Data Anonymization in Health Monitoring Systems

Introduction

Data anonymization is a crucial technique in healthcare data management that ensures sensitive patient information remains private and secure, even in the case of data breaches or unauthorized access. In the context of an elderly care wearable device, health metrics such as body temperature, heart rate, and oxygen saturation (SpO2) need to be safeguarded. The provided code demonstrates an effective approach to anonymizing health data by fetching real-time data from external APIs, applying AES encryption to sensitive information, and allowing controlled decryption when required. This ensures that health metrics are stored and transmitted securely without compromising the privacy of the individual.

Data Fetching and Transformation

The first step in the process involves collecting health data from Thing Speak. These metrics are fetched using requests and transformed into a structured format using pandas. The fetched data includes timestamps and corresponding health readings, which are important for tracking and analyzing the user's health over time.

Data Transformation Steps:

1. API Requests: The code connects to two channels (Channel 1 and Channel 2) to retrieve health data. Each channel is configured with an API key stored in an .env file for security, ensuring that the requests are authenticated.
2. Data Structuring: Once the data is fetched, it is transformed into a pandas DataFrame, where the relevant fields (e.g., timestamp, Body_temp, heartRate, and SP02) are extracted. Column names are renamed for clarity, and the timestamp is converted into a proper datetime format.
3. Data Cleaning: The health metrics (e.g., body temperature, heart rate) are converted into numeric values with rounding, ensuring the data is in the correct format for further processing.

At this stage, the sensitive data is now well-structured and ready for the anonymization process through encryption.

Anonymization via AES Encryption

To ensure privacy, the health metrics are encrypted using AES (Advanced Encryption Standard), a symmetric encryption algorithm that is widely recognized for its strength and efficiency. AES encryption ensures that even if data is intercepted, it cannot be deciphered without the corresponding decryption key.

Encryption Process:

1. Key Generation: A new 256-bit AES key is generated using os.urandom(32). This key is essential for both the encryption and decryption process.
2. Encryption:

- Initialization Vector (IV): For each encryption operation, a unique 128-bit IV is generated. The IV ensures that even if the same data is encrypted multiple times, the output will be different, thus preventing pattern recognition.
- AES Encryption: The health data (body temperature, heart rate, and SpO2) is then encrypted using AES in CFB (Cipher Feedback) mode, which allows encryption of data in smaller blocks and ensures secure transmission.
- Base64 Encoding: After encryption, the IV and ciphertext are concatenated and encoded using Base64 to ensure that the encrypted message can be safely transmitted or stored in a database.

For example, the body temperature data from Channel 1 is converted from its original numeric form into an encrypted string, making it impossible to decipher without the AES key.

The result is an anonymized dataset where the sensitive health metrics have been securely encrypted, rendering them inaccessible to unauthorized parties.

Controlled Decryption

In scenarios where the anonymized data needs to be analyzed or presented to authorized users (e.g., healthcare providers), decryption can be performed. The AES key used during encryption is essential for this process, and without it, the data remains secure and anonymous.

Decryption Process:

1. Base64 Decoding: The encrypted data is first decoded from Base64, separating the IV and the ciphertext.
2. AES Decryption: Using the same AES key and IV, the encrypted health metrics are decrypted back into their original numeric values using the CFB mode.
3. Data Restoration: The decrypted values (e.g., body temperature, heart rate) are then stored in a new column alongside the original encrypted values, allowing authorized users to see both the anonymized and original data.

By implementing decryption only when necessary, the system ensures that sensitive data remains protected while still providing flexibility for authorized access when required.

Advantages of This Anonymization Approach

1. Data Privacy: The use of AES encryption ensures that sensitive health metrics cannot be accessed by unauthorized individuals, even if the database or communication channel is compromised.
2. Data Integrity: The encryption process does not alter the original structure or meaning of the data, ensuring that the encrypted data remains accurate and can be decrypted back to its original form without loss of information.
3. Compliance with Privacy Standards: The anonymization techniques used, such as encryption and the secure handling of API keys, help meet data protection standards and regulations such as GDPR (General Data Protection Regulation) and HIPAA (Health Insurance Portability and Accountability Act).

4. Flexible Decryption: By providing a controlled decryption process, the system balances the need for data privacy with the ability to access sensitive information when necessary.

Implementing anonymization in the warehouse solution

To implement anonymization in the warehouse solution, we began by allowing users to specify which columns contain sensitive information. This step involved prompting the user to input the names of the sensitive columns they want to encrypt, ensuring flexibility in adapting the anonymization process to different datasets. Once the sensitive columns were identified, the data was encrypted using a secure method. This ensures that even if the data is accessed, the sensitive information remains secure and can only be decrypted by authorized users with the correct key. This dynamic, column-based approach ensures the data stored in the warehouse is compliant with privacy regulations while maintaining security.