

CTEC3451 - P2665421, Luke Dawson
Supervisor - Mehmet Kiraz

First Deliverable

Privacy-Preserving Biometric Authentication



CONTENTS

CONTENTS.....	2
LITERATURE REVIEW.....	4
Introduction.....	4
Biometric Authentication.....	5
Challenges in Biometric Authentication.....	7
Technical Foundations & Mathematical Background for FHE.....	9
Fully Homomorphic Encryption.....	10
Integration of FHE with Biometric Authentication.....	11
Current Trends and Future Research.....	13
Conclusion.....	14
FUNCTIONAL REQUIREMENTS.....	15
Introduction.....	15
User Characteristics.....	15
Guest User:.....	15
Base User:.....	16
Administration User:.....	16
System Operations.....	17
Biometric Data Management.....	17
Use Cases.....	17
Use-case Lookup Table:.....	17
UC-PPBA01: Decide on Biometric Storage and Authentication.....	18
UC-PPBA02: Check for Existing Biometrics Registration.....	18
UC-PPBA03: Input Biometric Data for Authentication.....	18
UC-PPBA04: Register Biometric Data.....	19
UC-PPBA05: Store and Encrypt Biometric Data.....	19
UC-PPBA06: Authenticate Guest User Biometrics.....	20
UC-PPBA07: Authenticate and Store Base User Biometrics.....	20
UC-PPBA08: Access Database Log (Administrator Only).....	20
UC-PPBA09: Print Authentication Result.....	21
TEST PLANS.....	21
Test Strategy.....	21
Test Objectives.....	21
Test Strategy.....	22
Functional Test Cases.....	22
Security Test Cases.....	23

Performance Test Cases.....	23
Usability Test Cases.....	24
SYSTEM DESIGN.....	25
Detailed System Overview.....	25
Focus on Back-End Development:.....	25
Front-End UI.....	26
Back-End Architecture.....	26
IMPLEMENTATION REPORT.....	28
Implementation.....	28
Security Measures.....	29
Challenges and Solutions.....	29
Future Enhancements.....	29
Conclusion.....	30
References.....	31
Appendix.....	33

LITERATURE REVIEW

Introduction

This literature review's purpose is to analyse the integration of Privacy-Preserving Biometric Authentication (PPBA) using Homomorphic Encryption (HE). In an era where data breaches and privacy violations are increasingly prevalent, the secure handling and protection of biometric data has emerged as pivotal concerns (Jain, Ross, & Prabhakar, 2004). This review aims to critically evaluate the current state of research in this field, exploring how HE, especially FHE (Fully Homomorphic Encryption) can augment the security and privacy of biometric authentication systems while upholding their efficiency and reliability.

Biometric authentication, particularly fingerprint recognition, has been widely recognised for its uniqueness and ease of application, becoming an integral part of modern security protocols (Rathgeb & Uhl, 2011). However, the widespread adoption of biometric systems across various sectors exposes them to potential cyberattacks, raising significant privacy and security issues (Ratha, Connell, & Bolle, 2001). The imperative to safeguard individual biometric data against unauthorised access and misuse stands at the forefront of technological advancements in this area. FHE, a cryptographic technique that enables computation on encrypted data without requiring decryption (Gentry, 2009), presents a novel approach to enhancing the strengths of biometric system security. This is done by integrating FHE with biometric systems; authentication processes can be performed while maintaining the encrypted state of the underlying biometric data, thereby preserving privacy, and reinforcing security (Bringer, Chabanne, & Pointcheval, 2007).

This review plans to explore various dimensions of PPBA, encompassing the challenges encountered by traditional biometric systems, the underpinnings and potential of FHE, and how their integration can effectively address prevailing security and privacy concerns. After examining existing reports, I will highlight the advancements, limitations, and prospects of the integration, contributing significantly to the discourse in cybersecurity, data privacy, and biometric authentication (Osadchy et al., 2017).

Biometric Authentication

Biometric Authentication represents a sophisticated security paradigm, leveraging unique physiological or behavioural characteristics for individual identification. This technology has gained prominence due to its enhanced security and user convenience compared to traditional authentication methods like passwords or PINs (Jain, Ross, & Prabhakar, 2004). Biometrics relies on the premise that individual traits such as fingerprints, facial features, iris patterns, and voices are distinct and difficult to replicate, thereby offering a robust authentication mechanism.

There are several types of biometric authentication, and each has its own set of applications and advantages. For example, fingerprint recognition is one of the most prevalent and studied forms and is renowned for its ease of use and high accuracy (Rathgeb & Uhl, 2011). Facial recognition technology has been recently propelled by advancements in machine learning and artificial intelligence and is increasingly being used for identity verification in smartphones and security systems. Iris recognition offers one of the highest levels of security due to the unique patterns in the human iris.

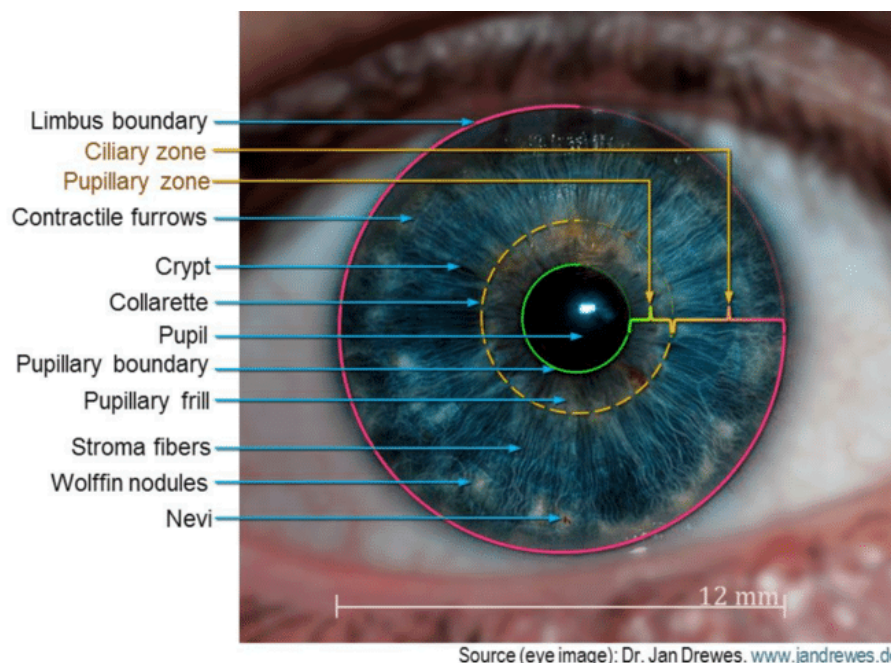


Figure 1. Shows the different patterns used when implementing Iris recognition (Research Gate)₂₀

Biometric authentication applications are extensive and diverse within security because they are used for access control in high-security areas, border control, and law enforcement. Financial sectors also leverage biometric authentication for secure banking transactions and fraud prevention. Additionally, its use in consumer electronics, particularly in smartphones and laptops for secure access, underscores its widespread acceptance and importance. One of the biggest industries that utilises biometric authentication is healthcare. The healthcare industry integrated biometrics for patient identification, ensuring accurate medical record matching and reducing errors. Furthermore, biometric systems play a crucial role in the management programs implemented by governments globally (Rtha, Connell, & Bolle, 2001).

