Blockchain for Increasing Reusability of Patient Medical Data

1 Project Overview

The increasing availability of real-world medical data presents significant opportunities for advancing medical research. However, despite its potential, several challenges persist in securely reusing this data in a way that balances accessibility with patient privacy. Papers such as "What prevents us from reusing medical real-world data in research" by Julia Gehrmann et al. (2023) and "FAIR Aspects of a Health Information Protection and Management System" by Jaime Delgado and Silvia Llorente (2022) highlight the barriers researchers face, including issues around data fragmentation, consent management, and compliance with privacy regulations like HIPAA and GDPR.

This project, Blockchain for Increasing Reusability of Patient Medical Data, leverages blockchain technology to address these challenges. By utilizing a decentralized ledger for recording permissions and secure, off-chain data storage, this system provides a verifiable, citable, and privacy-preserving method for researchers to access patient medical data. The blockchain ensures that all interactions with patient data are logged and immutable, while cryptographic mechanisms, including RSA and AES encryption, safeguard patient privacy and ensure that only authorized parties can access sensitive information.

1.1 Introduction

This document describes a blockchain-based medical data sharing system that ensures secure, permissioned access to patient records stored off-chain by Verified Organizations (VO). The system uses RSA and AES encryption and JWTs to facilitate secure, traceable, and privacy-preserving data transactions. Importantly, no actual medical data is stored on-chain; instead, encrypted pointers and approval signatures are passed between the requester, patient, and VO to ensure privacy. We explore two main user stories:

1. Accessing data referenced in the blockchain after reading a citation.
2. Adding new data to the blockchain.

2 User Story 1: Accessing Data Referenced in a Citation

Context: A researcher (Requester) reads a citation referencing medical data stored off-chain by a VO. The data is referenced on-chain via a transaction ID that acts as a pointer.

2.1 Steps

1. **Data Discovery:** The researcher locates the transaction ID in the citation, which links to the on-chain pointer referencing the data. This pointer is stored in Transaction 0: Data Pointer. The data could be found as a citation in a medical resarch paper.

2. **Access Request:** The researcher submits an access request on the blockchain, including the transaction ID of the cited data (from Transaction 0: Data Pointer) and their public key. This creates Transaction 1: Access Request.

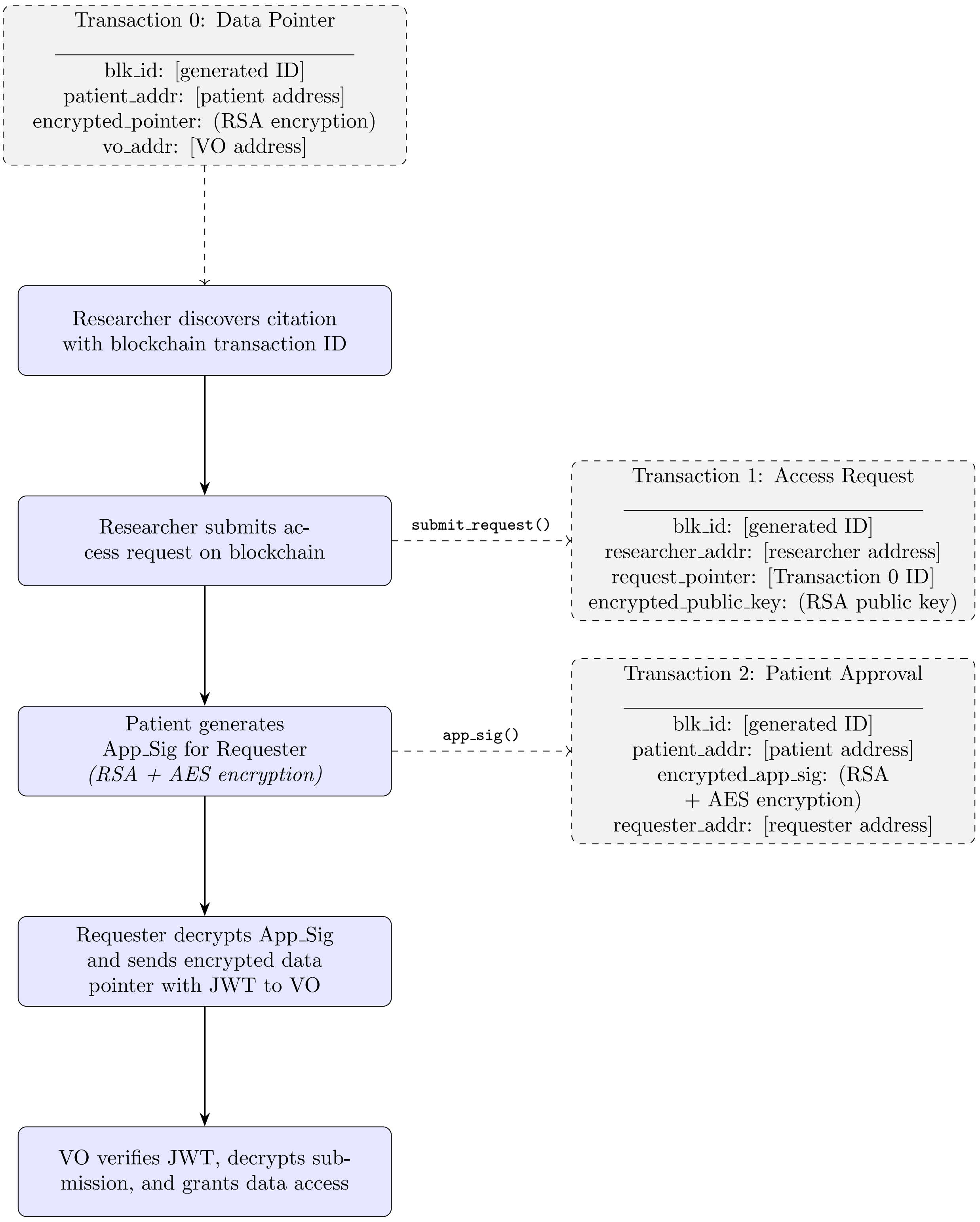
3. **Patient Approval:** The patient generates an approval signature (App_Sig) using their private key and the requester's public key taken from Transaction 1: Access Request (field: encrypted_public_key). The patient encrypts the data pointer (from Transaction 0: Data Pointer) using AES, then encrypts the AES key with the requester's public key using RSA.

