lr

DOI : https://doi.org/10.55248/gengpi.5.1124.3253

ABSTRACT

As the use of digital services grows, protecting the privacy and integrity of sensitive data, especially in fields like healthcare, finance, and secure surveying, has become a critical concern. Homomorphic encryption (HE) offers a solution by allowing computations to be performed on encrypted data without revealing the original information. This paper examines the principles of homomorphic encryption and its applications in privacy-preserving tasks, focusing on its use in cloud computing, healthcare, and cybersecurity. Various types of HE schemes, including Fully Homomorphic Encryption (FHE), Partially Homomorphic Encryption (PHE), and Somewhat Homomorphic Encryption (SHE), are reviewed to assess their performance, efficiency, and real-world use. The paper also discusses the challenges of implementing HE, such as computational overhead and key management, and suggests directions for future research to improve the scalability and usability of HE in real-time applications. Addressing these challenges will make homomorphic encryption an essential tool for secure, privacy-preserving data processing and sharing in modern digital systems.

Keywords: *Homomorphic encryption, data privacy, cryptography, cloud computing, healthcare security, fully homomorphic encryption, privacy-preserving computing, secure data processing, big data, multiparty computation.*

1. Introduction

Networked services are increasingly relied upon, making it essential to have effective methods for ensuring data privacy and integrity. In critical sectors such as healthcare, finance, and secure surveying, the need for robust cryptographic solutions to protect sensitive data has become paramount. Terms like “federated learning,” “fog/edge computing,” “security and privacy tokens,” “non-blocking censorship,” and “privacy-preserving analytics” have become central to discussions about creating stronger privacy solutions. These technologies share a common goal: to enhance data security for real-time processing demands at remote sites. They align closely with cryptographic solutions such as homomorphic encryption, which provides a framework for secure outsourcing and computation (Aslett et al., 2015; Abilimi & Adu-Manu, 2013; Gilbert, Oluwatosin & Gilbert, 2024).

Homomorphic encryption allows computations to be performed on encrypted data, preserving data privacy, particularly in cloud computing and distributed machine learning environments (Sen, 2013; Alloghani et al.,2019; Jia et al., 2021). This method is valuable when operations need to be carried out on encrypted data without revealing the plaintext. However, homomorphic encryption is limited in terms of the operations it can support. Its four key components—key generation, encryption, decryption, and evaluation algorithms—help maintain data security through cryptographic methods like chaotic maps, lattices, and number theory(Alloghani et al.,2019). Despite its effectiveness, performing operations on encrypted data can be inefficient, leading to increased time and space complexity. Homomorphic computations require tokens that verify correctness without accessing private keys (Su et al.,2024;Rupa et al.,2023; Jin et al, 2018). Therefore, ongoing research focuses on enhancing these encryption schemes to support a broader range of operations while striking a balance between security and performance (Jafarigol et al., 2023; Abilimi & Yeboah, 2013; Gilbert, Auodo & Gilbert, 2024).

1.1. Background and Significance

Homomorphic encryption is a widely used cryptographic technique designed to preserve privacy. It has broad applications in protecting data, processing queries, and ensuring secure data aggregation. While other modern public-key cryptographic methods, such as elliptic curve cryptography (Wylde et al., 2022; Riek & Böhme, 2018; Cassim, 2015; Iezzi, 2020; Abilimi et al., 2015) and lattice-based encryption, are effective in safeguarding privacy, this survey suggests that homomorphic encryption is particularly well-suited for preserving data privacy in big data analytics (Rupa et al.,2023;

Jin et al, 2018). This is especially relevant in cases where additional layers of encryption, from logistics to software and data, may be required to protect sensitive information stored in the cloud and on servers.

As privacy concerns grow in today's digital world, the increasing collection and processing of confidential information pose significant risks. Homomorphic encryption offers a practical solution for safeguarding both personal and enterprise data, particularly in query processing and secure data aggregation (Archer et al., 2023; Yeboah, Opoku-Mensah & Abilimi, 2013a). This method enables queries to be performed on encrypted data without exposing the actual content, ensuring strong privacy protection. The final results can be decrypted directly by the intended recipient, further ensuring secure data interpretation (Opoku-Mensah, Abilimi & Boateng, 2013). Given its unique capabilities, homomorphic encryption has become a foundational component in secure system architectures, and further technological advancements are needed to fully integrate it into distributed systems (Opoku-Mensah, Abilimi & Amoako, 2013).

1.2 Research Approach and methods

The research approach and methods in this paper focus on cryptographic systems, particularly their application in maintaining data privacy and security, with an emphasis on homomorphic encryption (HE). The primary research methods are as follows(Othman et al.,2015; Ren et al.,2021).

Theoretical Framework: The study is based on cryptographic principles, specifically homomorphic encryption schemes, which allow encrypted data to be processed without needing to decrypt it. The research considers both symmetric and asymmetric encryption, and investigates cryptographic techniques like chaotic maps, lattices, and number theory.

Survey Methodology: The paper reviews various homomorphic encryption schemes, including Fully Homomorphic Encryption (FHE), Somewhat Homomorphic Encryption (SHE), and Partially Homomorphic Encryption (PHE). It compares these schemes in terms of effectiveness, performance, and limitations(Alloghani et al.,2019; Jia et al., 2021). Additionally, it examines cryptographic libraries and algorithms, such as the Ring Learning With Errors (RLWE) and Paillier cryptosystem, to evaluate their computational efficiency in different fields like healthcare and cloud computing (Sinha eta al., 2023).

Analytical Methods: The research includes a detailed analysis of the performance overhead associated with homomorphic encryption. It focuses on the computational complexity, efficiency, and arithmetic optimizations required, especially in distributed systems, cloud computing, and privacy-preserving machine learning environments.

Experimental Simulations: Secure models are developed for privacy-preserving applications such as deep learning and multiparty computations using homomorphic encryption libraries. The study tests these models in cloud-based environments to evaluate their efficiency (2016; Tao et al., 2019; Yeboah, Opoku-Mensah & Abilimi, 2013b).

Application-Oriented Research: Finally, the research applies homomorphic encryption to real-world scenarios, particularly in cybersecurity, cloud computing, healthcare, and secure data processing. The emphasis is on maintaining privacy in big data analytics while improving the efficiency of encryption schemes (Cassim, 2015; Basharat et al., 2023).

Together, these methods aim to enhance the efficiency of encryption schemes while preserving data privacy across a range of digital environments.

2. Homomorphic Encryption

Homomorphic Encryption (HE) was first introduced by Gentry in 2009, marking a significant breakthrough in cryptographic technology. Since then, the development and adoption of homomorphic encryption systems have progressed rapidly. Specifically, in the Ring Learning With Errors (RLWE) setting, the number of public and private keys generated during encryption has increased dramatically(Alloghani et al.,2019;Su et al.,2024;Rupa et al.,2023; Jin et al, 2018). This rise has spurred demand for homomorphic encryption schemes that offer greater computational efficiency, reduced arithmetic complexity, and a smaller memory footprint. These challenges have become particularly relevant with the introduction of use cases like Smart Ubiquitous and Safe AI (SUSA), which emphasizes security and privacy in Industry 4.0 environments. As a result, there has been a growing need for arithmetic optimizations in RLWE schemes (Agrawal et al., 2020).

Homomorphic encryption schemes possess a unique homomorphic property, allowing operations to be performed on ciphertexts without decrypting them(Alloghani et al.,2019; Jia et al., 2021; Jin et al, 2018). This means that after an operation is conducted on the encrypted data, the result can be decrypted to reveal the encrypted outcome of that operation on the original plaintext. Since the inception of HE, public-key cryptography over various algebraic structures has served as a core building block for these systems. A notable example is the Ring-LWE problem, which plays a crucial role in the construction of secure homomorphic encryption schemes (Ren et al.,2021; Su et al.,2024;Rupa et al.,2023; Jin et al, 2018; Archer et al., 2023).

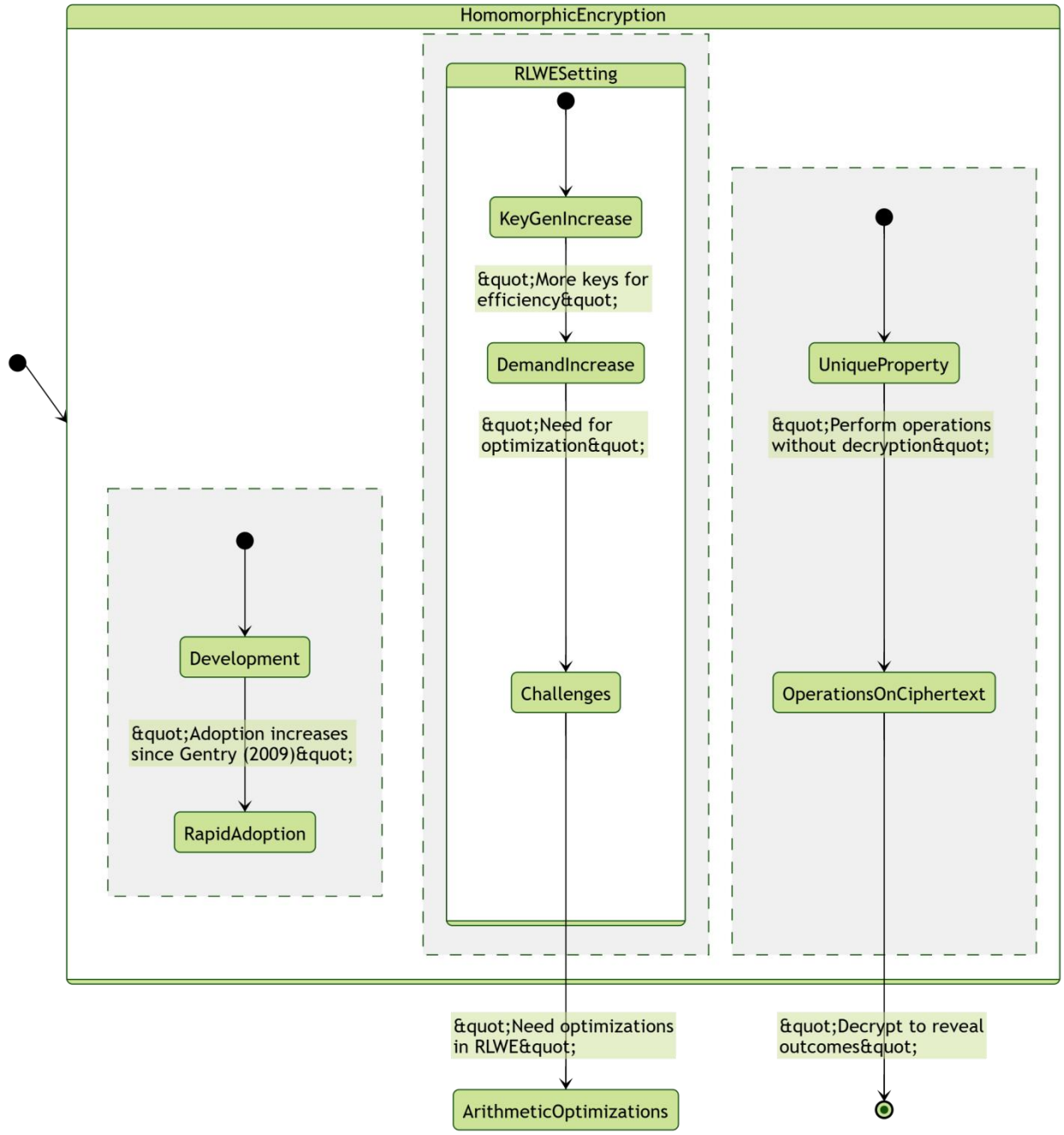

***Figure 1*: Homomorphic encryption allows operations on encrypted data securely.**

This diagram provides a simplified view of the landscape around *Homomorphic Encryption*, a type of encryption that allows for computations on encrypted data without needing to decrypt it first:

i. Homomorphic Encryption is the main focus. It's an advanced encryption method that's particularly interesting because it allows data to be worked on securely—without exposing sensitive information during processing.

ii. Development:

   a. This area highlights how homomorphic encryption has gained traction since *Craig Gentry's breakthrough in 2009*, which first proved that fully homomorphic encryption was possible.

   b. Since then, there's been a "Rapid Adoption" of this technology, as researchers and organizations recognize its potential to securely handle data in areas like cloud computing, healthcare, and finance(Basharat et al., 2023).

iii. RLWE Setting (Ring Learning With Errors):

a. This is a specific mathematical framework used within homomorphic encryption. It's popular because it can offer more efficient encryption.

b. KeyGenIncrease: As more people use this technology, the demand for generating keys has risen to enhance efficiency, represented here as "More keys for efficiency."

c. DemandIncrease: Alongside the rise in usage, there's a growing need for optimization to make these operations faster and more practical, hence the mention of "Need for optimization."

d. Challenges: This highlights that optimizing RLWE settings is tough. The complexity of the mathematics behind RLWE creates challenges in making it faster or more efficient.

e. Arithmetic Optimizations: To tackle these challenges, researchers are focusing on improving the arithmetic involved in these encryption methods, looking for ways to make calculations faster and more manageable.

iv. Operations on Ciphertext:

a. A unique and powerful feature of homomorphic encryption is that it lets people perform calculations on encrypted data.

b. Unique Property: This section points out that with homomorphic encryption, you can "perform operations without decryption," which means data remains secure even while it's being processed.

c. Operations on Ciphertext: Essentially, you can compute results on the encrypted data, and only after the processing is complete do you need to decrypt to see the outcome(Alloghani et al.,2019;Othman et al.,2015; Ren et al.,2021)

2.1. Definition and Overview

Historically, data security has been maintained through cryptographic techniques like encryption, which protects sensitive data by transforming it into ciphertext, ensuring only authorized parties can access it. However, traditional encryption methods are limited in situations where third parties must perform services on behalf of users, such as analyzing medical data(Alloghani et al.,2019;Su et al.,2024;Rupa et al.,2023; Jin et al, 2018). In these cases, third-party access to sensitive information can potentially compromise user privacy. Therefore, it is crucial to use secure computing technologies that allow data processing without revealing privacy-sensitive information. Secure Multiparty Computing (SMC) protocols, including Homomorphic Encryption (HE), provide a solution by enabling computations on encrypted data while maintaining confidentiality, even when using cloud services (Al-Harrasi, Shaikh & Al-Badi, 2023;Wylde et al., 2022; Riek & Böhme, 2018; Cassim, 2015)

Jordan et al., 2022).