The importance of biometric authentication comes from its ability to establish a secure and seamless method of validating identities, which is critical in today's digital atmosphere. Its applications exceed convenience, offering enhanced security for personal and organisational data. This technology is instrumental in preventing theft and ensuring access and sensitive information is shielded, which is a growing concern in our increasingly online world (Sutcu, Sencar, & Memon, 2007). Despite this, the rise in biometric usage also brings challenges, particularly in privacy and data security. The ethical collection, storage, and use of biometric data are predominant. As biometric systems become more embedded in daily transactions, continuous improvement and responsible implementation are essential to maintain the delicate balance between security and privacy rights (Osadchy et al., 2017).

Challenges in Biometric Authentication

While biometric authentication offers numerous benefits, it also faces various significant obstacles, particularly in terms of the privacy concerns, security risks, and technical limitations. Privacy concerns are paramount, as biometric data, once compromised, cannot be replaced like simple passwords or PINs (Ratha, Connell, & Bolle, 2001). This is due to the nature of biometric data being irreversible, making leakage pose a severe risk to the users' privacy. Ethical considerations are also a challenge especially surrounding the collection and storage of the delicate data, due to the potential for misuse or unauthorised surveillance (Cavoukian & Stoianov, 2009).

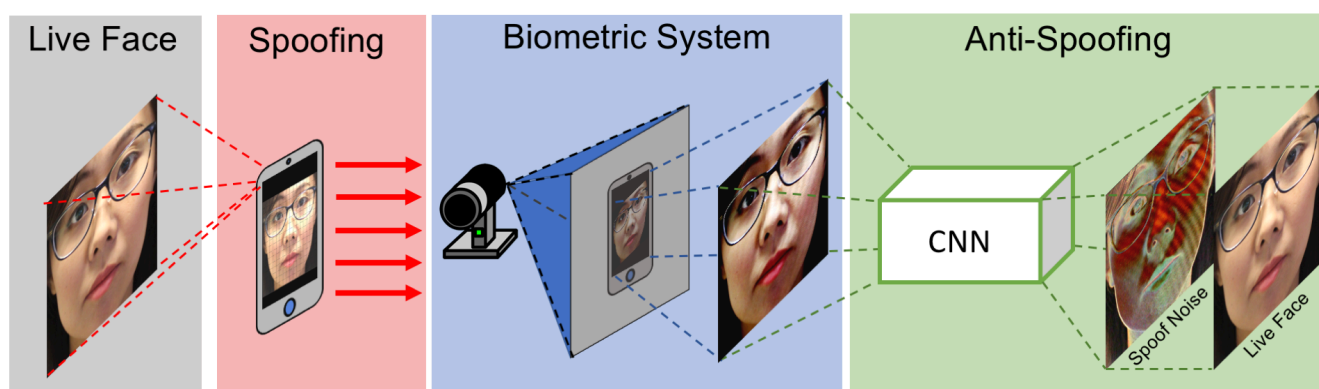


Figure 1. Shows how spoofing can be achieved through faking facial images (Research Gate)₂₁

Security risks within biometric systems are multifaceted. They encompass different threats for example, spoofing attacks, where fake biometric traits such as synthetic fingerprints or altered facial images are used to deceive the system. Additionally, biometric databases, if inadequately protected, become targets for cyberattacks, leading to massive data breaches (Osdachy et al., 2017). These risks accentuate the need for robust security protocols and continuous monitoring to detect and mitigate the potential threats.

Benefits Of Using Biometrics (%)

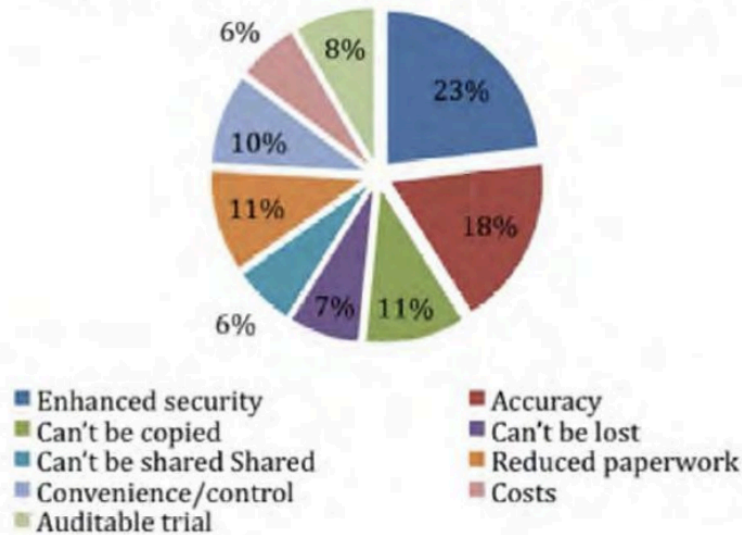


Figure 2. Strengths of Advantages (Figure from Biometric Authentication. Types of biometric identifiers)²²

Disadvantages Of Using Biometrics (%)

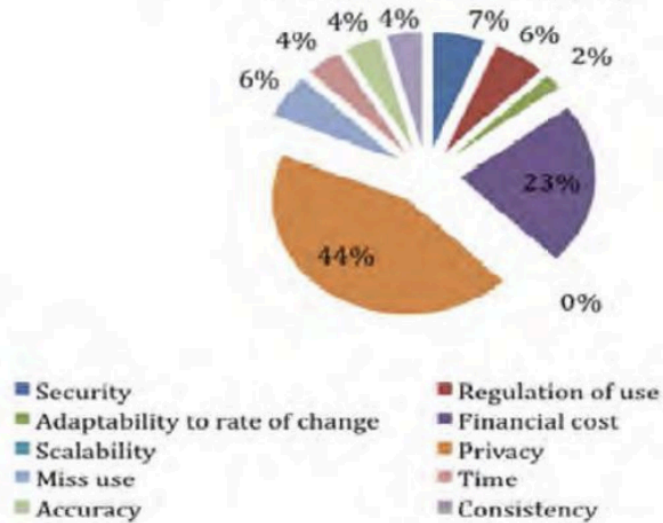


Figure 3. Impact of disadvantages (Figure from Biometric Authentication. Types of biometric identifiers)²²

