

Privacy Preservation and Access Control for Sharing Electronic Health Records Using Blockchain Technology

Insaf Boumezbeur , Karim Zarour 

LIRE Laboratory, Software and Information Systems Technologies Department, Faculty of Information and Communication Technology, Constantine 2 University – Abdelhamid Mehri, Nouvelle ville Ali Mendjli BP67A, Constantine, Algeria

Corresponding author: Insaf Boumezbeur (insaf.boumezbeur@univ-constantine2.dz)

Abstract

Sharing of Electronic Health Records (EHRs) is of significant importance in health care. Lately, a cloud-based electronic health record sharing scheme has been used extensively to share patient records among various healthcare organizations. However, cloud centralization may compromise patients' privacy and security. Due to the special features of blockchain, it is important to see this technology as a promising solution to resolve these issues. This article proposes a privacy-preserving, secure EHR sharing and access control framework based on blockchain technology. The proposal aims to implement EHR blockchain technology and ensure that electronic records are stored safely by specifying user access permissions. We emulate the cryptographic primitives and use smart contracts to describe the relationships between the EHR owner and EHR user through the proposed system on the Ethereum blockchain. We assess the proposal results based on encryption and decryption time and the costs of the smart contract. The encryption and decryption times are proportional to the size of the EHR, which varies from 128 KB to 128 MB. When it comes to encryption, the smallest EHR takes 0.0012 s to encrypt, while the largest EHR, which is 128 MB, takes 1.4149 s. On the other hand, a 128 KB EHR takes 0.0013 s to decrypt, whereas a 128 MB EHR requires 1.6284 s. As a result, performance evaluation and security analysis confirm that the proposal is secure for practical application.

Keywords

Blockchain; Encryption; Electronic health record; Privacy.

Citation: Boumezbeur, I., & Zarour, K. (2022). Privacy-Preserving and Access Control for Sharing Electronic Health Record using Blockchain Technology. *Acta Informatica Pragensia*, 11(1), 105–122. <https://doi.org/10.18267/j.aip.176>

Academic Editor: Michal Dolezel, Prague University of Economics and Business, Czech Republic

Copyright: © 2022 by the author(s). Licensee Prague University of Economics and Business, Czech Republic.

This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution License (CC BY 4.0).

1 Introduction

An electronic health record (EHR) is a compilation of an individual's health-related information, including personal information, medical images, medical conditions and medications. EHR sharing will help both patients and medical institutions in a few ways (HIMSS, 2006). Firstly, data sharing can facilitate medical research. Secondly, it would be better for multiple healthcare institutions to cooperate, such as physicians obtaining patients' medical histories and medical service coverage in a foreign country. Thirdly, regulations and standards can be established and reinforced to promote healthy EHR sharing, offer more trust among various medical institutions, provide more patient care and strengthen medical science.

However, the trend of cloud storage of EHRs and all these great benefits also poses security challenges that prevent the implementation of cloud e-health applications (Abukhousa et al., 2012). EHRs have been more widely available, although use has been limited due to security concerns (Sauermaun et al., 2013). Some technological obstacles remain that impede EHR sharing and render it complicated in several ways. Thus, one of the most significant challenges in healthcare systems is secure sharing of medical data. The sharing mode raises a lot of privacy and security issues that could thwart its widespread adoption (Boumezbear and Zarour, 2018). Such data are sensitive, making patients and medical organizations reluctant to share them because they need protection against unauthorized access. Security issues include the secure sharing of EHRs among patients and other healthcare services in cloud environments. Unauthorized individuals can obtain malicious access to EHRs without patient authorization, undermining data privacy, security and integrity of cloud-based e-health systems. In addition, patients may have difficulty monitoring and maintaining their cloud-based health records shared by healthcare providers. Therefore, it is important to suggest appropriate access control solutions for sharing EHRs among cloud environments.

Recently, the use of blockchain services to enhance medical and e-health services has become a growing development. Satoshi Nakamoto initially brought blockchains to the globe in the form of the popular cryptocurrency Bitcoin (Shuaib et al., 2014). It is a decentralized architecture. It consists of an open and distributed ledger that records all transactions between two parties in an efficient, verifiable and permanent way (Rajput et al., 2019). It contains individual entries in the form of interconnected blocks organized in a single list known as a chain. After authentication by a network of interconnected validating nodes, each transaction is appended to the blockchain (Siddiqui et al., 2020a; Siddiqui et al., 2020b). Unlike traditional databases, the blockchain enables members in a distributed network to exchange electronic currency without requiring a centralized, trusted third party (Agbo et al., 2019; Hardin and Kotz, 2019). It relies on validators (typically miners) to operate as third parties and decentralize transaction validation (Hölbl et al., 2018).