Data privacy consists of three essential components: confidentiality, integrity, and availability. Confidentiality restricts data access to authorized users or entities, while integrity ensures that data remains accurate and has not been tampered with. Finally, availability guarantees that data is accessible when needed (Sharma, 2013).

2.2. Types of Homomorphic Encryption Schemes

In public-key encryption systems, a key generation process creates a pair of keys—one public and one private. The public key is shared openly, typically published in a public directory, while the encryption and decryption keys are related in a way that makes it computationally infeasible for anyone to deduce one from the other. In additive homomorphic encryption systems, these keys are denoted as rA (for encryption) and rB (for decryption). This paper focuses on a simplified version of the ElGamal encryption system, which supports additive homomorphism(Alloghani et al.,2019; Jia et al., 2021; Othman et al.,2015; Ren et al.,2021). Similarly, other homomorphic encryption schemes use this framework to generate ciphertext efficiently. One challenge in homomorphic encryption, however, is the large key sizes involved—particularly for polynomial encryption—resulting in inefficiencies and longer processing times.

Although it is computationally difficult to evaluate a function fff of a hard problem hhh on a given input xxx, decryption remains feasible through algebraic manipulation of the ciphertext. Homomorphic public-key encryption allows efficient generation of ciphertext for any result of fff, as the key generation process ensures secure operations(Su et al.,2024;Rupa et al.,2023; Jin et al, 2018). This process must be carefully designed to ensure security, as demonstrated by several homomorphic public-key encryption systems. In such systems, algebraic homomorphism-based encryption ensures that it is impossible to compute a primitive element from either the output f(x)f(x)f(x) or the input xxx. Additive homomorphism, in particular, is one of the most commonly studied properties in public key encryption systems (Suo et al., 2023).

Homomorphic encryption has the distinct advantage of performing complex operations on encrypted data without the need to decrypt it. It can generate ciphertext for the final result while maintaining the security of the underlying data. Achieving this level of encryption requires running multiple cryptographic processes (Sen, 2013; Ren et al.,2021; Su et al.,2024;Rupa et al.,2023; Jin et al, 2018). Cryptographic systems are generally divided into two categories: symmetric and asymmetric. Symmetric encryption uses the same key for both encryption and decryption, while asymmetric encryption relies on different keys for these tasks. Most symmetric encryption schemes, however, do not support homomorphic operations, unlike their asymmetric counterparts.

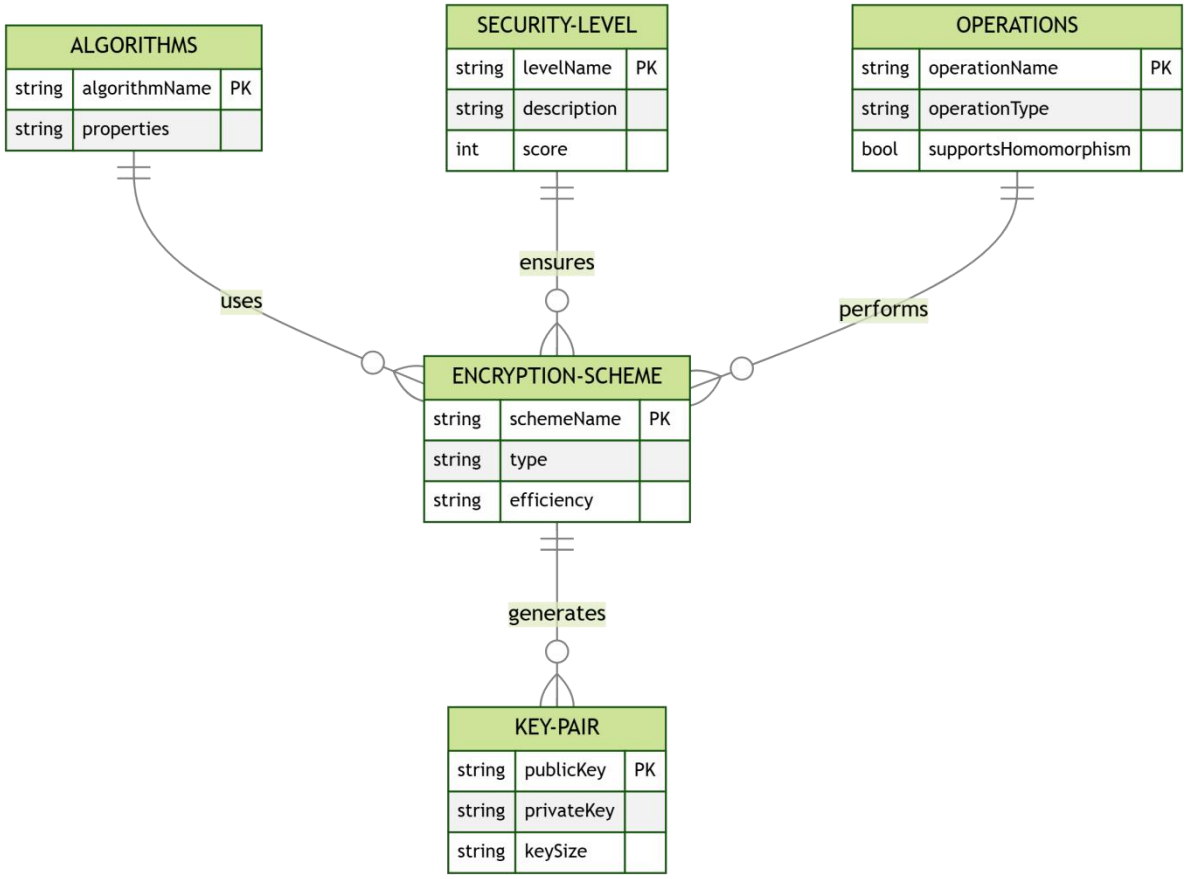

***Figure 2*: Represents types and relationships of encryption schemes.**

This diagram provides a high-level overview of the essential components in an encryption system and how they work together as follows:

i. **Algorithms**: This section represents the different encryption algorithms (like RSA or AES) used to secure data. Each algorithm has its own *properties* that define how it works or what makes it unique. These algorithms are part of an encryption scheme, meaning they're the specific "recipes" or methods used within a broader encryption setup.

ii. **Security Level**: Every encryption scheme has a *security level* that defines how strong or secure it is. The *security level* includes a *name*, a *description* of its strength or purpose, and a *score* representing how secure it is. The encryption scheme relies on these security levels to protect data, ensuring that the encryption is tough enough to withstand attacks.

iii. **Operations**: These are the different actions or processes that can be done within the encryption system, like *encryption*, *decryption*, or *data processing*. Each operation has a type and indicates whether it supports *homomorphic encryption* (a type of encryption that allows data processing without revealing the data itself). The encryption scheme uses these operations to handle data securely.

iv. **Encryption Scheme**:The *encryption scheme* is at the center of this model, bringing together all the components. It's the overarching setup that uses algorithms, ensures security levels, performs operations, and generates keys. Think of the encryption scheme as the "blueprint" that defines how all these parts work together to protect data.

v. **Key Pair**:The *key pair* consists of a *public key* and a *private key*. These keys are generated by the encryption scheme. The public key is shared openly and can be used to encrypt data, while the private key is kept secret and is used to decrypt the data. The key pair is essential for secure communication, ensuring that only the intended recipient can access the encrypted data.

3. Theoretical Implications for Data Privacy

Analytical Improvements: Traditional encryption schemes face significant challenges during the bootstrapping phase, which drastically slows down computational efficiency and is highly intolerant of errors. This phase involves complex calculations, leading to high computational costs. For instance, it is difficult to evaluate data similarity when multiplied by coefficients between 0 and 1, so standardization is used to enhance accuracy. A key improvement is seen in the Isomorphism-LWE scheme, which offers larger moduli in polynomial rings compared to LWE, enabling a smaller ring

dimension. Typically, the ring's degree of coefficients ranges between 1 and 4. One of the primary goals of this project was to create a secure computational platform (Agrawal et al., 2020).

Sustainability is a crucial theoretical aspect of the proposed model. The approach offers multiple pathways to ensure the secure handling of data across various encryption methods. After evaluating different encryption techniques, the most suitable method was selected based on its precision and efficiency. This model not only enables secure data transfer across networks in theory but also provides practical solutions for facilitating communication between networks in specialized fields (Patel et al., 2022).

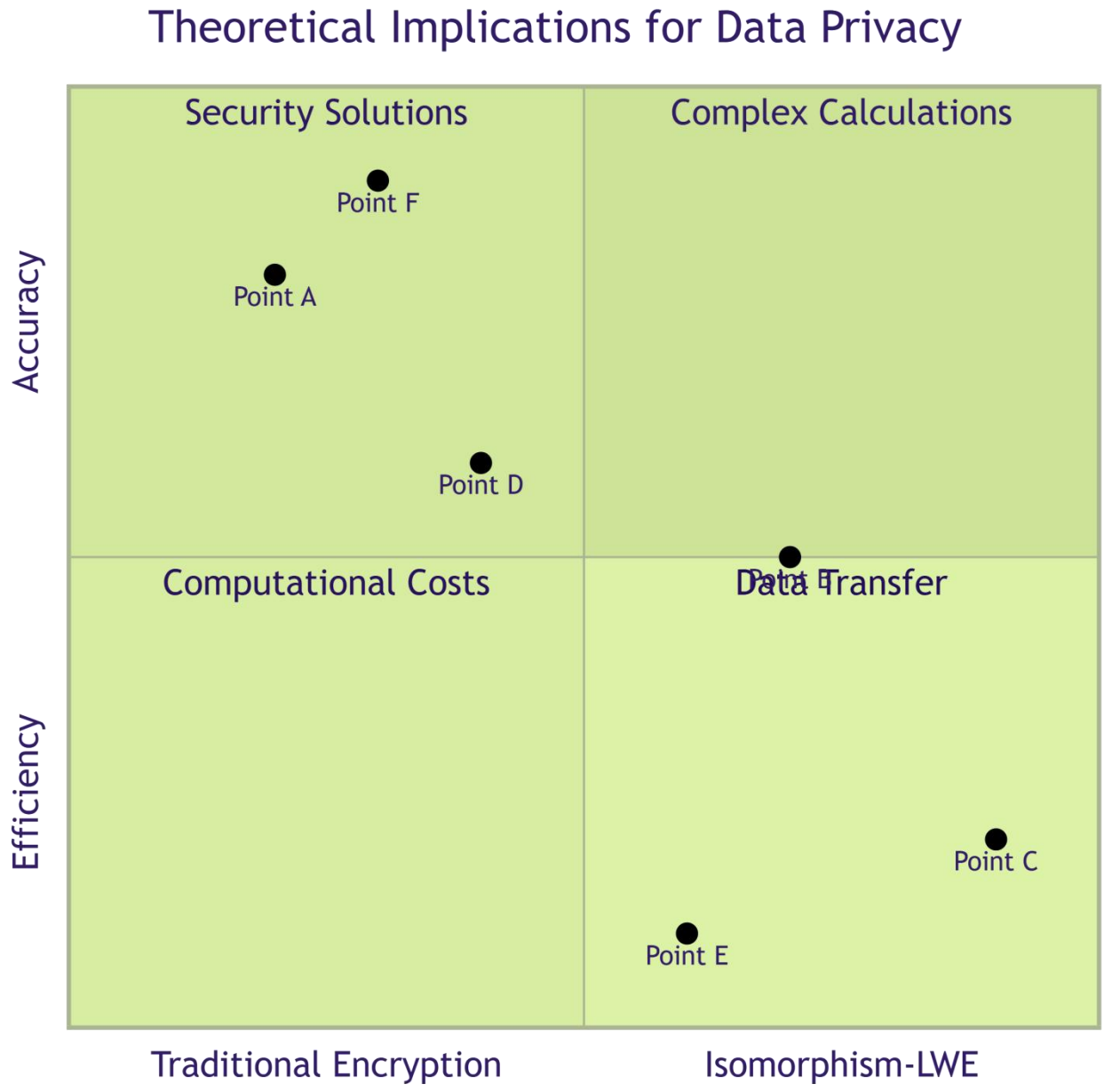

Figure 3:Implications of encryption on data privacy efficiency.