Technical limitations present another layer of challenge in biometric systems. The accuracy and reliability of these systems are often reliant on the quality of the biometric sensors and environmental factors. For instance, a fingerprint scanner may fail to accurately read a print if the finger is dirty or injured, leading to false negatives. Similarly, the systems have an obligation to contend with the natural changes in biometric data over time, such as ageing in facial recognition or voice changes, which can result in false positives or negatives (Jagadeesan et al., 2010). Moreover, the need for standardisation across various platforms and devices to ensure interoperability adds to the complexity, particularly as biometric technologies continue to evolve and diversify.

Despite having many challenges, biometric authentication systems offer significant advantages and benefits. One of the key strengths is the enhanced security provided by using unique biological characteristics, making it extremely difficult for unauthorised users to replicate or forge access (Jain, Ross, & Prabhakar, 2004). This level of security is particularly beneficial in sectors where identity verification is crucial, such as banking and law enforcement as I mentioned earlier. Additionally, biometrics offer a high degree of convenience, eliminating the need to remember passwords or carry identification.

In conclusion, while biometric authentication is a powerful tool in enhancing security, the challenges it faces in terms of privacy, security, and technical aspects must be addressed completely. Overcoming these challenges is critical for the ethical and effective implementation of biometric systems across diverse sectors. As technology advances, continuous research and development are vital to address these concerns.

Technical Foundations & Mathematical Background for FHE

Before exploring Full Homomorphic Encryption (FHE), it's essential to understand the underlying technical and mathematical principles. In its most basic form, FHE is based on complex algebraic structures, particularly lattice-based cryptography. Lattice cryptography operates on the geometry of numbers and involves solving problems related to the arrangement of points at regular intervals in a multidimensional space, known as lattices (Ajtai, 1996).

The fundamental operation in lattice-based cryptography, and by extension FHE, is polynomial arithmetic (algebraic expressions that consist of variables and coefficients). The security of FHE largely hinges on the hardness of problems like the Shortest Vector Problem (SVP) and Learning with Errors (LWE) in these lattice structures (Regev, 2009). These problems involve finding short vectors within a lattice or solving linear equations with noise, respectively, both of which are a challenge when using HE. The LWE problem has emerged as a foundation for constructing various cryptographic schemes, including the creation of FHE.

In FHE, the process of encryption and decryption is like performing operations on polynomials. The encryption takes a message, represented as a polynomial, and mixes it with a form of noise within the lattice structure. This ensures that the encrypted ciphertext appears random and indistinguishable from actual randomness to an observer without the decryption key. The homomorphic properties of FHE allow for arithmetic operations, mainly addition and multiplication, to be performed directly on the ciphertexts. When decrypted, the result of these operations corresponds accurately to the result that would have been obtained if the same operations had been performed on the unencrypted plaintexts.

This background sets the stage for appreciating the complexity and ingenuity of FHE algorithms in securely processing encrypted data, a key feature that has significant implications in the realm of PPBA.

Fully Homomorphic Encryption

Fully Homomorphic Encryption (FHE) portrays a ground-breaking development in the field of cryptography, as it offers a solution to the longstanding challenge of performing computations on encrypted data without needing to decrypt it first. The foundation of FHE is its ability to allow operations on ciphertext, ensuring that the result, when decrypted, will match the outcome of the operations performed on plaintext (Paillier, 1999). This unique property of FHE ensures that sensitive data remains encrypted throughout the processing phase, providing a robust framework for data security and privacy.

Data confidentiality is paramount, using FHE accommodates for this through allowing diverse and transformative applications. As mentioned in a previous paragraph the healthcare sector is a major target point for biometric authentication, FHE enables researchers to securely analyse encrypted medical records for research without compromising patient privacy (Gentry, 2009). When integrating FHE with biometrics, the potential is particularly striking. This is due to biometric systems, while offering a secure form of authentication, often raise privacy concerns due to the sensitive nature of the data involved. Assimilation of FHE into biometric systems could allow secure data processing without risking the information exposure. For example, a biometric authentication system equipped with FHE can verify an individual's identity by processing encrypted biometric data, thereby ensuring that the user's biometric information remains confidential and secure from potential breaches (Erdogmus & Marechal, 2017).

In conclusion, FHE stands as a revolutionary tool in the realm of data security, offering unprecedented privacy-preserving capabilities. Its integration with biometric authentication systems marks a significant step towards addressing the privacy concerns inherent in biometric data processing. As this technology continues to evolve, it is poised to redefine the landscape of secure data processing and privacy-preserving computational practices.

Integration of FHE with Biometric Authentication

The integration of Full Homomorphic Encryption (FHE) with biometric authentication systems represents a significant advancement in enhancing privacy and security. Technical approaches to this integration include the application of FHE algorithms to encrypt biometric data, such as fingerprints or facial scans, before they are used or transmitted. This ensures that the biometric data remains in an encrypted form throughout the authentication process, mitigating the risk of interception or unauthorised access (Brakerski & Vaikuntanathan, 2011).