Due to blockchain's immutability and use of cryptographic functions for secure communication, blockchain is well suited for accurate EHR information sharing (Gordon and Catalini, 2018; Catalini and Gans, 2020). This technology can change the healthcare system in different fields, such as secure exchange of EHRs and data access control among different medical institutions to improve privacy and data protection (Mayar et al., 2020). It offers a potential future data sharing approach that could enable collaborative clinical decision making in telemedicine and precision medicine (Cheng et al., 2018). Blockchain implementation could also create a new paradigm for exchanging health information (HIE) by making EHRs more effective and secure. It will also provide promising solutions to promote treatment delivery and thus revolutionize the healthcare system. Therefore, various blockchain companies such as HealthNautica (HealthNautica, 2022), Factom (FACTOM, 2022), Capital One (Capitalone, 2022) and Gem (Gem, 2022) are collaborating to preserve medical data using blockchain technologies. Besides, many implementations of health data management systems, such as those reported by Qin et al. (2021), Zhang and Lin (2018), Pournaghi et al. (2020), Li et al. (2018) and Alam et al. (2021), are emerging. Each application provides various solutions for various conditions.

This article proposes a privacy-preserving EHR sharing scheme focused on cloud storage and blockchain technology. Indeed, cryptography is important in securing and maintaining a user's personal information. In the proposal, the original EHRs are uploaded in an encrypted form to a cloud, and only signature and encryption keys are reserved in the blockchain. Secure data sharing could be achieved via smart contracts that manage the access control of users. Patients should have full control of their own EHRs when adopting the new system, and users or care providers may make practical use of the data without disclosing patient privacy. The main contributions of this paper are summarized as follows:

- A blockchain-based cryptographic and access control scheme for sharing EHRs is proposed using Ethereum smart contracts.
- The proposed scheme uses symmetric and asymmetric algorithms to encrypt EHR and secret keys for ensuring confidentiality and privacy.
- The smart contract used in the proposed scheme is intended to manage users' access control. It ensures that the data owner has full control over who has access to their health record.
- We analyse the proposal performance with cloud computing usability tests provided by the Google Storage Platform (GSP) and Ethereum blockchain using Solidity to implement the smart contract. The results show that the suggested framework is feasible.

The remainder of this paper is organized as follows. Section 2 brings an overview of related works. Section 3 describes the system architecture, system workflow and the smart contract. The performance and discussion of the proposal are presented in Section 4. Finally, Section 5 concludes the paper.

2 Literature Review

Li et al. (2018) proposed a new data preservation system (DPS) based on blockchain as a secure storage solution, ensuring that the stored data are primitive and verifiable while maintaining user privacy. The DPS used data storage mechanisms and cryptographic algorithms to achieve security.

An access control model using blockchain for PHRs was proposed by Thwin and Vasupongayya (2019). The authors used proxy re-encryption as access control and other cryptographic techniques to preserve confidentiality. They stored the encrypted records in a cloud while the metadata was stored in the blockchain. In this model, the data sharing process is dependent on an intermediary, the proxy server responsible for re-encryption. Thus, the proxy server is in charge of the encryption keys and other information required for the authentication process.

Wang et al. (2018) proposed a secure EHR system built on blockchain technology and a cryptosystem mechanism to enable fine-grained access control and guarantee the authentication and confidentiality of cloud-stored EHR medical data. They proposed a new cryptographic primitive called hybrid attribute/identity-based encryption and signature (C-AB/IB-ES). The ABE and IBE are used to encrypt the medical data, while the IBS is used to implement digital signatures.

HBasechainDB (Sahoo and Baruah, 2018) is a scalable blockchain framework leveraging the Hadoop database. It used the blockchain pipelining and federated consensus to create the blocks. The blockchain on top of this architecture contains all the necessary dependencies, but the blocks are maintained on the Hadoop database to increase the blockchain technology scalability. This study was beneficial in understanding that blockchain can be used with other scalable platforms to improve or overcome the scalability of this platform. It is also able to explore the data present on the blockchain.

Xia et al. (2017) proposed a blockchain-based data sharing (BBDS) scheme that offers access control policies associated with sensitive health data. The BBDS focused on identity-based authentication, which enhances the efficiency of the healthcare system. Besides, it used cryptographic techniques in a blockchain network to achieve security.

3 System Model

This section presents the architecture, the workflow and the implementation of the smart contract. Table 1 summarizes the basic notations used in this paper.

Table 1. Notations.