4. **On-Chain Approval:** The patient's encrypted approval signature is stored on-chain in Transaction 2: Patient Approval, which includes the patient's address, the encrypted App_Sig, and the requester's address.

5. **Researcher Processing and VO Submission:** The requester decrypts the approval signature from Transaction 2: Patient Approval (field: encrypted_app_sig) using their private RSA key. This decryption reveals the AES key and IV. The requester then uses this AES key and IV to decrypt the encrypted data pointer. Importantly, this decrypted data pointer is still encrypted with the VO's public key. The requester sends this encrypted data pointer to the VO along with a JWT containing claims about the request, including their address (from Transaction 1: Access Request), the retrieved record ID, and an expiration time.

6. **VO Verification and Data Access:** The VO first verifies the JWT using the requester's public key (from Transaction 1: Access Request). If the JWT is valid, the VO then decrypts the submission (which originates from Transaction 2: Patient Approval) with their private key to retrieve the original record ID. If both the JWT and the submission are verified, the VO grants access to the off-chain data corresponding to the retrieved record ID.

2.2 Diagram


Transaction 0: Data Pointer
blk_id: [generated ID]
patient_addr: [patient address]
encrypted_pointer: (RSA encryption)
vo_addr: [VO address]
Researcher discovers citation with blockchain transaction ID
Researcher submits access request on blockchain
submit_request()
Transaction 1: Access Request
blk_id: [generated ID]
researcher_addr: [researcher address]
request_pointer: [Transaction 0 ID]
encrypted_public_key: (RSA public key)
Patient generates App_Sig for Requester
(RSA + AES encryption)
app_sig()
Transaction 2: Patient Approval
blk_id: [generated ID]
patient_addr: [patient address]
encrypted_app_sig: (RSA + AES encryption)
requester_addr: [requester address]
Requester decrypts App_Sig and sends encrypted data pointer with JWT to VO
VO verifies JWT, decrypts submission, and grants data access

2.3 Encryption Overview

- The **data pointer** (medical record ID) is encrypted using RSA with the VO's public key in Transaction 0.
- The **approval signature** (App_Sig) uses a combination of RSA and AES encryption: the data pointer is encrypted with AES, and the AES key is encrypted with the requester's RSA public key.
- The **VO submission** is the data pointer re-encrypted with the VO's public key.
- **JWTs** are used for API authentication, allowing secure access to the off-chain data. The JWT is signed with the requester's private key and verified with their public key from Transaction 1.

Implications: This flow ensures that all data access is traceable on the blockchain, but the data itself is shared off-chain through encrypted channels. This preserves patient privacy while allowing for verifiable citations.