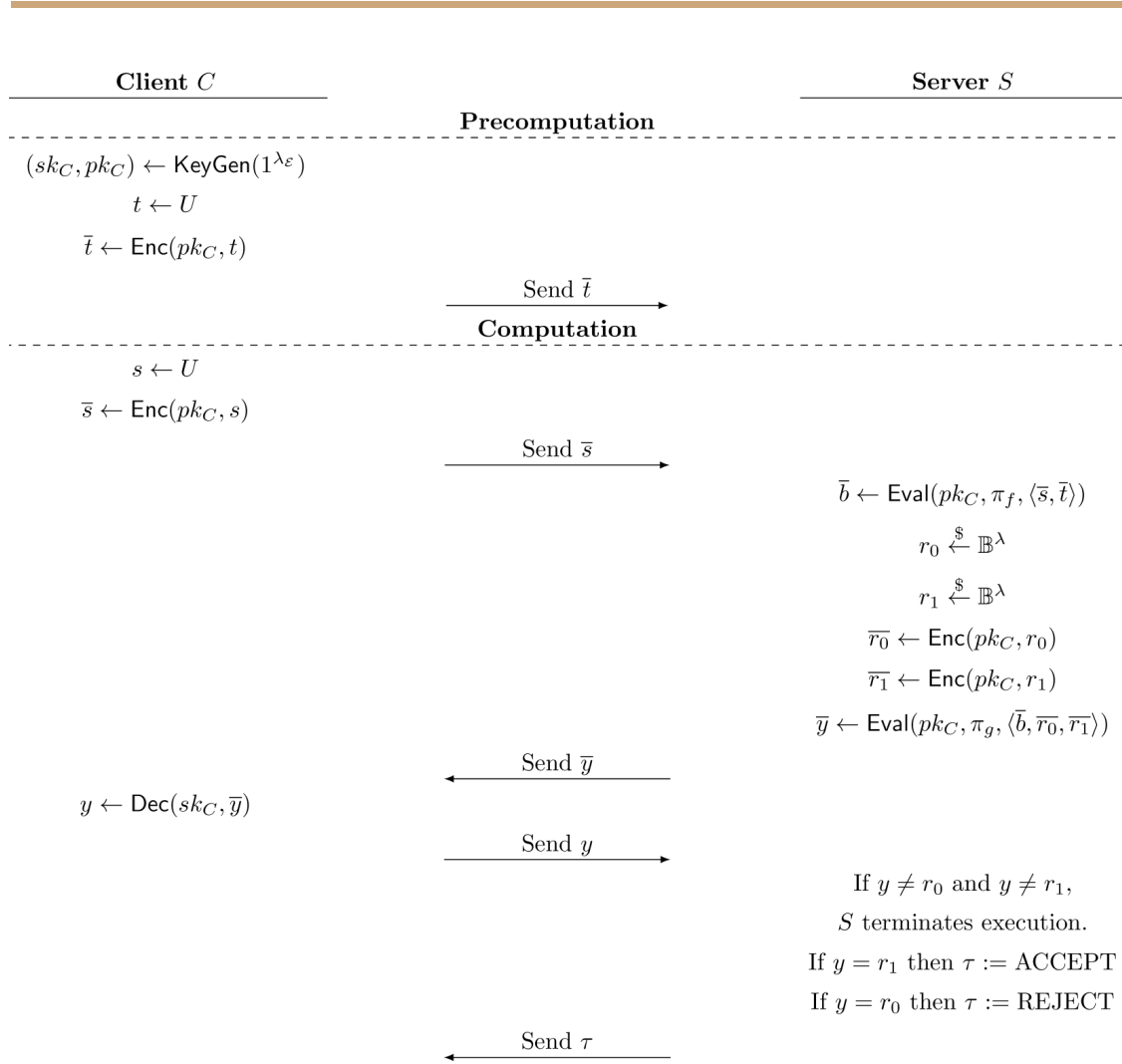


Figure 4. Integration of Full Homomorphic Encryption in Biometric Authentication Systems (Figure adapted from 'Privacy-Preserving Biometric Matching Using Homomorphic Encryption')₂₃

The above diagram illustrates a homomorphic encryption protocol adapted for biometric authentication systems, ensuring computations on biometric data while keeping privacy. Initially the client generates a public-private key pair (pk_C, sk_C) and selects a random value 't'. The client will then send 't' to the server, which then chooses its own random value 's' and encrypts it with the clients public key, this gives the output 'x'. This encrypted value is taken and then passed onto the client which will decrypted it using the private key to retrieve 'y' which should be equal to 's'.

The client will then return 'y' to the server for verification. If 'y' matches the expected outcome it is either accepted or rejected. This process allows the server to authenticate the

client using their biometric data without ever accessing the actual data in an unencrypted form.

Case studies in this sector have demonstrated the practicality and effectiveness of this integration. For example, a study conducted by Naehrig et al. (2011) explored the use of FHE in securing fingerprint biometric systems. The study illustrated how biometric data could be encrypted and then matched against an encrypted template within a secure environment, ensuring that the actual biometric data is never exposed. Another case study by Bos et al. (2013) presented a practical implementation of FHE in facial recognition systems, showcasing the potential of FHE in real-world biometric applications.

The advantages of integrating FHE with biometric authentication are immense. The primary benefit is the significant enhancement of privacy and security. Since biometric data is highly volatile, encrypting this data using FHE ensures that it remains confidential and secure against potential breaches or misuse. This integration also helps in complying with strict data protection regulations and privacy laws, which is increasingly important in today's digital society.

However, there are limitations to this integration. The most notable is the computational complexity and the consequent latency introduced by FHE operations. FHE algorithms, while secure, are computationally intensive, which can lead to slower processing times and inefficiencies in large-scale or real-time biometric systems (Gentry, Halevi, & Smart, 2012). Additionally, integrating FHE into existing biometric systems can be challenging, requiring substantial modifications to the infrastructure, and possibly leading to compatibility issues.

Current Trends and Future Research

The landscape of Privacy-Preserving Biometric Authentication (PPBA) combined with Full Homomorphic Encryption (FHE) is rapidly evolving, marked by significant recent developments and promising avenues for future research. One of the latest trends involves enhancing the computational efficiency of FHE algorithms. This improvement is crucial for practical applications, particularly in processing large biometric data (Cheon et al., 2017). Additionally, efforts are being made to develop lightweight cryptographic models that can integrate FHE into mobile and Internet of Things (IoT) devices, thereby widening its application range (Chillotti et al., 2016).

Looking ahead, the research community is focusing on devising more efficient FHE algorithms, capable of handling extensive biometric datasets with reduced processing time. Anticipating the emergence of quantum computing, there is also a notable interest in exploring quantum-resistant algorithms within FHE, ensuring long-term security and viability (Peikert, 2016). The intersection of machine learning and FHE presents another exciting research direction. Leveraging machine learning algorithms in combination with FHE could lead to more adaptive, robust, and secure biometric authentication systems, offering a significant advancement in the field (Dowlin et al., 2016).

In conclusion, the future of PPBA with FHE is set to be transformative. The ongoing research and development in this domain are not only enhancing the existing capabilities but also paving the way for innovative solutions that could revolutionise secure data processing across various industries. The integration of advanced machine learning techniques, the exploration of quantum-resistant cryptographic models, and the focus on efficiency and practicality are likely to define the next generation of PPBA systems.

Conclusion

In conclusion, this literature review has thoroughly explored the integration of Full Homomorphic Encryption (FHE) in Privacy-Preserving Biometric Authentication (PPBA), with a special insight on fingerprint recognition. This review emphasises the significance of biometric authentication in today's digital landscape, highlighting the challenges of privacy, security risks, and technical limitations. It also delves into the transformative potential of FHE in enhancing data security and privacy, particularly in biometric systems.