Notation	Description
EHR	Electronic health record
EO	EHR owner
EU	EHR user
CS	Cloud storage
BC	Blockchain
SK	Secret key
PuK	Public key
PrK	Private key
MD	Message digest
SIG	Signature

3.1 Electronic Health Record (EHR)

The Health Information Management Systems Society defines the EHR as a longitudinal electronic record of patient health information generated by one or more encounters in any care delivery setting (HIMSS, 2020). The EHR can create a complete record of clinical contact and assist with other tasks such as evidence-based decision support, quality monitoring and reporting. Any healthcare organization or doctor that a patient visit records the patient's information, including current health status and vital signs. It also contains previous treatments, current medications and other important personal information. This patient health record is very important for a doctor, and it is equally important that it be available online and immediately. It must also be updated and accessible when needed.

In the proposed scheme, the EHR data are divided into different parts illustrated in Figure 1, concerning the correlation of the information contained in each part as follows:

- Personal information: Includes the patient's private information (**EHR₁**).
- Contact information: Includes the patient's emergency contact information (**EHR₂**).
- Patient demographics: Contains information about the population needed for research and marketing purposes (**EHR₃**).
- Administrative and billing data: Includes administrative information (**EHR₄**).
- Laboratory report: Includes laboratory information and test results (**EHR₅**).
- Radiology report: Includes radiology information, images and results (**EHR₆**).
- Medical information: Includes medical and medication information (**EHR₇**).
- Insurance report: Contains information about the insurance company (**EHR₈**).



Figure 1. Electronic health record (EHR) model.

3.2 System architecture

The overall architecture of the proposed privacy-preserving and access control scheme for sharing EHRs using blockchain technology (BACP-EHR) is presented in Figure 2. The EHR data will be encrypted using the EHR owner's secret key and stored in cloud storage shared via a secure socket layer (SSL) to ensure confidentiality. Thus, the secret key is encrypted using a public key. The encryption key, as well as all necessary information, would then be stored on blockchain for an authentication process. The EHR would be available to the EHR owner or other users, including the healthcare providers. The proposed framework contains three layers, namely a data collecting layer, a data storing layer and a data sharing layer, and four entities: the EHR owner (EO), the EHR user (EU), cloud storage (CS) and blockchain (BC). The role of each entity is defined as follows:

Data collecting layer: refers to the patient who visits doctors in hospitals or medical institutions for healthcare treatment. The patient obtains his/her electronic health records, including individual private health data generated after their interactions. The main entity in this layer is the *EHR owner (EO)*: it is an entity (e.g., individual or organization) that owns the EHR data to be shared. In our scheme, the EO is primarily responsible for complete control of his/her EHR. He/she must establish an access control policy on his/her EHR and grant or refuse access permissions for his/her EHR data to others at will.

Data storing layer: contains two entities:

- *Cloud storage (CS)*: The cloud stores encrypted EHRs uploaded by the owner.
- *Blockchain (BC)*: The blockchain is an entity that stores the smart contracts, the EHR signature and encrypted keys.

Data sharing layer: In this layer, the EHR user (EU) is an entity that can be an individual (e.g., patient, doctor) or an organization (e.g., hospital, health insurance company, medical research institute) that accesses the patient's EHRs for beneficial purposes.

The overall architecture of BAcP-EHR is presented as follows:

- **Step 1:** Interactions between the patient and his or her doctor and specialist produce primary data. This information contains patient records, current issue information and other physiological information.
- **Step 2:** The EHR owner uploads the ciphertext to the cloud storage and sends the encrypted keys and other information to the blockchain.
- **Step 3:** The EHR user sends a request to the blockchain to access the EHR, retrieve the EHR and decrypt it.