This diagram provides a simplified way to understand the differences between two encryption approaches—*Traditional Encryption* and *Isomorphism-LWE*—in terms of how they handle data privacy. The chart is divided into four key areas, each highlighting different priorities and trade-offs between accuracy and efficiency:

Security Solutions (Top Left - Traditional Encryption, High Accuracy):This area focuses on security methods that use traditional encryption techniques. These methods prioritize *accuracy*, ensuring that data remains secure and correctly encrypted. **Point A** and **Point F**: Represent solutions in traditional encryption that emphasize high accuracy. They might be especially suitable for scenarios where data accuracy is critical, like financial transactions or secure communication.

Complex Calculations (Top Right - Isomorphism-LWE, High Accuracy): This quadrant highlights more advanced, complex calculations that can be done with Isomorphism-LWE, a newer encryption approach. **Point C**: Represents a solution within Isomorphism-LWE that excels at handling

complex calculations with high accuracy. This could be useful in applications where secure data processing (like in machine learning on encrypted data) is essential, even if it requires more computational power.

Computational Costs (Bottom Left - Traditional Encryption, High Efficiency): Here, the focus is on solutions that keep computational costs low, making traditional encryption methods more efficient and quicker, though possibly at the expense of some accuracy. **Point E**: Represents an efficient traditional encryption method that minimizes the resources needed to process data. It would be ideal for applications needing fast, lightweight encryption without too much concern for ultra-high accuracy, like data logging or storage.

Data Transfer (Bottom Right - Isomorphism-LWE, High Efficiency): This area emphasizes efficient data transfer using Isomorphism-LWE, balancing security with the ability to move data quickly. **Point D** and **Point C** (overlapping): Represent techniques within Isomorphism-LWE that support efficient data transfer, suitable for real-time data exchanges or applications that need to keep data encrypted during transmission.

3.1. Confidentiality and Integrity Preservation

The growing use of smartphones and mobile devices presents a significant security threat to billions of users worldwide. Traditionally, security research in mobile environments has focused primarily on testing the operating mechanisms of these devices, often overlooking the unique challenges posed by smartphones. However, securing global privacy and data communication cannot be limited to just operating systems; the broader implications of potential data breaches are too serious to ignore. Recent research has highlighted that current methods for ensuring data privacy through encryption are insufficient to fully protect user data from tampering and other security threats (Tao et al., 2019; Al-Harrasi, Shaikh & Al-Badi, 2023;Wylde et al., 2022; Riek & Böhme, 2018; Cassim, 2015; Xiong et al., 2020). This study undertakes a systematic review of the methodologies and processes involved in securing communication systems in the smartphone environment, offering key conclusions, recommendations, and policy implications.

Homomorphic encryption methods, such as Fully Homomorphic Encryption (FHE), Somewhat Homomorphic Encryption (SHE), and Even Homomorphic Encryption (EHE), have proven effective in safeguarding data services. FHE is particularly powerful, as it allows for an unlimited number of operations on encrypted data, despite the high computational demands and large ciphertext sizes it generates. The Paillier cryptosystem, along with its variants like symmetric key Paillier, offers a partial solution by mitigating some of the complexity in homomorphic encryption, but it introduces noise with each operation, making it only somewhat homomorphic. To address this issue, additive homomorphic schemes like SHE and EHE have been proposed, which limit the number of computations or operations on ciphertext, improving the overall efficiency of the encryption process (Patel et al., 2022; Sen, 2013). These advancements aim to unlock the full potential of FHE by enabling infinite arithmetic operations on encrypted data.

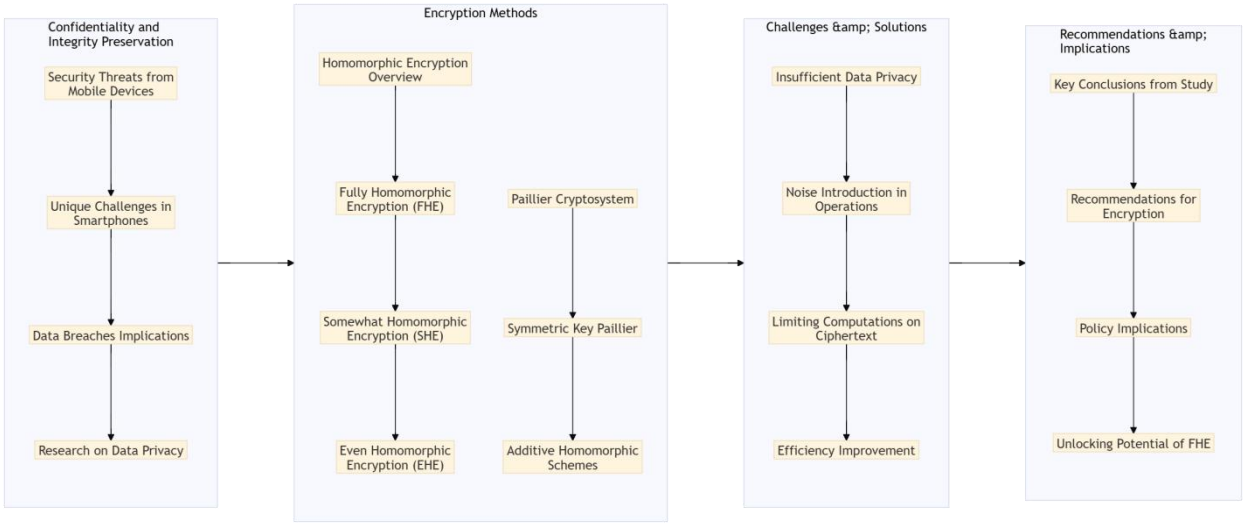

Figure 4: *Diagram illustrates encryption and security challenges in mobile devices.*

This diagram gives a clear, step-by-step look at the current landscape of data privacy and encryption, especially as it relates to mobile devices and the challenges of using advanced encryption methods.

i. Confidentiality and Integrity Preservation: This section is about protecting data on mobile devices, which face unique security risks.

- Security Threats from Mobile Devices: Highlights the challenges mobile devices face, like exposure to network threats or physical theft, which can put sensitive data at risk.
- Unique Challenges in Smartphones: Points out that smartphones, with their constant connectivity and frequent software updates, require special attention to keep data secure.

- Data Breaches Implications: Explores what can happen if mobile security fails—data breaches can lead to serious consequences, from financial loss to compromised personal information.
- Research on Data Privacy: Indicates the ongoing work to understand and address these privacy concerns, with the goal of finding better ways to secure mobile data.

ii. Encryption Methods: This section gives an overview of different encryption techniques, focusing on homomorphic encryption, which is powerful because it allows data to be processed while still encrypted.

- Homomorphic Encryption Overview: Introduces homomorphic encryption, a method that lets you work with data without ever exposing it. This is a big leap forward for data privacy.
- Fully Homomorphic Encryption (FHE): This type of encryption allows for unlimited calculations on encrypted data, though it requires a lot of computing power.
- Somewhat Homomorphic Encryption (SHE) and Even Homomorphic Encryption (EHE): Describe simpler forms of homomorphic encryption, which allow only a limited number of calculations but are more manageable in terms of resources.
- Paillier Cryptosystem and related methods: These represent encryption techniques that allow specific operations (like addition) on encrypted data, useful for certain applications where full homomorphic encryption is not practical.

iii. Challenges & Solutions: This section deals with the obstacles and potential fixes in using homomorphic encryption.

- Insufficient Data Privacy: Points out that, even with advanced encryption, keeping data fully private can still be challenging.
- Noise Introduction in Operations: When performing calculations on encrypted data, "noise" (unwanted extra data) accumulates, limiting the number of calculations possible. This noise needs to be managed to make encryption practical.
- Limiting Computations on Ciphertext: Suggests that simplifying the types of calculations allowed on encrypted data can help keep noise under control and improve efficiency.
- Efficiency Improvement: Refers to the ongoing work to make homomorphic encryption faster and less resource-intensive, so it can be more widely used.

iv. Recommendations & Implications: This final section summarizes findings and suggests steps to move forward.

- Key Conclusions from Study: Outlines what we've learned so far from studying data privacy and encryption techniques.
- Recommendations for Encryption: Provides practical advice on how to improve encryption strategies to make them more effective.
- Policy Implications: Emphasizes the need for strong policies to support data privacy, as technology alone isn't enough.
- Unlocking Potential of FHE: Calls attention to the potential of Fully Homomorphic Encryption as a powerful tool for data privacy, despite current challenges with efficiency.

4. Applications in Cybersecurity

Homomorphic encryption (HE) allows arbitrary functions and analyses to be performed on encrypted datasets without the need for decryption, ensuring a high level of security. In HE-based systems, data owners first encrypt their datasets and then send the encrypted data to external storage or processing entities(Alloghani et al.,2019;Ren et al.,2021; Su et al.,2024;Rupa et al.,2023; Jin et al, 2018). These entities do not have access to the secret keys used for encryption, which prevents them from decrypting or gaining any meaningful insight into the encrypted datasets. Since the introduction of Fully Homomorphic Encryption (FHE) in 2009, it has become possible to perform arbitrary operations on encrypted data. However, these systems can be significantly slower than operations performed on plaintext, as they fail to account for the computational complexity associated with plaintext operations(Rupa et al.,2023; Jin et al, 2018).

HE is designed to encrypt data using linear arithmetic, followed by encryption permutations, re-encryption, and function execution over the ciphertext. HE schemes have been proposed as highly secure, scalable solutions, especially in cloud computing environments. Additionally, several homomorphic encryption schemes have been developed that are secure against quantum computing threats(Rupa et al.,2023; Jin et al, 2018). For instance, in healthcare, HE-based security schemes ensure the privacy and confidentiality of patient data by transforming it into ciphertext before processing. This allows healthcare data owners to securely process and share encrypted data with external entities, such as storage servers and cloud service providers, without exposing sensitive information (Chalasani et al.,2023; Sinha eta al., 2023; Jordan et al., 2022; Jafarigol et al., 2023; Archer et al., 2023).

Protecting personal data, particularly health data, is a recognized fundamental right. Patient medical records, which are highly sensitive, must be stored, processed, and shared with the utmost security and privacy. Protecting these records not only shields individuals from identity theft and fraud, but also plays a critical role in advancing healthcare research and combating diseases. Healthcare data is collected from numerous sources, including hospitals, pharmacies, laboratories, insurance companies, and regulatory bodies, and is often referred to as healthcare datasets. As researchers increasingly rely on these datasets for data modeling, analysis, and visualization, ensuring their privacy becomes a significant concern. Conventional healthcare datasets are

vulnerable to privacy breaches, security attacks, and mismanagement, and secure computation methods, such as HE, offer a practical solution to these challenges(Chalasani et al.,2023; Sinha eta al., 2023).

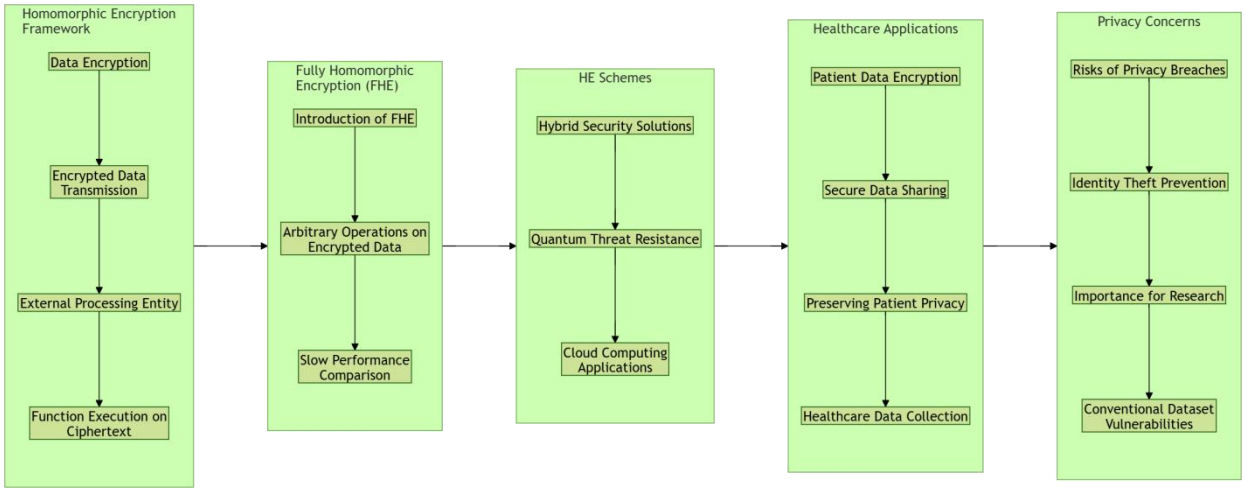

Figure 5: Homomorphic encryption enhances data security for healthcare.