The implications of this project are far-reaching, indicating a change in thinking in how biometric data is processed and protected. The integration of FHE in biometric systems not only fortifies data privacy but also paves the way for authentication methods.

In conclusion, the continual advancement in cryptographic techniques, like FHE, and their application in biometric authentication, presents a promising future for cybersecurity. This integration is a critical step towards safeguarding privacy in an increasingly interconnected world, marking a significant stride in secure data processing and privacy-preserving technologies.

FUNCTIONAL REQUIREMENTS

Introduction

The Privacy-Preserving Biometric Authentication (PPBA) system is an innovative authentication solution that employs biometric data for user verification while prioritising the utmost privacy standards. The functional requirements are a crucial step in defining the system's capabilities and establishing the boundaries of its operation.

User Characteristics

The system will categorise users based on their interaction level with the system and the extent of access privileges they hold.

Guest User:

A Guest User is an individual who interacts with the system without logging in or registering. This role is only used to explore the public-facing aspects of the system with minimal access.

Privileges	
Limited Access	Guest Users can view public information within the platform without logging in or registering
Temporary Biometric Authentication	Guest Users can have their biometrics authenticated but not stored in the system permanently.
No Biometric Storage	Guest Users cannot store their biometric data within the systems database.

Base User:

The Base User is the individual who has been registered and authenticated with the PPBA system. This user type can manage personal biometric data and utilise authentication services securely.

Privileges	
Biometric Registration	Base Users can register and store their biometric data within the system.
Biometric Authentication	Base Users can authenticate using their stored biometric data to access personal services and secure areas of the system.
Data Encryption	The biometric data provided is encrypted using the FHE scheme before being stored in the database.
Access to Personalised Services	Base Users can access personalised services post-authentication.

Administration User:

The Administration User is tasked with maintaining the operational integrity of the system. This role has further applications and permissions compared to the Base User, including managing user accounts, overseeing the systems security protocols, and ensuring compliance with data protection regulations.

Privileges	
Full Biometric Management	Administrators can manage biometric data enrollment and authentication for all users.
Comprehensive Access Control	Administrators can access all levels of the system, including user management and monitoring of authentication attempts.
Database Access	Administrators have access to the database logs to review stored biometric data and oversee the system's security protocols.
System Oversight	Administrators maintain the operational integrity of the system, manage user accounts, and ensure compliance with data protection regulations.

System Operations

Biometric Data Management

- **Encryption:** The system encrypts biometric data at the point of registration using the FHE scheme, this allows for complex computations on the data whilst maintaining its encrypted state.
- **Secure Storage:** Encrypted biometric data is stored in a secure, access-controlled database. The encryption ensures that even if unauthorised access to the storage occurs the confidentiality of the biometric data is not compromised.
- **Privacy-Preserving Authentication:** During authentication, the system performs all operations on encrypted data, ensuring that the users data is not exposed in any form.

Use Cases

Use-case Lookup Table:

The table shown below can be used as a look-up guide for all the use-cases in the system.

<u>Use-Case Look-up Table</u>			
General use Cases	Use-Case Identifier	Priority	Inherited by
(Applicable to all users)	UC-PPBA01 - Decide on Biometric Storage and Authentication	High	Guest User, Base User, Administrator
	UC-PPBA02 - Check for Existing Biometric Registration	High	Guest User, Base User, Administrator
	UC-PPBA03 - Input Biometric Data for Authentication	High	Base User, Administrator
	UC-PPBA04 - Register Biometric Data	High	Base User, Administrator
	UC-PPBA05 - Store and Encrypt Biometric Data	High	Base User, Administrator

Guest User	UC-PPBA06 - Guest User Authenticate Biometrics	Medium	N/A
Base User	UC-PPBA07 - Base User Authenticate and Store Biometrics	High	Administrator
Administrator	UC-PPBA08 - Access Database Log	High	N/A
	UC-PPBA09 - Print Authentication Result	High	

Table 1: Use-case Lookup Table

UC-PPBA01: Decide on Biometric Storage and Authentication

Main Success Scenario:

1. The system presents the option to store and authenticate biometrics.
2. The user decides whether to proceed with biometric authentication or not.

Extensions:

2a. If the user decides not to proceed, the system logs the decision and ends the session.

UC-PPBA02: Check for Existing Biometrics Registration

Main Success Scenario:

1. The system inquires if the user has previously registered biometric data.
2. The user responds affirmatively or negatively.

Extensions:

2a. If the user is unsure, the system offers a way to check their registration status using identifiable information.

UC-PPBA03: Input Biometric Data for Authentication

Main Success Scenario:

1. The system prompts the user to input their biometric data for facial or fingerprint recognition.
2. The user provides the required biometric data.
3. The system processes and verifies the biometric data against the database.

Extensions:

3a. If the biometric data does not match, the system denies access and advises the user accordingly.

UC-PPBA04: Register Biometric Data**Main Success Scenario:**

1. The system asks the user if they want to register their biometric data.
2. Upon user's agreement, the system guides the user through the biometric data registration process.
3. The user completes the registration, and the system securely stores the biometric data.

Extensions:

3a. If the registration process is unsuccessful, the system provides troubleshooting steps or the option to try again.

UC-PPBA05: Store and Encrypt Biometric Data**Main Success Scenario:**

1. The user submits their biometric data to be stored.
2. The system encrypts the biometric data using the FHE scheme.
3. The encrypted data is stored in the database securely.

Extensions:

3a. If encryption fails, the system retries the process or prompts the user for re-submission.

UC-PPBA06: Authenticate Guest User Biometrics**Main Success Scenario:**

1. The system provides a temporary authentication process for guest users.
2. The guest user submits their biometric data for a one-time authentication.
3. The system processes the authentication and grants temporary access without storing the biometrics.

Extensions:

3a. If authentication fails, the guest user is informed and no temporary access is granted.

UC-PPBA07: Authenticate and Store Base User Biometrics**Main Success Scenario:**

1. Base users submit their biometrics for authentication and storage.
2. The system encrypts and verifies the biometrics against the stored data.
3. Upon successful authentication, the biometrics are stored in the user's database record.

Extensions:

3a. If storage or authentication fails, the system notifies the user and offers to repeat the process.

UC-PPBA08: Access Database Log (Administrator Only)**Main Success Scenario:**

1. The administrator requests access to the database logs.
2. The system presents the encrypted logs.

-
3. The administrator reviews the biometric data logs for auditing or monitoring purposes.

Extensions:

3a. If the logs are inaccessible, the system reports the error to the administrator and logs the incident for further investigation.