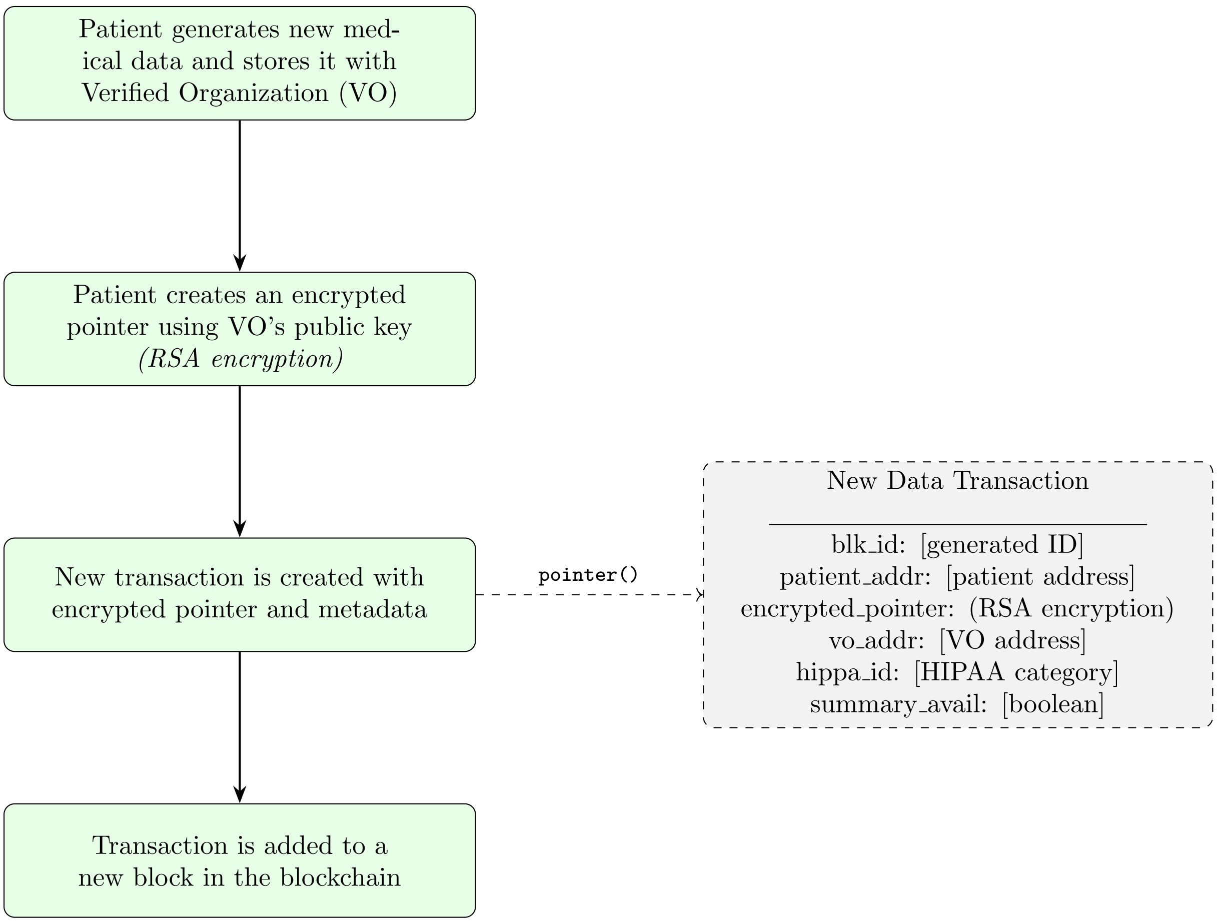
3 User Story 2: Adding New Data to the Blockchain

Context: A patient generates new medical data that they wish to make available via a VO, who holds the data off-chain. The data pointer is stored on-chain for future access.

3.1 Steps

1. **Data Creation:** The patient generates medical data (e.g., test results) that the VO stores off-chain.
2. **Pointer Creation:** The patient creates an encrypted pointer using the VO's public key, ensuring that only the VO can decrypt it. This uses the `pointer()` method from the Patient class.
3. **Transaction Creation:** A new transaction is created with the encrypted pointer and other metadata, including the patient's address, VO's address, and HIPAA category.
4. **Block Addition:** The transaction is added to a new block in the blockchain, which is then mined and added to the chain.

3.2 Diagram


Patient generates new medical data and stores it with Verified Organization (VO)
Patient creates an encrypted pointer using VO's public key
(RSA encryption)
New transaction is created with encrypted pointer and metadata
pointer()
New Data Transaction
blk_id: [generated ID]
patient_addr: [patient address]
encrypted_pointer: (RSA encryption)
vo_addr: [VO address]
hippa_id: [HIPAA category]
summary_avail: [boolean]
Transaction is added to a new block in the blockchain

3.3 Encryption Overview

- The **data pointer** (medical record ID) is encrypted with RSA using the VO's public key.
- All relevant transaction details are stored on-chain, providing an auditable trail.

Implications: This process ensures that the patient retains control over their medical data, allowing them to decide who can access it and when, while maintaining privacy through off-chain storage of actual medical data.

4 Conclusion

This system provides privacy and traceability, ensuring that patients control access to their medical data while enabling verifiable citations in research. By using RSA encryption, AES encryption for approval signatures, and JWTs for API authentication, the system ensures secure off-chain data sharing while maintaining an immutable, on-chain audit trail. The blockchain structure allows for efficient searching and verification of transactions, enhancing the overall security and usability of the medical data sharing system.