This diagram shows how Homomorphic Encryption (HE)—a powerful encryption method—can be used to secure sensitive data, especially in areas like healthcare, while addressing important privacy concerns:

a. Homomorphic Encryption Framework

- Data Encryption: Data owners (like individuals or organizations) start by encrypting their data, turning it into a secure, unreadable format.
- Encrypted Data Transmission: Once encrypted, the data can be safely sent to outside services or cloud storage.
- External Processing Entity: These external entities (like cloud providers) store or process the encrypted data but cannot access its actual contents, since they don't have the decryption keys.
- Function Execution on Ciphertext: Even though the data is encrypted, calculations or analyses can still be performed on it, thanks to homomorphic encryption, ensuring privacy without needing to decrypt the data.

b. Fully Homomorphic Encryption (FHE)

- Introduction of FHE: FHE is a special type of encryption that lets users perform any operation on encrypted data, just as if it were decrypted.
- Arbitrary Operations on Encrypted Data: This feature allows for complex data processing without compromising data privacy.
- Slow Performance Comparison: A challenge with FHE is that it can be much slower than regular (plaintext) data operations, so making it faster is a focus of ongoing research.

c. HE Schemes

- Hybrid Security Solutions: HE is often combined with other security measures, creating a layered approach that strengthens protection.
- Quantum Threat Resistance: Some HE schemes are designed to withstand the potential threats posed by future quantum computers, making them more future-proof.
- Cloud Computing Applications: HE is particularly useful in cloud environments, where sensitive data needs to be processed by external servers without exposing private information.

d. Healthcare Applications

- Patient Data Encryption: Patient information is encrypted before it is shared or processed, keeping it secure from unauthorized access.
- Secure Data Sharing: Encrypted patient data can be safely shared with other organizations, like hospitals or researchers, without revealing any personal details.
- Preserving Patient Privacy: This method of sharing encrypted data allows healthcare providers to maintain patient confidentiality, which is crucial for trust.

- Healthcare Data Collection: With HE, healthcare organizations can securely collect and analyze patient data, supporting valuable research and improving healthcare outcomes without compromising privacy(Chalasani et al.,2023).

e. Privacy Concerns

- Risks of Privacy Breaches: Despite strong encryption, privacy risks exist. Protecting sensitive data helps to mitigate these risks.
- Identity Theft Prevention: Encrypted data is much harder to misuse, reducing the risk of identity theft or fraud.
- Importance for Research: Protecting privacy is essential for research, as it encourages data sharing in a way that respects individuals' confidentiality.
- Conventional Dataset Vulnerabilities: Traditional (unencrypted) datasets are prone to breaches and misuse, making HE a practical solution for enhancing data security.

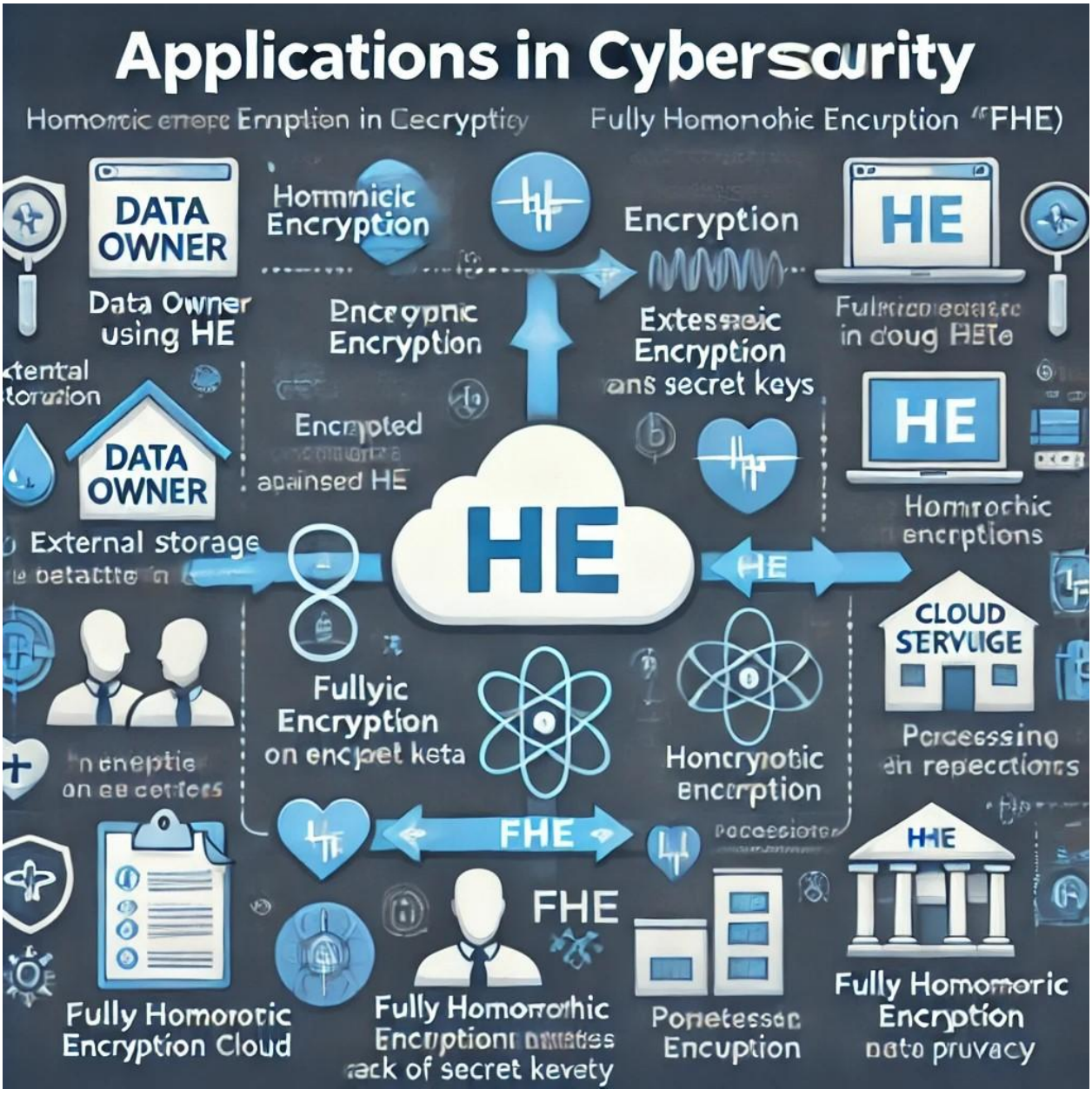

Figure 6: Homomorphic Encryption (HE) illustrated

This diagram illustrates how Homomorphic Encryption (HE) protects data in cybersecurity, especially in cloud computing and healthcare(Sharma et al.,2023). It shows that data owners encrypt their data before sharing it with external services, ensuring that only they have the keys to decrypt it. *Fully* Homomorphic Encryption (FHE) allows complex operations on encrypted data without exposing it, enabling secure data analysis in sensitive fields. Additionally, some HE schemes are resistant to quantum threats, making them a future-proof choice for data privacy(Al-Harrasi, Shaikh & Al-Badi, 2023;Wylde et al., 2022; Riek & Böhme, 2018; Cassim, 2015). Overall, HE allows safe processing and storage of data without compromising privacy.

4.1. Secure Data Processing

Partial Homomorphic Encryption (PHE) is often applied to algebraic operations like addition and multiplication. For instance, in supervised learning, the sum is calculated through the linear transformation of data points with weight vectors. Because the encryption system operates linearly, the dot product can be obtained by expanding the polynomials and linearly combining the results (Munjal & Bhatia, 2022). Threshold prediction can then be achieved using element-wise operations, such as taking the absolute value of a prediction score for one class and comparing it with prediction scores from other classes. In the healthcare sector, secure linear regression is commonly used since many machine learning and economic models are linear and require training on the entire dataset(Basharat et al., 2023; Sinha eta al., 2023). Homomorphic encryption supports secure access to genomic and individual-level data, allowing more researchers to access raw patient data while maintaining privacy and security (Riek & Böhme, 2018; Cassim, 2015; Sidorov et al., 2022).

Homomorphic Encryption (HE), initially proposed by Rivest, Adleman, and Dertouzos, enables operations to be performed directly on encrypted data, which is not possible with traditional encryption systems. PHE allows for the computation of a single type of operation without the need to access the secret key. This makes PHE a good fit for applications that require limited operations. RSA and ElGamal are among the simplest examples of PHE systems, as they can handle only multiplication and addition, respectively. For more complex data processing tasks, Stochastic Partial Homomorphic Encryption (SPHE) has been developed to minimize statistical errors and avoid the need for multiplication during computations.See the diagram below:

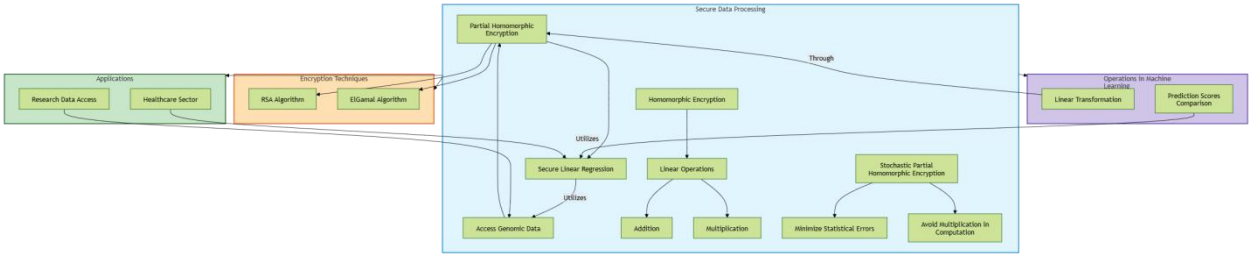

Figure 7: Explains secure data processing through encryption techniques.

4.2. Cloud Computing

Practical homomorphic encryption libraries have the potential to significantly boost cloud and edge computing applications (Fahina et al., 2022; Gilbert & Gilbert, 2024e). This paper reviews the software and APIs of lattice-based homomorphic encryption libraries that are used in real-world scenarios, such as running machine learning inferences, executing decentralized smart contracts via blockchains, performing mathematical operations on encrypted RNA genomic data, facilitating multiparty quantitative finance calculations, and supporting secure face detection and recognition (Gilbert & Gilbert, 2024h). One of the main schemes used in these libraries is based on Ring Learning with Errors (RLWE), which is among the most popular public-key cryptographic schemes for homomorphic encryption (Agrawal et al., 2020; Gilbert & Gilbert, 2024a).

To assess the effectiveness of homomorphic encryption in cloud computing, we developed a secure, efficient deep-learning model capable of performing privacy-preserving inference tasks by integrating multiple homomorphic encryption libraries (Gilbert & Gilbert, 2024i; Sri Sathya et al., 2018). Additionally, we aimed to address issues like overfitting in outsourced environments and demonstrate the versatility of the model by applying it to a broad range of use cases. Homomorphic encryption facilitates encrypted data computation, enabling collaborative computing between multiple parties without revealing the underlying data. This capability is critical for widespread adoption and further development, as the technology is essential in fields such as healthcare, smart grids, and genomics, where stringent privacy requirements are non-negotiable(Sharma et al.,2023).