UC-PPBA09: Print Authentication Result

Main Success Scenario:

1. After biometric data input, the system processes the authentication.
2. The system determines the level of access granted based on user type and authentication result.
3. The system displays a message indicating the user's authentication status and access level.

Extensions:

3a. If the system cannot determine the access level, it prompts the user for additional verification.

TEST PLANS

Test Strategy

For the PPBA system, testing will help ensure the acknowledgement of proper functionality within cryptography and data handling operations. This series of testing cases will validate that each component meets its technical and functional expectations, as defined by the requirements. The test plan will illustrate the testing strategy for the prototype, providing indicative test cases that demonstrate how the system should operate and be measured from initiation to the final validation step.

Test Objectives

Testing will validate the first iteration across several areas:

- **Functionality:** Confirm that all operations, including data capture, encryption, decryption, and file storage, work as intended.
- **Security:** Ensure that the encrypted data remains secure through all operations and cannot be compromised.
- **Performance:** Assess the system's performance, especially focusing on the response time for the operations.
- **Usability:** Evaluate the command-line interface for ease of use and error handling.

Test Strategy

- **Unit Testing:** Isolate each function to confirm its behaviour.
- **Integration Testing:** Test the workflow from data input to output ensuring everything interacts correctly.
- **Security Testing:** Perform vulnerability assessments and validate encryption strength.
- **Usability Testing:** Evaluate the user's experience through CLI interactions.

KEY FOR TEST CASES



Functional Test Cases

Test-ID	Description	Process	Expected Result	Actual Result	Status
FTC-01	Data Capture and Preprocessing	Simulate the capture and preprocessing of biometric data	A preprocessed data array is generated		
FTC-02	Encryption	Encrypt the preprocessed data using the TenSEAL context	encrypted data is produced		
FTC-03	Decryption	Decrypt the encrypted data	Original preprocessed data	Decrypted data matches if operations are	

			is returned to its original state	reversed before decryption	
FTC-04	File I/O	Write and read encrypted data from the database	Data integrity is maintained through the process.	Data written and read back matches	
FTC-05	Encrypted Comparison	Compare two encrypted data entries to validate they are the same without decryption	An encrypted result indicating match or mismatch is obtained		

Security Test Cases

Test-ID	Description	Process	Expected Result	Actual Result	Status
STC-11	Encrypted Data Integrity	Verify the integrity of the encrypted operations	Encrypted data cannot be interpreted without decryption	Encrypted data appears random with no noticeable patterns	
STC-12	FHE Operations	Perform and reverse operations on the data	Decrypted results match the original data.	decrypted data only matches the original data when operations are reversed and not when directly decrypt from encrypted state	
STC-13	Encrypted Password Comparison	Perform a secure comparison of two encrypted passwords	The Comparison yields an encrypted binary result indicating a match (1) or no match (0).		

Performance Test Cases

Test-ID	Description	Process	Expected Result	Actual Result	Status
PTC-21	Encryption Performance	Measure time taken for encryption	Encryption occurs within a acceptable duration	Time taken is acceptable, varies depending on data size	

PTC-22	Decryption Performance	Measure time taken for decryption	Decryption occurs within a acceptable duration	Time taken is acceptable, varies depending on data size	
PTC-23	End-toEnd Authentication Workflow	Execute a complete authentication process including registration and login attempt	The system successfully completes the process with an encrypted comparison, indicating match or no match.		

Usability Test Cases

Test-ID	Description	Process	Expected Result	Actual Result	Status
UTC-31	CLI Feedback	Interact with the CLI to perform system operations	The CLI provides clear instructions	CLI prompts are clear and informative	
UTC-32	Error Handling	Intentionally cause errors in the CLI	The system provides helpful error messages and stops operations from continuing	Error messages are displayed and incorrect operations are not executed	

SYSTEM DESIGN

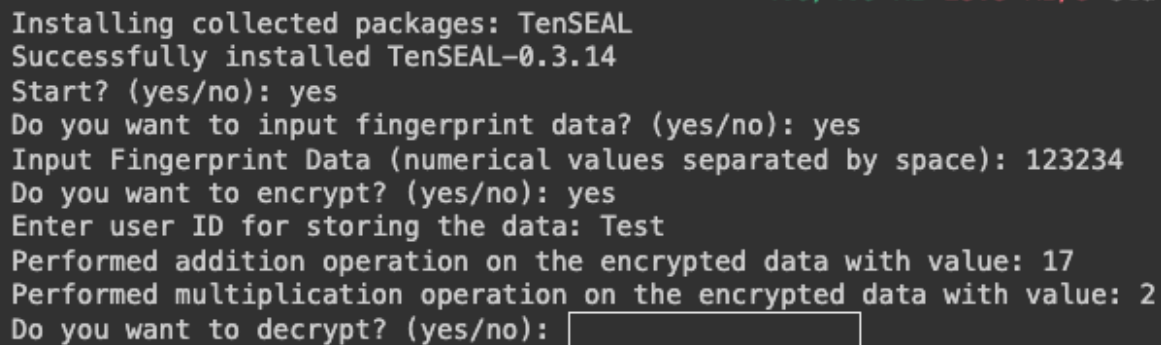
Detailed System Overview

The Privacy-Preserving Biometric Authentication systems architecture is designed to help prioritise core cryptographic functionalities, ensuring that sensitive biometric data is encrypted and processed with tight security. Recognising the importance of privacy, the system's initial development phase has focused on the development of the back-end of the system purposely deferring from developing a front-end UI.

Focus on Back-End Development:

Due to this system focusing on cryptographic operations the back-end of the project is the foundation of the entire operation. The components developed currently include:

- **Cryptography:** This system leverages the advanced cryptographic library TenSEAL to implement the encryption techniques. The choice of the CKKS scheme for FHE ensures that arithmetic operations can be performed on encrypted data without ever exposing the raw biometric data.
- **Data Processing:** Due to lack of equipment in regards to the acquisition of biometric data, I will be simulating real-world data capture methods. This includes a preprocessing step, essential for preparing data for encryption, to ensure consistency and reliability before data enters the operational stage.
- **Secure Storage:** In order to get a secure database, I am using a simulated file system that manages and stores the encrypted data. This choice was made to ensure that the data is never left encrypted or exposed during any operational process.
- **Command-Line Interface (CLI):** As no development for the front-end aspect has been developed, the system currently uses a CLI as its primary mode of interaction. The CLI caters to all functions such as starting the authentication process, inputting data, and triggering encryption/decryption operations. The back-end is developed in Python, as it allows for flexibility and scalability needed for future enhancements, including the front-end integration.

A terminal window showing the installation and usage of TenSEAL. The text is as follows:

```
Installing collected packages: TenSEAL
Successfully installed TenSEAL-0.3.14
Start? (yes/no): yes
Do you want to input fingerprint data? (yes/no): yes
Input Fingerprint Data (numerical values separated by space): 123234
Do you want to encrypt? (yes/no): yes
Enter user ID for storing the data: Test
Performed addition operation on the encrypted data with value: 17
Performed multiplication operation on the encrypted data with value: 2
Do you want to decrypt? (yes/no): 
```

Figure 5 - Shows the current CLI interface

Front-End UI

Given the security-sensitive nature of the biometric authentication system, the initial design iterations have been solely concentrated on the back-end to ensure that the foundations for the system are secure. While the front-end UI is not yet created in the current phase of the system, its eventual integration is planned to ensure it aligns seamlessly with the back-end structure. The planned UI will serve as a medium between the user and the cryptographic operations taking place in the back-end.

- **Security-Centric Interface:** The front-end will be created with security as the main forefront, this is to ensure that the user interactions do not compromise the encrypted state of the data .
- **Intuitive User Experience:** Ease of use is a key feature. The design will be intuitive, minimising user errors and guiding the user through the authentication process with clear instructions.

Back-End Architecture

- **Data Capture and Preprocessing:**

The system starts with the data capture process, this, in a real-world scenario, would involve collecting biometric data from a user through a sensor. The 'Capture_mock_fingerprint_data' function simulates this by generating a list of

random numbers, each representing a point in a biometric profile. **Appendix Figure 1.**

Following the data capture, the data undergoes preprocessing to standardise and prepare it for encryption. The preprocessing step ensures that data is consistent. **Appendix Figure 2.**

- **Encryption with Fully Homomorphic Encryption**

With the user's biometric data captured and preprocessed, the data is now encrypted. The 'create_context' and 'encrypted_biometric_data' functions are central to this process. In order to use encryption the TenSEAL library is applied. **Appendix Figure 3.**

- **Secure Storage**

Once the data is encrypted the biometric data must be securely stored. This is handled through the 'write_data' function, this function writes the encrypted data to the file system, simulating a secure database storage operation. **Appendix Figure 4.**

- **Decryption and Comparison**

The system needs to be able to decrypt the data and perform secure comparisons. The 'decrypt_result' function converts the encrypted data back into plaintext in a secure environment. **Appendix Figure 5.**

The comparison of encrypted data is handled by the 'compare_encrypted_data' function; this function compares the two ciphertexts without revealing their plaintexts. **Appendix Figure 6.**

- **User Interaction and System Flow**

Currently the system's user interaction is done through the command-line interface as no Front-end UI is designed yet. **Appendix Figure 7.**

IMPLEMENTATION REPORT

Implementation

This implementation report will give an insight into the details of the prototype of a Privacy-Preserving Biometric Authentication (PPBA) system. This system encompasses the strengths of the Fully Homomorphic Encryption scheme to ensure that the sensitive biometric data remains encrypted throughout the processing and storing phases, offering a high level of privacy and security. The first draft demonstrates back-end functionality, including data capture, preprocessing, encryption/decryption, and secure computation.

The first iteration primarily consists of back-end components that are written in Python, as I am utilising the TenSEAL library for my implementation of FHE. No front-end aspects have been developed yet, instead, it currently operates through a command-line interface, which guides the users through the various steps of the authentication process.

1. Data Capture and Preprocessing:

- The prototype simulates the capture of biometric (fingerprint data) by generating random numerical values, which represent the unique features of a fingerprint.
- Preprocessing is depicted as a placeholder function, highlighting where actual preprocessing steps like normalisation and feature extraction would occur.

2. Encryption / Decryption with FHE:

- The core of this iteration is the encryption and decryption of the biometric data using a FHE scheme. A TenSEAL context is created to handle encryption parameters and keys.
- Biometric data is encrypted into a CKKS vector (Ciphertext) and then serialised for the storage and transmission.
- Decryption operations are provided to convert the encrypted data back into its original form.

3. Secure Computation:

-
- In order to demonstrate the capabilities of FHE, this iteration performs addition and multiplication operations on the encrypted data.
 - The operations are also reversible, allowing for the recovery of the original data after decryption, allowing for the confirmation of the successful application.

4. Storage:

- The Encrypted data is stored in a file system, emulating a database. With each user's data being stored in a separate file, identified by a user-provided ID.

5. CLI Operations:

- The first iteration operates via CLI, which provides a straightforward way to interact with the system. Users can start the process, input data, and choose to encrypt, decrypt and verify integrity of the process.

Security Measures

Security considerations are paramount when working with biometric information. To ensure that biometric data is never exposed in plaintext during computations. Random values used in secure computations introduce an additional layer of security, creating an infeasible process to hinder reverse-engineering applications.

Challenges and Solutions

One of the key challenges was trying to implement the FHE operations to ensure that they are both secure and efficient. The solution involved a careful selection of encryption parameters and the use of lazy evaluation for the encrypted vectors, supported by the TenSEAL library.

Future Enhancements

I am hoping that the next phase and iterations will involve the development of a user-friendly front-end system, this is likely to be done through the use of a web application, which will coincide with the back-end. In an ideal world I would also integrate

the use of real biometric sensors and more advanced preprocessing techniques, however, due to skill and time restraints this enhancement is unlikely.

Conclusion

The first prototype of the system creates a demonstration that effectively shows the potential of FHE when securing sensitive biometric data. While currently limited to CLI and back-end operations, the system lays a solid foundation for a comprehensive authentication platform.