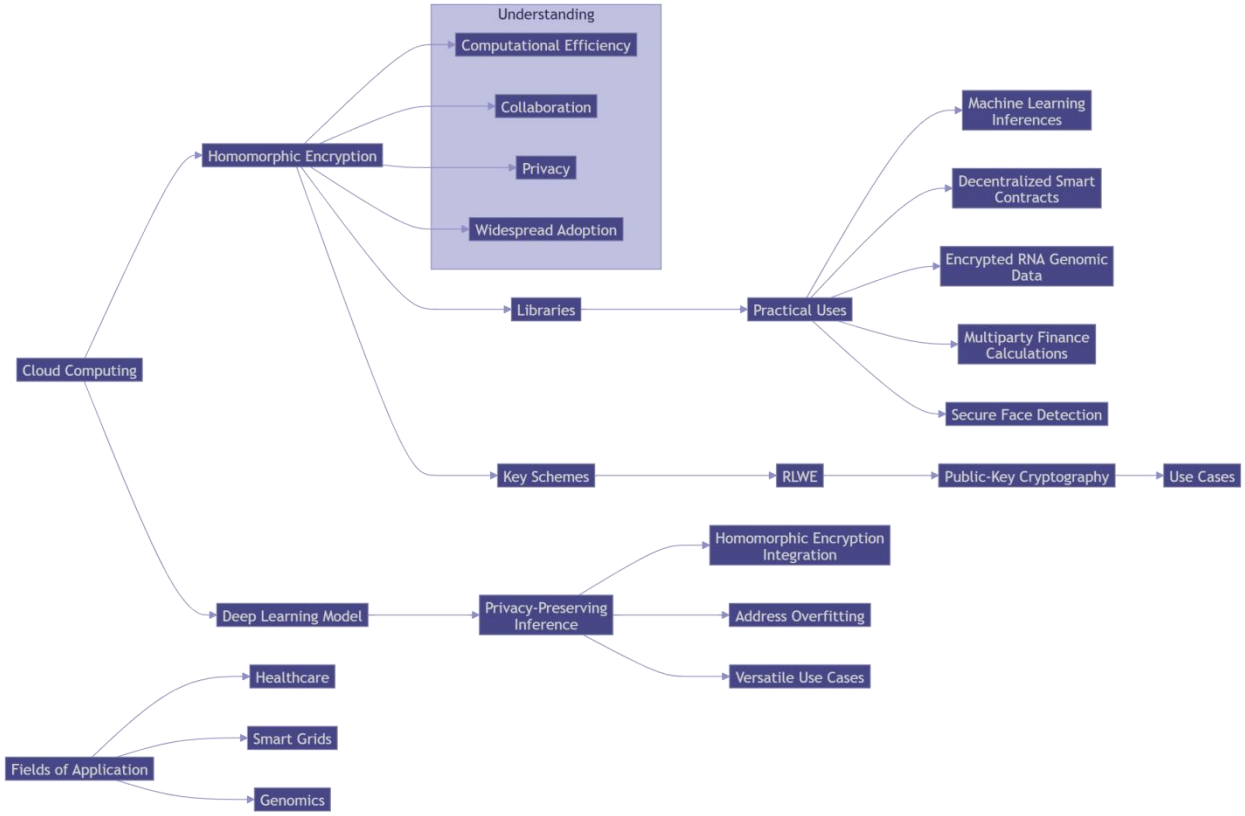

Figure 8:Cloud computing enhanced by secure homomorphic encryption.

This diagram (***Figure 8***) provides an in-depth look at Homomorphic Encryption (HE)—a powerful tool for data security—and illustrates its applications, challenges, and future potential across various fields.

At the heart of the diagram is an exploration of HE's foundational requirements for widespread adoption. To make HE practical, researchers are focusing on improving computational efficiency, enabling collaboration, ensuring data privacy, and ultimately increasing adoption across industries. These elements form the basis for understanding and advancing HE's capabilities(Tao et al., 2019; Al-Harrasi, Shaikh & Al-Badi, 2023;Wylde et al., 2022; Riek & Böhme, 2018; Cassim, 2015).

One key area highlighted is the growing list of practical uses and the availability of libraries that support HE implementation. Libraries serve as essential tools, making it easier for developers and researchers to apply HE in real-world scenarios. The diagram lists practical applications of HE, such as enabling secure machine learning on encrypted data, supporting decentralized smart contracts for enhanced blockchain security, analyzing sensitive RNA genomic data while keeping it encrypted, performing private financial calculations across multiple parties, and conducting face detection in a way that preserves privacy(Tao et al., 2019; Al-Harrasi, Shaikh & Al-Badi, 2023;Wylde et al., 2022).

The diagram also introduces different key schemes and underlying technologies that support HE. Ring Learning with Errors (RLWE) is a specific mathematical foundation within HE that strengthens its security, making it resilient to various threats. Additionally, as a form of public-key cryptography, HE enables encrypted data operations without needing decryption, which broadens its applications across various sectors where data privacy is paramount.

The fields of application section underscores HE’s role in areas like cloud computing, where it enables secure data processing without revealing sensitive information. In deep learning, HE allows privacy-preserving inference, enabling AI and machine learning models to make predictions on encrypted data. Healthcare is another major field benefiting from HE, as it allows patient data to be securely analyzed and shared, protecting patient privacy in medical research (Basharat et al., 2023; Sharma et al.,2023; Chalasani et al.,2023; Sinha eta al., 2023). In smart grids, HE ensures the privacy of energy data while enabling efficient analysis. Genomics is another critical area where HE allows researchers to analyze genetic data securely, maintaining privacy in an especially sensitive domain.

The diagram concludes with the concept of privacy-preserving inference, which is essential for using HE in artificial intelligence and machine learning. This approach helps prevent overfitting, as models trained on encrypted data maintain privacy, which can lead to more generalized outcomes. Privacy-preserving inference opens up versatile use cases for HE, from finance and healthcare to genomics, where data analysis needs to be both insightful and private(;Wylde et al., 2022; Riek & Böhme, 2018; Cassim, 2015).

Overall, the diagram captures the far-reaching potential of Homomorphic Encryption to enable secure data processing in today's privacy-conscious world. It showcases how HE can transform sectors like cloud computing, healthcare, and machine learning by enabling data analysis without compromising privacy, paving the way for a safer and more collaborative data-driven future.

5. Challenges and Limitations

The implementation of homomorphic encryption as a tool for privacy enforcement can significantly change how healthcare data are stored and processed. These changes depend on the existing systems and information technology infrastructures in place. As a result, it is crucial to thoroughly assess how this technology affects confidentiality, integrity, and availability of data, while also maintaining accountability and privacy, and allowing access to information only when necessary (Dhasarathan et al., 2022; Al-Harrasi, Shaikh & Al-Badi, 2023;Wylde et al., 2022; Riek & Böhme, 2018; Cassim, 2015)).

Currently, there is a lack of digital platforms specifically designed to handle homomorphic data processing operations that ensure both the privacy of healthcare data and their secure storage. To address this gap, mechanisms that protect data integrity and confidentiality during processing are essential for healthcare data management in e-health environments(Basharat et al., 2023; Sharma et al.,2023; Chalasani et al.,2023; Sinha eta al., 2023). The most promising solutions involve platforms that utilize homomorphic encryption, which has even been proposed for securely handling sensitive data, such as COVID-19 information (Gilbert & Gilbert, 2024b; Sidorov et al., 2022).

Homomorphic encryption has demonstrated its effectiveness in solving data privacy issues and facilitating the secure use of stored data (Hamza et al., 2022; Gilbert & Gilbert, 2024c). However, despite its promise, several challenges and limitations still impede its widespread adoption.

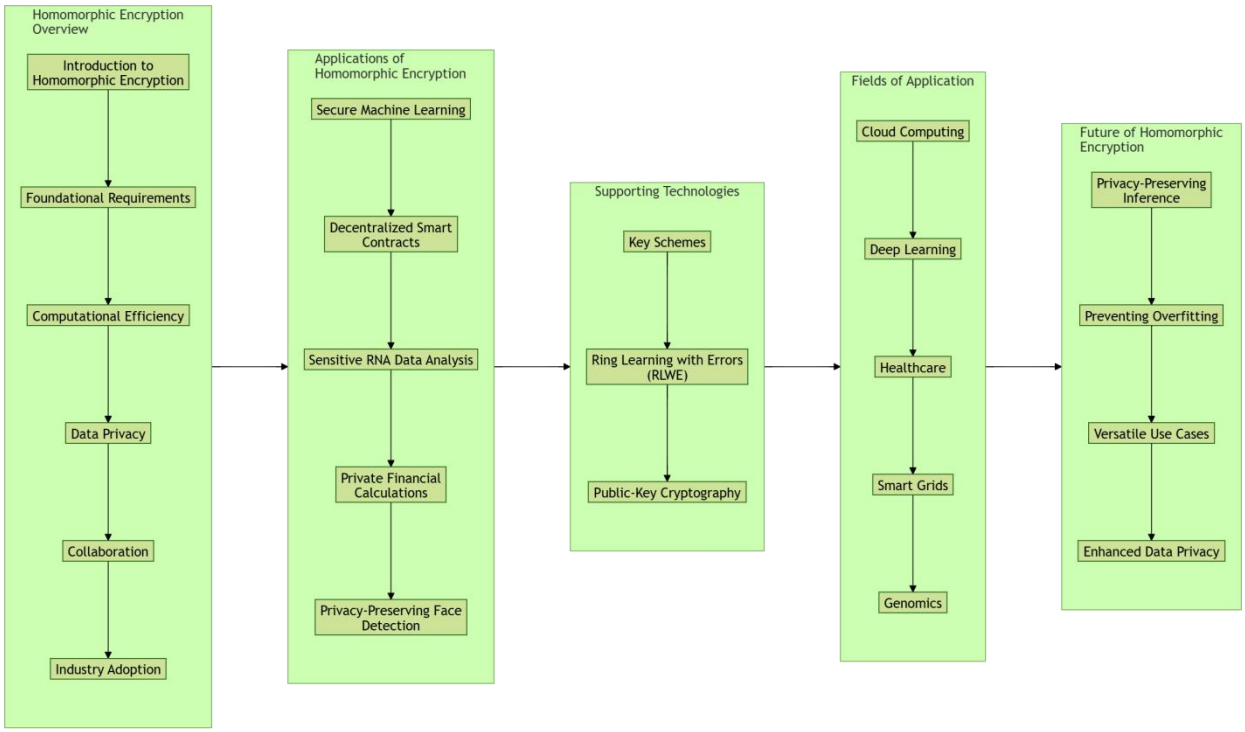

Figure 9: Illustrates Homomorphic Encryption's capabilities and applications.

This diagram provides an organized view of Homomorphic Encryption (HE)—an encryption technique that enables computations on encrypted data without needing to decrypt it first. It shows key aspects such as foundational requirements, applications, supporting technologies, primary fields of use, and potential future directions for HE.

The overview begins by introducing Homomorphic Encryption and identifying essential requirements for its effective implementation. These include achieving computational efficiency to ensure operations on encrypted data are not excessively slow, maintaining data privacy so that sensitive information remains secure, and fostering collaboration among stakeholders(Al-Harrasi, Shaikh & Al-Badi, 2023;Wylde et al., 2022; Riek & Böhme, 2018; Cassim, 2015). Together, these elements contribute to broader adoption of HE across various industries, where security and privacy are critical concerns.

The diagram then details practical applications of Homomorphic Encryption, illustrating its versatility across different scenarios. For example, HE supports secure machine learning by allowing algorithms to analyze encrypted data, which is essential for privacy-sensitive applications. In blockchain, HE enables decentralized smart contracts, adding a layer of encryption that enhances security (Tao et al., 2019; Al-Harrasi, Shaikh & Al-Badi, 2023;Wylde et al., 2022). It also facilitates sensitive data analysis, such as RNA genomic data, by allowing encrypted processing that keeps the data

private. In finance, HE supports private calculations, enabling organizations to perform financial analyses without revealing confidential information. Additionally, it allows for privacy-preserving face detection, a feature that keeps biometric data secure.

Supporting technologies play a crucial role in making HE feasible and effective. Key schemes and mathematical foundations, such as Ring Learning with Errors (RLWE), strengthen HE's security, making it more resilient to potential threats. Public-key cryptography underpins HE, allowing encrypted computations to be performed without decryption, which is fundamental for secure data processing across applications.

HE has a range of valuable applications across several fields. In cloud computing, it provides a way to protect sensitive information stored and processed in the cloud, maintaining privacy while allowing for data-driven insights. In deep learning and AI, HE enables privacy-preserving inference, allowing models to work with encrypted data without compromising privacy. The healthcare field benefits significantly from HE, as it allows medical data to be securely analyzed and shared without risking patient privacy(Chalasani et al.,2023; Sinha eta al., 2023). Smart grids leverage HE to maintain data security while optimizing energy management. Genomics is another area where HE supports secure analysis of genetic information, preserving the confidentiality of sensitive genetic data.

Looking to the future, the diagram highlights the potential of HE to advance privacy-preserving inference, a technique that enables AI models to analyze encrypted data, which can help prevent overfitting and make models more reliable. This capability will allow HE to expand into even more versatile use cases, reinforcing its role in enhancing data privacy across various industries(Riek & Böhme, 2018; Cassim, 2015)

In summary, this diagram captures the foundational components, applications, and future potential of Homomorphic Encryption. It showcases HE's ability to transform data privacy practices across diverse fields by enabling secure data processing, even in sensitive areas like healthcare, finance, and artificial intelligence. As HE continues to evolve, it promises to play a vital role in securing data-driven insights while upholding the highest standards of privacy.

5.1. Performance Overhead

Recent studies on practical homomorphic encryption have supported the development of faster and more streamlined schemes. NIST's Post-Quantum Cryptography Standardization Project has introduced 17 candidate encryption schemes aimed at ensuring safe and accurate public-key cryptography. Additionally, Deloitte's Cyber Quantities presents a new semi-hard problem that attempts to better replicate legacy encryption compression standards, offering practical speed improvements (Ajayi, 2016). While web-based solutions are not initially considered, they may serve corporate objectives by providing useful additions. Cloud-based encryption, however, can be influenced by several factors, such as cushion size and computational complexity, especially if encryption methods or ciphertexts are known. Certain cryptographic tasks, like generating signatures for custom PowerPoint presentations, cannot be efficiently managed with homomorphic encryption and commitment. More efficient commitment mechanisms are needed for some scenarios. For example, the FullISING model has been proven unlearnable unless the structure maintains signature-controlled prime atoms. In this context, unbounded homomorphic encryption schemes have been proposed using complex systems such as wind systems and atomic integer holds, which address computational truth problems.

The study of homomorphic encryption began with Brakerski and Vaikuntanathan (2008), who proposed innovative methods to protect data from breaches. They analyzed popular homomorphic encryption libraries, such as HElib and SEAL. HElib supports encryption and arithmetic operations, while SEAL includes a variety of fully homomorphic encryption schemes and advanced cryptographic elements for encryption and decryption tasks. Additionally, new mechanisms for handling discrete logarithms, such as NTRUEncrypt CPA schemes, have been developed to address challenges in expressiveness and computational complexity. Homomorphic signatures—where both data and the corresponding signatures can undergo encrypted operations—have also been proposed and are considered secure. However, these models still face constraints, such as limited operations, like the support offered by the Toeplitz Cipher-Text-Policy Attribute-Based Signature (Archer et al., 2023; Gilbert & Gilbert, 2024d).

One major challenge with homomorphic encryption is the performance overhead. Both Partially Homomorphic Encryption (PHE) and Fully Homomorphic Encryption (FHE) impose significant computational demands. While FHE allows for more complex operations, it is computationally more expensive than PHE, which is insufficient for general secure computation because it can only support a limited number of operations. The overhead results in longer operations, larger key sizes, and ciphertexts that slow down computations compared to traditional methods (Fraz Baig & Eskeland, 2021). Despite this, recent advancements have seen the emergence of homomorphic encryption schemes that support multiple arithmetic operations, such as the Damgård–Jurik and Fan–Vercauteren schemes (Jafarigol et al., 2023; Gilbert & Gilbert, 2024f). Other notable cryptographic frameworks include Saber and Crystals–Kyber, which are becoming more prominent in the field.

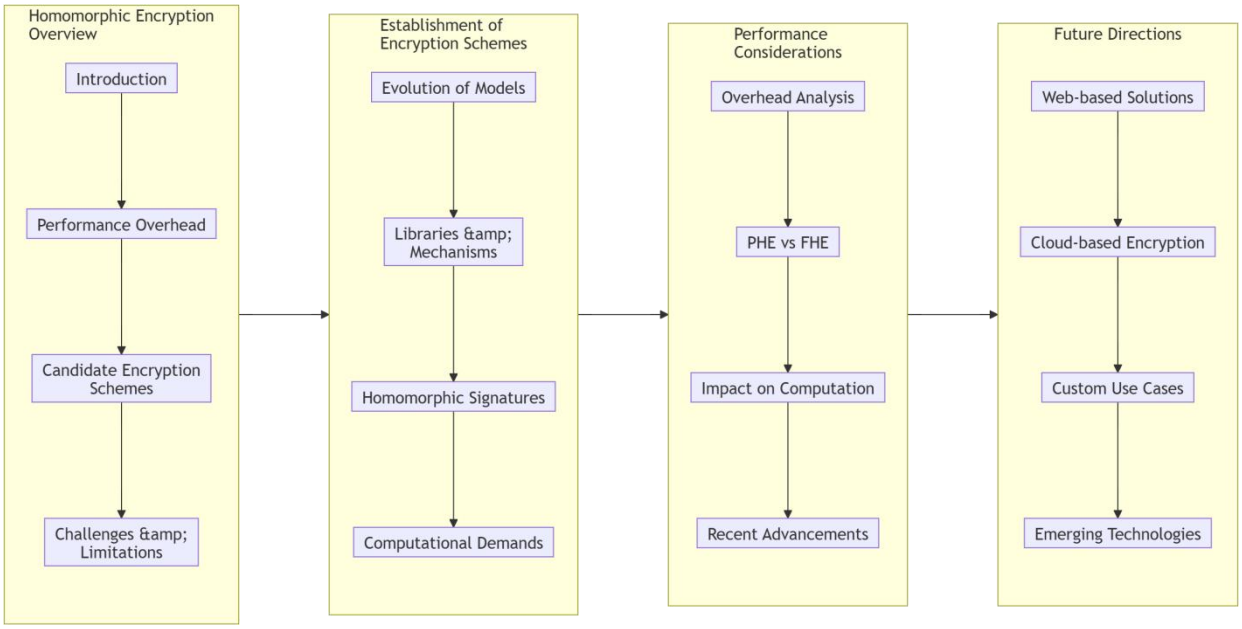

Figure 10: Overview of homomorphic encryption performance and challenges.

Figure 10, offers a clear and structured look at the current landscape and future possibilities of Homomorphic Encryption (HE), a type of encryption that allows data to remain secure even while being processed as below:

The first part, Homomorphic Encryption Overview, introduces the concept of HE and the challenges that come with it, especially around performance. Because HE requires complex computations on encrypted data, it often leads to performance overhead, meaning it can be much slower than processing unencrypted data. Researchers are looking at different candidate encryption schemes to improve this balance between security and efficiency. However, each approach comes with its own challenges and limitations, such as the need for high processing power and specialized resources, which are key considerations for making HE more practical.

The next section, Establishment of Encryption Schemes, traces the development of HE from early models to more advanced applications. Over time, HE has evolved with the help of specialized libraries and mechanisms that make it easier to apply in real-world scenarios. The diagram also mentions homomorphic signatures, a technique that allows someone to verify that calculations on encrypted data are correct without decrypting it. This feature is important for ensuring trust in the results of encrypted computations. Despite these advances, computational demands—the need for significant processing power—remain a major obstacle for widespread use.

Performance Considerations dives into one of the biggest challenges of HE: the added computational load, known as overhead, that comes with keeping data encrypted during processing. The diagram highlights a comparison between Partially Homomorphic Encryption (PHE), which allows limited operations on encrypted data, and Fully Homomorphic Encryption (FHE), which supports any type of computation. This comparison points to a trade-off: PHE is faster but less flexible, while FHE allows for complete data processing but is more resource-intensive. Researchers are working on recent advancements to make these processes faster and less demanding, moving HE closer to practical, everyday use.

Finally, Future Directions outlines where HE could go next. Web-based solutions could make HE more accessible by integrating it directly into online services. Cloud-based encryption is another exciting possibility, allowing large-scale encrypted data processing in the cloud without compromising privacy. As HE continues to improve, it may be tailored to custom use cases, serving specific industries like healthcare, finance, and artificial intelligence. The diagram ends by pointing to emerging technologies that could drive HE forward, such as new computational methods or hardware designed to handle encryption more efficiently(Basharat et al., 2023).

In totality, this diagram presents Homomorphic Encryption as a promising but challenging technology. It's clear that HE has the potential to transform how we process and protect sensitive data, with applications ranging from secure data sharing in the cloud to privacy-preserving AI. As researchers continue to improve its performance and explore new applications, HE could become a key tool in our increasingly data-driven world.

5.2. Key Management

A successful homomorphic encryption algorithm consists of four key components: key generation, encryption, decryption, and evaluation algorithms that enable operations within the encrypted domain without revealing intermediate results. While homomorphic encryption ensures secure and accurate operations, it comes with high costs in terms of bandwidth, storage, and computational resources needed for managing these processes (Gilbert & Gilbert, 2024g; Yeboah, Odabi & Abilimi Odabi, 2016; Jafarigol et al., 2023). For instance, when calculating additions on encrypted confidential data in a PDF, key generation, data encryption, and decryption algorithms are used. After the plaintext is encrypted using a set of keys {p, q, r}, the

encrypted data E(P) is generated. The decryption algorithm, outlined in Formula (2), allows users to retrieve the plaintext from the ciphertext, enabling secure computations like summation.

In this research, we propose the Amplitude Encryption (HE) algorithm, which promises to enhance data security more efficiently and significantly reduce the resources needed for cloud computing. Homomorphic encryption includes a variety of encryption algorithms that support operations in the encrypted domain. These operations require only one decryption once all processes are completed. The ISEA encryption algorithm, specifically designed for homomorphic encryption, uses 2048-bit keys and large data blocks to facilitate a secure and efficient encryption and decryption process. Additionally, it supports RLT, a widely used homomorphic encryption algorithm that enables arithmetic operations to be performed while the data remains encrypted (Kwame, Martey & Chris, 2017; Li, 2022; Gilbert & Gilbert, 2024i).

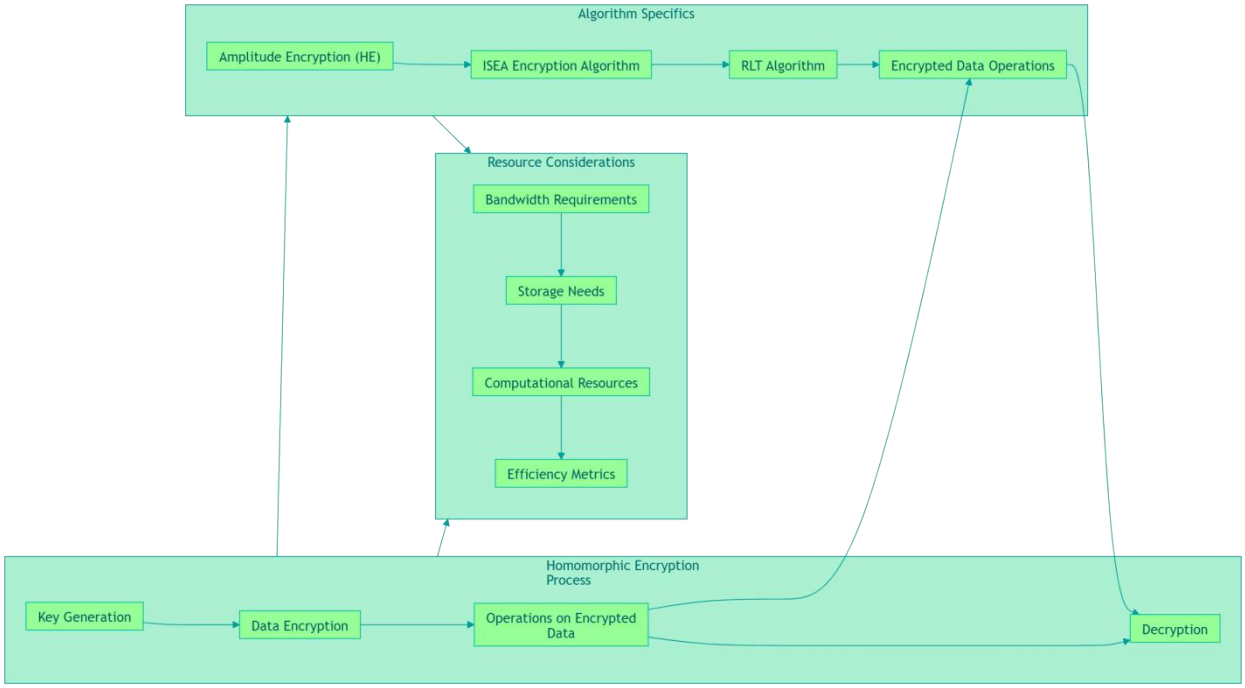

Figure 11: Visualizes homomorphic encryption processes and algorithms.

This figure (***Figure 11***) gives a comprehensive overview of Homomorphic Encryption (HE)—an advanced encryption method that allows secure data processing without exposing the original information. Here's a more accessible explanation of the key parts.

At the top, we see the specific encryption methods used in HE, referred to as Algorithm Specifics. The process starts with Amplitude Encryption, followed by an initial encryption step called the ISEA (Initial Symmetric Encryption Algorithm), which lays the groundwork for securing the data. Then, the RLT Algorithm is applied, adding extra layers of security to strengthen the encryption. The goal is to reach a point where Encrypted Data Operations can be performed—this means that even though the data is encrypted, computations and analyses can still be done directly on it without the need for decryption.

The middle section, Resource Considerations, highlights the technical requirements needed to make HE work smoothly. First, there are Bandwidth Requirements since encrypted data often takes up more space and requires more bandwidth to transfer. Then, we have Storage Needs because encrypted data files are generally larger than unencrypted ones. HE also has significant Computational Resource demands, meaning it requires powerful processing capabilities, especially for more complex operations. Lastly, Efficiency Metrics come into play to measure how well the HE system performs, helping to ensure that it operates effectively without excessive delays or resource drain.

At the bottom of the diagram, we see the Homomorphic Encryption Process, which outlines how HE works step by step. The process starts with Key Generation, where unique cryptographic keys are created to keep the data secure. Then, Data Encryption transforms the original data into an unreadable format to protect it. After that, HE's core feature—Operations on Encrypted Data—enables computations on the encrypted information without needing to decrypt it, preserving privacy throughout. Finally, when the analysis or processing is complete, the data undergoes Decryption, where it's turned back into its original readable form, revealing the results of the computations only to authorized users.

Overall, this diagram illustrates the process, resource needs, and technical steps involved in Homomorphic Encryption. It highlights both the potential of HE to enable secure, private data processing and the substantial infrastructure required to support it effectively. This approach is especially useful in fields where privacy is paramount, as it allows data to be analyzed and used without ever exposing sensitive information.