References

1. Jain, A., Ross, A. & Prabhakar, S. (2004). An Introduction to Biometric Recognition. *IEEE Transactions on Circuits and Systems for Video Technology*, 14(1), pp. 4-20.
2. IEEE Xplore. Biometric Authentication Security: An Overview. [online] Available at: <https://ieeexplore.ieee.org> (Accessed: 10 November 2023).
3. Rathgeb, C. & Uhl, A. (2011). A survey on biometric cryptosystems and cancelable biometrics. *EURASIP Journal on Information Security*, 2011(1), pp. 1-25.
4. ResearchGate. Biometric Authentication: A Review. [online] Available at: <https://www.researchgate.net> (Accessed: 11 November 2023).
5. ScienceDirect. The design and evaluation of adaptive biometric authentication systems. [online] Available at: <https://www.sciencedirect.com> (Accessed: 11 November 2023).
6. PMC. Biometrics for Internet-of-Things Security: A Review. [online] Available at: <https://www.ncbi.nlm.nih.gov> (Accessed: 11 November 2023).
7. Bringer, J., Chabanne, H. & Pointcheval, D. (2007). Cryptanalysis of a Privacy-Preserving Biometric Authentication Scheme. *Information Processing Letters*, 107(5), pp. 165-169.
8. Ratha, N., Connell, J. & Bolle, R. (2001). Enhancing security and privacy in biometrics-based authentication systems. *IBM Systems Journal*, 40(3), pp. 614-634.
9. Sutcu, Y., Sencar, H. & Memon, N. (2007). A Secure Biometric Authentication Scheme Based on Robust Hashing. *7th ACM Workshop on Multimedia and Security*, pp. 111-120.
10. Osadchy, M., Hernandez-Castro, J., Gibson, S., Dunkelman, O. & Pérez-Cabo, D. (2017). Fast Secure Computation for Small Population Over the Clouds. *IEEE Transactions on Dependable and Secure Computing*, 15(4), pp. 562-576.
11. Paillier, P. (1999). Public-Key Cryptosystems Based on Composite Degree Residuosity Classes. *EUROCRYPT*, pp. 223-238.
12. IEEE Spectrum. The Future of Fully Homomorphic Encryption. [online] Available at: <https://spectrum.ieee.org> (Accessed: 12 November 2023).
13. Harvard University. Fully Homomorphic Encryption with Applications to Privacy-Preserving. [online] Available at: <https://dash.harvard.edu> (Accessed: 12 November 2023).
14. Erdogmus, N. & Marechal, A. (2017). A Comparative Analysis of Homomorphic Encryption Schemes Suitable for Secure Biometric Authentication. *IEEE International Carnahan Conference on Security Technology*, pp. 1-8.
15. NCBI. A Review of Homomorphic Encryption for Privacy-Preserving Biometrics. [online] Available at: <https://www.ncbi.nlm.nih.gov> (Accessed: 12 November 2023).
16. IEEE Xplore. Multimodal biometric authentication using Fully Homomorphic Encryption. [online] Available at: <https://ieeexplore.ieee.org> (Accessed: 12 November 2023).
17. Emerald.com. Privacy-preserving biometrics authentication systems using fully homomorphic encryption. [online] Available at: <https://www.emerald.com> (Accessed: 12 November 2023).
18. Semantic Scholar. Secure Fingerprint Authentication with Homomorphic Encryption. [online] Available at: <https://www.semanticscholar.org> (Accessed: 13 November 2023).
19. IEEE Xplore. A Survey on Homomorphic Encryption for Biometrics Template Security. [online] Available at: <https://ieeexplore.ieee.org> (Accessed: 13 November 2023).
20. (No date) 4 complexity and uniqueness of human iris. fine textures ... - researchgate. Available at:

https://www.researchgate.net/figure/Complexity-and-uniqueness-of-human-iris-Fine-textures-on-the-iris-form-unique-biometric_fig3_315477410 (Accessed: 13 November 2023).

21. (No date a) Face de-spoofing: Anti-spoofing via noise modeling - researchgate. Available at: https://www.researchgate.net/profile/Xiaoming-Liu-37/publication/328157652_Face_De-spoofing_Anti-spoofing_via_Noise_Modeling_15th_European_Conference_Munich_Germany_September_8-14_2018_Proceedings_Part_XIII/links/5c16ef48299bf139c75e25ef/Face-De-spoofing-Anti-spoofing-via-Noise-Modeling-15th-European-Conference-Munich-Germany-September-8-14-2018-Proceedings-Part-XIII.pdf (Accessed: 13 November 2023).
22. (No date a) Biometric authentication. types of biometric identifiers - theseus. Available at: https://www.theseus.fi/bitstream/handle/10024/44684/Babich_Aleksandra.pdf (Accessed: 14 November 2023).
23. Pradel, G. and Mitchell, C. (2021). Privacy-Preserving Biometric Matching Using Homomorphic Encryption. [image] Available at: <https://ar5iv.org/abs/2111.12372> (Accessed: 14 November 2023).
24. Ajtai, M. (1996). Generating Hard Instances of Lattice Problems. (Accessed: 14 November 2023)
25. Regev, O. (2009). On Lattices, Learning with Errors, Random Linear Codes, and Cryptography. (Accessed: 14 November 2023)

Appendix

Figure 1

```
def capture_mock_fingerprint_data():  
    # Simulating the capture of biometric data as a list of random floats  
    return [random.uniform(0, 1) for _ in range(10)]
```

Figure 2

```
def preprocess_fingerprint_data(fingerprint_data):  
    # Placeholder for preprocessing biometric data  
    return fingerprint_data
```

Figure 3

```
def create_context():  
    # Creating a TenSEAL context with specified parameters for the CKKS  
    context = ts.context(ts.SCHEME_TYPE.CKKS, poly_modulus_degree=16384,  
        context.generate_galois_keys()  
        context.global_scale = 2**40  
    return context  
  
def encrypt_biometric_data(data, context):  
    # Encrypting the biometric data and serializing it for secure storag  
    encrypted_data = ts.ckks_vector(context, data)  
    return encrypted_data.serialize()
```

Figure 4

```
def write_data(file_name, data):  
    # Writing encrypted data to a file (simulating secure database storage)  
    with open(file_name, 'wb') as file:  
        file.write(data)
```

Figure 5

```
def decrypt_result(encrypted_result, context):  
    # Decrypting the data using the TenSEAL context  
    auth_result = ts.lazy_ckks_vector_from(encrypted_result)  
    auth_result.link_context(context)  
    decrypted_result = auth_result.decrypt()  
    return decrypted_result
```

Figure 6

```
def compare_encrypted_data(encrypted_data1, encrypted_data2, context):  
    # Logic to compare two encrypted pieces of data  
    encrypted_vector1 = ts.lazy_ckks_vector_from(encrypted_data1)  
    encrypted_vector2 = ts.lazy_ckks_vector_from(encrypted_data2)  
    encrypted_vector1.link_context(context)  
    encrypted_vector2.link_context(context)  
    comparison_result = encrypted_vector1 - encrypted_vector2  
    comparison_result.square_()  
    decrypted_result = comparison_result.decrypt()  
    threshold = 1e-5  
    is_match = all(value < threshold for value in decrypted_result)  
    return 1 if is_match else 0
```

Figure 7

```
def privacy_preserving_biometric_authentication():  
    # CLI to guide the user through the authentication process  
    start = input("Start? (yes/no): ")  
    if start.lower() != "yes":  
        print("Exited.")  
        return
```