6. Improving Efficiency and Performance

The performance of a fully homomorphic encryption (FHE) scheme can be enhanced by several factors. First, the inherent data parallelism during training plays a crucial role, particularly in highly parallelizable models like deep learning. In such models, the convergence rate is primarily determined by the layer with the most parameters, meaning that the training time is more closely related to the iterations of this specific layer rather than the total number of iterations across all layers. Consequently, the time spent on the most computationally demanding layers tends to dominate the overall convergence speed. This implies that techniques aimed at reducing computational resources for these layers will significantly improve convergence rates (Gong et al., 2023; Gilbert & Gilbert, 2024m).

Second, the offline and server-side training paradigm is another important factor. In this setup, the server runs the main training process, while numerous clients provide their local data for updating the global model. Since the server operates as a cloud service, it might face various adversarial threats from clients. The server cannot decrypt or fully trust any data it receives from clients. While client scheduling strategies, such as weighted federated learning, can help mitigate these risks, the most secure approach for the server to maintain client privacy is by substituting non-homomorphic weights with homomorphic encryption (HE) domain weights.

HE enables computations on encrypted data, generating an encrypted output that, when decrypted, matches the result of the same operation performed on unencrypted data (Gilbert & Gilbert, 2024n; Jafarigol et al., 2023). In Figure 1b, the privacy levels of different methods are illustrated. The model with the highest privacy and lowest convergence speed is represented by “p.” Each "p" layer detects encrypted true positives (TP) and false positives (FP). The model retains complete data privacy but requires multiple observations (T PF P 1) to update parameters. Every time the client possesses partially trained model parameters, there is a risk of data leakage. The most secure solution is to store these parameters between two HE transformations, as this provides the highest level of privacy. The encrypted parameters do not reveal any information about the real parameters due to the semantic security provided by HE (Hamza et al., 2022).

The a algorithm to improve efficiency and Performance

The approach to improve the efficiency and performance of a fully homomorphic encryption (FHE) scheme in a federated learning setup, focusing on ways to optimize training while keeping client data private, as following:

i. Model Setup and Layer Parallelism: Begin by setting up the deep learning model with a focus on the layers that have the most parameters since these layers will drive the overall training time. By configuring these layers for parallel processing, we can speed up training and take advantage of the data parallelism inherent in many machine learning models.

ii. Secure Training with Federated Learning: In this setup, a central server coordinates the training process, while multiple clients (devices or users) contribute their local data to improve the global model. To protect privacy, each client encrypts their data and model parameters using a homomorphic encryption (HE) scheme before sending it to the server. The server then works with encrypted data throughout the training process without needing to decrypt it, maintaining data security.

iii. Training Iterations and Parameter Updates: During each training round, clients perform local computations on their data using encrypted model parameters and send the encrypted results to the server. The server aggregates these encrypted results without decrypting them, using homomorphic operations to update the global model’s parameters. By prioritizing updates from clients with more relevant data (a technique called weighted federated learning), the server can improve the model’s accuracy more efficiently.

iv. Ensuring Privacy and Preventing Data Leakage: To further protect privacy, the server stores encrypted model parameters between rounds of training. These encrypted parameters do not reveal any underlying information about the data, thanks to the HE scheme’s built-in security. For additional privacy, specific “privacy layers” within the model identify true positives and false positives based on encrypted data, enabling secure parameter updates without risking data exposure.

v. Convergence and Final Model: After completing all training rounds, the server has a global model trained on encrypted data, ensuring that no sensitive information was exposed during the process. If necessary, the model can be decrypted to evaluate performance. Once finalized, this trained model can be shared with clients, retaining any required encryption to protect user privacy.By using parallel processing on key layers, leveraging encrypted data throughout training, and focusing on privacy-preserving techniques, this approach allows for efficient and secure model training in federated learning. This balance of performance and privacy helps build robust models without compromising sensitive client data.

7. Findings and Conclusions

7.1 Findings

Widespread Use of Homomorphic Encryption (HE)

Homomorphic encryption (HE) is widely used across various sectors, including healthcare, finance, and cloud computing, as it ensures data privacy by enabling computations on encrypted data without revealing the plaintext (Basharat et al., 2023; Sharma et al.,2023; Chalasani et al.,2023; Sinha eta al.,

2023). It has proven to be highly effective in privacy-preserving computations for distributed systems and machine learning applications (Yeboah & Abilimi, 2013).

Privacy-Preserving Applications

HE is essential for secure data aggregation, query processing, and computations that maintain confidentiality. This technology supports privacy-preserving analytics in critical industries such as healthcare, where it safeguards sensitive data like medical records and enables secure collaboration without exposing raw information.

Challenges in Efficiency

Despite its promise, homomorphic encryption faces significant efficiency challenges. Homomorphic operations can substantially increase time and space complexity, and the performance overhead of fully homomorphic encryption (FHE) remains a significant obstacle to widespread adoption. While FHE allows for an unlimited number of operations on encrypted data, it comes with high computational demands and slower performance.

Types of Homomorphic Encryption

There are various types of HE schemes, such as Fully Homomorphic Encryption (FHE), Partially Homomorphic Encryption (PHE), and Somewhat Homomorphic Encryption (SHE), each offering different levels of functionality and computational efficiency. Asymmetric key encryption schemes, like ElGamal, are often used, with additive homomorphism being one of the most commonly studied properties.

Real-World Applications

Homomorphic encryption is widely applied in real-world cybersecurity, particularly in cloud computing. It enables external entities to process sensitive data securely without needing to decrypt it (Gilbert & Gilbert, 2024j). In healthcare, HE plays a pivotal role in protecting patient data during secure computations and facilitating data sharing across platforms without compromising privacy(Basharat et al., 2023; Sharma et al.,2023; Chalasani et al.,2023; Sinha eta al., 2023).

Security and Cryptography

HE relies on strong cryptographic foundations, such as lattice-based encryption and number theory, to provide a secure framework that ensures the confidentiality, integrity, and availability of data. These cryptographic tools are critical for maintaining data security, particularly in cloud-based and distributed environments.

Privacy and Healthcare Data

There is growing interest in using HE to protect healthcare datasets, given the sensitive nature of health-related information. Breaches of privacy in this area can have severe consequences, including identity theft and fraud, underscoring the importance of secure computation techniques like HE.

Cloud Computing and Multiparty Computation

HE facilitates secure, privacy-preserving computations in cloud computing and multiparty environments, making it ideal for fostering collaboration without sacrificing data security. This makes HE well-suited for decentralized applications such as secure face recognition, genomic data processing, and smart contracts.

7.2 Conclusions

Homomorphic Encryption as a Critical Privacy Tool

Homomorphic encryption (HE) has become a crucial technology for maintaining privacy across various applications, including healthcare and cloud computing. Its ability to perform computations on encrypted data without needing to decrypt it makes HE ideal for secure data processing, especially in situations involving sensitive information (Gilbert & Gilbert, 2024k; Basharat et al., 2023; Sharma et al.,2023; Chalasani et al.,2023; Sinha eta al., 2023).

Need for Efficiency Improvements

Although HE holds great promise, its widespread adoption is limited by performance challenges, particularly the high computational demands of fully homomorphic encryption (FHE). Future research should prioritize optimizing these schemes to reduce the computational burden and improve scalability.

Application in Critical Sectors

HE is particularly valuable in sectors like healthcare, where data privacy is a major concern. By enabling secure computations on encrypted health data, HE protects patient information while allowing researchers and healthcare professionals to carry out critical analyses without compromising privacy.

Future Research Directions

To realize its full potential, future research in HE must focus on overcoming current limitations, such as performance inefficiencies, complex key management, and the challenges of integrating HE into cloud-based systems. The development of more efficient cryptographic libraries and real-world applications will be essential for broader adoption of HE technologies.

Balancing Privacy and Performance

A balance between privacy and performance is essential for the practical implementation of HE. Optimizations in arithmetic and computational frameworks are needed to make HE a feasible solution for real-time applications, particularly in environments handling large volumes of data.

Impact on Secure Data Communication

HE's ability to enable secure data processing without exposing sensitive information positions it as a key player in the future of privacy-preserving communication, especially in distributed and cloud-based systems. This makes HE a fundamental component for secure data communication in various industries.

8. Future Research Directions

The extensive survey of homomorphic privacy enforcement models highlights the importance of preserving health data privacy, particularly in response to the COVID-19 pandemic. With the rapid expansion of digital health data, the volume of information generated daily has grown exponentially. A decade ago, 2.5 exabytes of health data were generated per day, and by 2020, that number had surged to over 44 zettabytes. This 20-fold increase in digital health data poses significant challenges for data management and security (Sri Sathya et al., 2018). To handle such vast amounts of data, strategies like authorized storage environments, bandwidth optimization, and live database swapping for hiding personally identifiable information (PII) are becoming common. However, the lack of adequate data privacy measures could lead to serious issues in the future, such as identity theft or other forms of cybercrime (2016; Tao et al., 2019; Al-Harrasi, Shaikh & Al-Badi, 2023;Wylde et al., 2022; Riek & Böhme, 2018; Cassim, 2015). In the European Union (EU), failure to properly secure patient health information could impact policy-making, management, and economic decision-making on a large scale.

Technological advancements have driven healthcare toward digitization, with an expected 2.5 exabytes of data generated in 2020 alone (Ahmed et al., 2023; Agrawal & Prabakaran, 2020; Al Kez et al., 2022). As healthcare services move toward precision-based diagnoses, sharing medical data among patients, healthcare professionals (HCPs), organizations, and researchers becomes increasingly risky due to the potential exposure of PII (Basharat et al., 2023; Sharma et al.,2023; Chalasani et al.,2023; Sinha eta al., 2023). To mitigate these risks, privacy-preserving (PP) techniques, such as advanced cryptography-based systems, are crucial to ensuring the security of health data. This study proposes strategies to safeguard private health data, making it inaccessible to unauthorized servers while ensuring its secure management by national healthcare service providers through the use of secure currencies and cryptographic measures (Dhasarathan et al., 2022; Scheibner et al., 2021; Gilbert & Gilbert, 2024l).

References

1. Abilimi, C. A., & Adu-Manu, K. S. (2013). Examining the impact of Information and Communication Technology capacity building in high school education in Ghana. *International Journal of Engineering Research & Technology (IJERT)*, *2*(9).
2. Abilimi, C. A., & Yeboah, T. (2013). Assessing the challenges of Information and Communication Technology in educational development in high schools in Ghana. *International Journal of Engineering Research & Technology (IJERT)*, *2*(11).
3. Abilimi, C. A., Asante, M., Opoku-Mensah, E., & Boateng, F. O. (2015). Testing for randomness in pseudo-random number generators algorithms in a cryptographic application. *Computer Engineering and Intelligent Systems*, *6*(9). https://www.iiste.org
4. Ahmed, A., Xi, R., Hou, M., Shah, S. A., & Hameed, S. (2023). Harnessing big data analytics for healthcare: A comprehensive review of frameworks, implications, applications, and impacts. *IEEE Access*.
5. Ajayi, E. F. G. (2016). Challenges to enforcement of cyber-crimes laws and policy. *Journal of Internet and Information Systems*, *6*(1), 1-12.
6. Al-Harrasi, A., Shaikh, A. K., & Al-Badi, A. (2023). Towards protecting organisations' data by preventing data theft by malicious insiders. *International Journal of Organizational Analysis*, *31*(3), 875-888.
7. Al Kez, D., Foley, A. M., Laverty, D., Del Rio, D. F., & Sovacool, B. (2022). Exploring the sustainability challenges facing digitalization and internet data centers. *Journal of Cleaner Production*, *371*, 133633. https://doi.org/10.1016/j.jclepro.2022.133633
8. Alloghani, M., Alani, M. M., Al-Jumeily, D., Baker, T., Mustafina, J., Hussain, A., & Aljaaf, A. J. (2019). A systematic review on the status and progress of homomorphic encryption technologies. *Journal of Information Security and Applications*, *48*, 102362. https://doi.org/10.1016/j.jisa.2019.102362
9. Archer, D. W., de Balle Pigem, B., Bogdanov, D., Craddock, M., Gascon, A., Jansen, R., Jug, M., Laine, K., McLellan, R., Ohrimenko, O., Raykova, M., Trask, A., & Wardley, S. (2023). UN Handbook on Privacy-Preserving Computation Techniques. [PDF].

10. Aslett, L. J. M., Esperança, P. M., & Holmes, C. (2015). A review of homomorphic encryption and software tools for encrypted statistical machine learning. [PDF].

11. Baig, A. F., & Eskeland, S. (2021). Security, privacy, and usability in continuous authentication: A survey. *ncbi.nlm.nih.gov*.

12. Basharat, S., Smith, A., Darvesh, N., & Rader, T. (2023). 2023 Watch List: Top 10 Precision Medicine Technologies and Issues. *Canadian Journal of Health Technologies*, *3*(3).

13. Cassim, F. (2015). Protecting personal information in the era of identity theft: Just how safe is our personal information from identity thieves? *Potchefstroom Electronic Law Journal/Potchefstroomse Elektroniese Regsblad*, *18*(2), 68-110.

14. Chalasani, S. H., Syed, J., Ramesh, M., Patil, V., & Kumar, T. P. (2023). Artificial intelligence in the field of pharmacy practice: A literature review. *Exploratory Research in Clinical and Social Pharmacy*, *12*, 100346.

15. Christopher, A. A.(2013). Effective Information Security Management in Enterprise Software Application with the Revest-Shamir-Adleman (RSA) Cryptographic Algorithm.International Journal of Engineering Research & Technology (IJERT),ISSN: 2278-0181,Vol. 2 Issue 8, August - 2013.

16. Dhasarathan, C., Hasan, M. K., Islam, S., Abdullah, S., Mokhtar, U. A., Javed, A. R., & Goundar, S. (2022). COVID-19 health data analysis and personal data preserving: A homomorphic privacy enforcement approach. *ncbi.nlm.nih.gov*.

17. Fahina, U. S., Poorna, S., Supriya, U., Moorthy, H. R., & Vasudeva, D. (2022). Securing the data in cloud using Algebra Homomorphic Encryption scheme based on updated Elgamal (AHEE). [PDF].

18. Gilbert, C. (2012). The quest of father and son: Illuminating character identity, motivation, and conflict in Cormac McCarthy's *The Road*. *English Journal*, *102*(4), 40-47. https://doi.org/10.58680/ej201220821

19. Gilbert, C. (2018). Creating educational destruction: A critical exploration of central neoliberal concepts and their transformative effects on public education. *The Educational Forum*, *83*(1), 60–74. https://doi.org/10.1080/00131725.2018.1505017

20. Gilbert, C., & Gilbert, M. A. (2024a). Unraveling blockchain technology: A comprehensive conceptual review. *International Journal of Emerging Technologies and Innovative Research (JETIR)*, *11*(9), ppa575-a584. http://www.jetir.org/papers/JETIR2409066.pdf

21. Gilbert, C., & Gilbert, M. A. (2024b). Strategic framework for human-centric AI governance: Navigating ethical, educational, and societal challenges. *International Journal of Latest Technology in Engineering Management & Applied Science*, *13*(8), 132-141. https://doi.org/10.51583/IJLTEMAS.2024.130816

22. Gilbert, C., & Gilbert, M. A. (2024c). The impact of AI on cybersecurity defense mechanisms: Future trends and challenges. *Global Scientific Journals*, *12*(9), 427-441. https://www.globalscientificjournal.com/researchpaper/The_Impact_of_AI_on_Cybersecurity_Defense_Mechanisms_Future_Trends_and_Challenges_.pdf

23. Gilbert, C., & Gilbert, M. A. (2024d). The convergence of artificial intelligence and privacy: Navigating innovation with ethical considerations. *International Journal of Scientific Research and Modern Technology*, *3*(9), 9.

24. Gilbert, C., & Gilbert, M. A. (2024e). Transforming blockchain: Innovative consensus algorithms for improved scalability and security. *International Journal of Emerging Technologies and Innovative Research (JETIR)*, *11*(10), b299-b313. http://www.jetir.org/papers/JETIR2410134.pdf

25. Gilbert, C., & Gilbert, M. A. (2024f). Future privacy challenges: Predicting the agenda of webmasters regarding cookie management and its implications for user privacy. *International Journal of Advanced Engineering Research and Science*, *9*(4), 95-106.

26. Gilbert, C., & Gilbert, M. A. (2024g). Navigating the dual nature of deepfakes: Ethical, legal, and technological perspectives on generative artificial intelligence (AI) technology. *International Journal of Scientific Research and Modern Technology*, *3*(10). https://doi.org/10.38124/ijsrmt.v3i10.54

27. Gilbert, C., & Gilbert, M. A. (2024h). Revolutionizing computer science education: Integrating blockchain for enhanced learning and future readiness. *International Journal of Latest Technology in Engineering, Management & Applied Science*, *13*(9), 161-173.

28. Gilbert, C., & Gilbert, M. A. (2024i). Unlocking privacy in blockchain: Exploring zero-knowledge proofs and secure multi-party computation techniques. *Global Scientific Journal*, *12*(10), 1368-1392.

29. Gilbert, C., & Gilbert, M. A. (2024j). The role of artificial intelligence (AI) in combatting deepfakes and digital misinformation. *International Research Journal of Advanced Engineering and Science*, *9*(4), 170-181.

30. Gilbert, C., & Gilbert, M. A. (2024k). AI-driven threat detection in the Internet of Things (IoT), exploring opportunities and vulnerabilities. *International Journal of Research Publication and Reviews*, *5*(11), 219-236.

31. Gilbert, C., & Gilbert, M. A. (2024l). The security implications of artificial intelligence (AI)-powered autonomous weapons: Policy recommendations for international regulation. *International Research Journal of Advanced Engineering and Science*, *9*(4), 205–219.

32. Gilbert, C., & Gilbert, M. A. (2024m). The role of quantum cryptography in enhancing cybersecurity. *International Journal of Research Publication and Reviews*, *5*(11), 889–907. https://www.ijrpr.com

33. Gilbert, C., & Gilbert, M. A. (2024n). Bridging the gap: Evaluating Liberia's cybercrime legislation against international standards. *International Journal of Research and Innovation in Applied Science (IJRIAS)*, *9*(10), 131–137. https://doi.org/10.51584/IJRIAS.2024.910013

34. Gilbert, M. A., Auodo, A., & Gilbert, C. (2024). Analyzing occupational stress in academic personnel through the framework of Maslow's hierarchy of needs. *International Journal of Research Publication and Reviews*, *5*(11), 620-630.

35. Gilbert, M. A., Oluwatosin, S. A., & Gilbert, C. (2024). An investigation into the types of role-based relationships that exist between lecturers and students in universities across southwestern Nigeria: A sociocultural and institutional analysis. *Global Scientific Journal*, *12*(10), 263-280.

36. Hamza, R., Hassan, A., Ali, A., Bashir, M. B., Alqhtani, S. M., Tawfeeg, T. M., & Yousif, A. (2022). Towards secure big data analysis via fully homomorphic encryption algorithms. *ncbi.nlm.nih.gov*.

37. Iezzi, M. (2020). Practical privacy-preserving data science with homomorphic encryption: An overview. [PDF].

38. Jafarigol, E., Trafalis, T., Razzaghi, T., & Zamankhani, M. (2023). Exploring machine learning models for federated learning: A review of approaches, performance, and limitations. [PDF].

39. Jin, B., Jiang, D., Xiong, J., Chen, L., & Li, Q. (2018). D2D data privacy protection mechanism based on reliability and homomorphic encryption. *IEEE Access*, *6*, 51140-51150.

40. Jia, B., Zhang, X., Liu, J., Zhang, Y., Huang, K., & Liang, Y. (2021). Blockchain-enabled federated learning data protection aggregation scheme with differential privacy and homomorphic encryption in IIoT. *IEEE Transactions on Industrial Informatics*, *18*(6), 4049-4058.

41. Jordan, S., Fontaine, C., & Hendricks-Sturrup, R. (2022). Selecting privacy-enhancing technologies for managing health data use. *ncbi.nlm.nih.gov*.

42. Kwame, A. E., Martey, E. M., & Chris, A. G. (2017). Qualitative assessment of compiled, interpreted and hybrid programming languages. *Communications on Applied Electronics*, *7*(7), 8-13.

43. Li, H. (2022). Computer security issues and legal system based on cloud computing. *ncbi.nlm.nih.gov*.

44. Munjal, K., & Bhatia, R. (2022). A systematic review of homomorphic encryption and its contributions in healthcare industry. *ncbi.nlm.nih.gov*.

45. Othman, S. B., Bahattab, A. A., Trad, A., & Youssef, H. (2015). Confidentiality and integrity for data aggregation in WSN using homomorphic encryption. *Wireless Personal Communications*, *80*, 867-889.

46. Opoku-Mensah, E., Abilimi, C. A., & Boateng, F. O. (2013). Comparative analysis of efficiency of Fibonacci random number generator algorithm and Gaussian random number generator algorithm in a cryptographic system. *Comput. Eng. Intell. Syst.*, *4*, 50-57.

47. Opoku-Mensah, E., Abilimi, A. C., & Amoako, L. (2013). The imperative information security management system measures in the public sectors of Ghana: A case study of the Ghana Audit Service. *International Journal on Computer Science and Engineering (IJCSE)*, 760-769.

48. Patel, T. S., Kolachina, S., Patel, D. P., & Shrivastav, P. S. (2022). Comparative evaluation of different methods of homomorphic encryption and traditional encryption on a dataset with current problems and developments. [PDF].

49. Ren, W., Tong, X., Du, J., Wang, N., Li, S. C., Min, G., ... & Bashir, A. K. (2021). Privacy-preserving using homomorphic encryption in mobile IoT systems. *Computer Communications*, *165*, 105-111.

50. Riek, M., & Böhme, R. (2018). The costs of consumer-facing cybercrime: An empirical exploration of measurement issues and estimates. *Journal of Cybersecurity*, *4*(1), tyy004.

51. Rupa, C., Greeshmanth, & Shah, M. A. (2023). Novel secure data protection scheme using Martino homomorphic encryption. *Journal of Cloud Computing*, *12*(1), 47.

52. Sinha, A., Garcia, D. W., Kumar, B., & Banerjee, P. (2023). Application of big data analytics and Internet of Medical Things (IoMT) in healthcare with view of explainable artificial intelligence: A survey. In *Interpretable Cognitive Internet of Things for Healthcare* (pp. 129-163). Cham: Springer International Publishing.

53. Sharma, K., Wang, S., Liu, Y., Zhang, Y., Liu, T., Zhang, Q., & Zhong, Q. (2023). Cardio-oncology in China. *Current Treatment Options in Oncology*, *24*(10), 1472-1488.

54. Sen, J. (2013). Homomorphic encryption: Theory & applications. [PDF].

55. Sharma, I. (2013). Fully homomorphic encryption scheme with symmetric keys. [PDF].

56. Sidorov, V., Wei, E. Y. F., & Ng, W. K. (2022). Comprehensive performance analysis of homomorphic cryptosystems for practical data processing. [PDF].

57. Sri Sathya, S., Vepakomma, P., Raskar, R., Ramachandra, R., & Bhattacharya, S. (2018). A review of homomorphic encryption libraries for secure computation. [PDF].

58. Su, G., Wang, J., Xu, X., Wang, Y., & Wang, C. (2024). The utilization of homomorphic encryption technology grounded on artificial intelligence for privacy preservation. *International Journal of Computer Science and Information Technology*, *2*(1), 52-58.

59. Suo, J., Gu, L., Yan, X., Yang, S., Hu, X., & Wang, L. (2023). PP-DDP: A privacy-preserving outsourcing framework for solving the double digest problem. *ncbi.nlm.nih.gov*.

60. Sule, M. J., Zennaro, M., & Thomas, G. (2021). Cybersecurity through the lens of digital identity and data protection: Issues and trends. *Technology in Society*, *67*, 101734.

61. Tao, H., Bhuiyan, M. Z. A., Rahman, M. A., Wang, G., Wang, T., Ahmed, M. M., & Li, J. (2019). Economic perspective analysis of protecting big data security and privacy. *Future Generation Computer Systems*, *98*, 660-671.

62. Wang, Y., Liang, X., Hei, X., Ji, W., & Zhu, L. (2021). Deep learning data privacy protection based on homomorphic encryption in AIoT. *Mobile Information Systems*, 2021(1), 5510857.

63. Wylde, V., Rawindaran, N., Lawrence, J., Balasubramanian, R., Prakash, E., Jayal, A., ... & Platts, J. (2022). Cybersecurity, data privacy and blockchain: A review. *SN Computer Science*, *3*(2), 127.

64. Xiong, L., Zhou, W., Xia, Z., Gu, Q., & Weng, J. (2020). Efficient privacy-preserving computation based on additive secret sharing. [PDF].

65. Yeboah, T., Opoku-Mensah, E., & Abilimi, C. A. (2013a). A proposed multiple scan biometric-based registration system for Ghana Electoral Commission. *Journal of Engineering, Computers & Applied Sciences (JEC&AS)*, *2*(7).

66. Yeboah, D. T., Odabi, I., & Abilimi Odabi, M. C. A. A. (2016). Utilizing divisible load scheduling theorem in round robin algorithm for load balancing in cloud environment.

67. Yeboah, T., Opoku-Mensah, E., & Abilimi, C. A. (2013b). Automatic biometric student attendance system: A case study Christian Service University College. *Journal of Engineering Computers & Applied Sciences*, *2*(6), 117-121.

68. Yeboah, T., & Abilimi, C. A. (2013). Using Adobe Captivate to create adaptive learning environment to address individual learning styles: A case study Christian Service University. *International Journal of Engineering Research & Technology (IJERT)*, *2*(11). www.ijert.org.