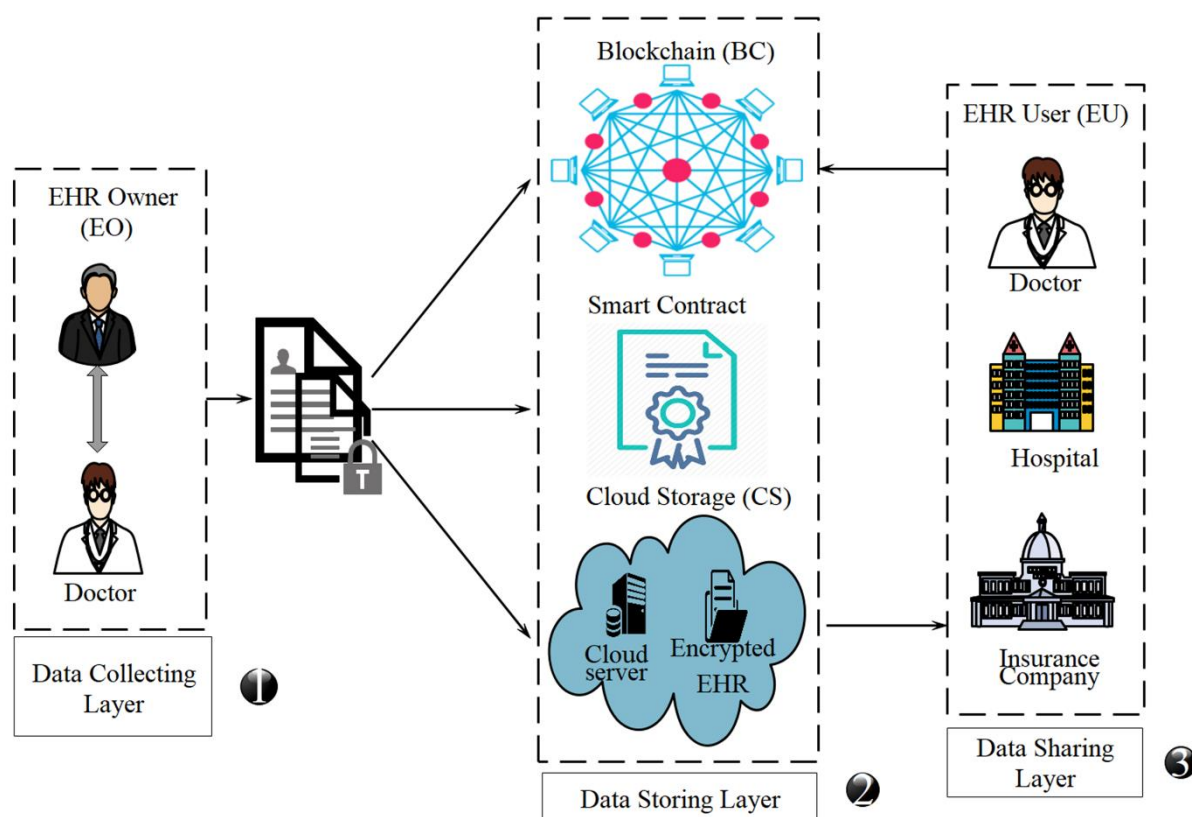


Figure 2. System architecture.

3.3 Threat model and security goals

Cloud servers are semi-reliable in our suggested system. The electronic health record is subject to an attempt to decode the encrypted text. A malicious adversary may capture, alter or forge health records. To deduce the plain text of the EHR, cloud and data requestors will consent. In the threat model, the security targets are as follows:

- **Data privacy:** The owner's original EHR cannot be disclosed to illegal individuals.
- **Data authenticity:** Those accessing the data can check the authenticity of the patient's EHR.
- **Data integrity:** Patient EHRs can be stored secured against tampering.
- **Data confidentiality:** Patient EHRs are securely stored and kept hidden from unauthorized persons.
- **Flexible access control:** Patients can determine how to access their EHRs, and only authorized persons can access patient EHR.
- **Authentication:** Before accessing EHR, users must be authenticated.

3.4 Scheme workflow

In this section, we explain in detail the workflow of the BAcP-EHR scheme. The proposal enables EHR owners to control their own EHRs. Based on cryptography techniques and blockchain technology, it performs privacy-preserving EHR sharing through the following steps. Figure 3 illustrates the proposed scheme in detail.

System setup: To implement the BAcP-EHR scheme, users should register unique accounts and create their keys. First, a symmetric key (SK) 128 bits in length is generated for each EO. The SK is the output of the hash function (SHA-1) employed by the SUN provider's pseudo-random number generation (PRNG) algorithm termed SHA1PRNG. The hash function is used to generate a stream of random numbers. The SK of an advanced encryption standard (AES) is used to encrypt the EHR. Each user in the blockchain obtains key pairs (PuK, PrK) by hashing a random number (RN) utilizing the 256-bit SHA-1 hash function to complete data sharing transactions. The key pair of the Rivest-Shamir-Adleman (RSA) asymmetric key encryption was used to encrypt SK and sign the original EHR.

Data storing: After generating all the keys, the EO encrypts the EHR using the SK to get the ciphertext C_{EHR} , then encrypts his/her symmetric encryption key using the public key PuK to get the ciphertext key C_K , as shown in Equations (1) – (2).

$$C_{EHR} = \text{Enc}_{EHR} (EHR_i (i \in [1;8]), SK) \quad (1)$$

$$C_K = \text{Enc}_{Key} (SK, \text{PuK}) \quad (2)$$

After that, he/she creates a hash of the encrypted EHR to be signed using Equation (3), where MD is the message digest. The private key is then used to sign the MD, and the encrypted hash is the digital signature (SIG). When the signature algorithm is completed, the EO sends the encryption EHR (C_{EHR}) to the cloud storage, as described in Equation (4).

$$MD = H(C_{EHR}) \quad (3)$$

$$SIG = (MD, \text{PrK}) \quad (4)$$

Then, he/she sends both SIG and encrypted keys (C_K) to the blockchain. Besides, he/she sends the access permissions to the smart contract as presented in 3.5 below. For example, all users' public keys are stored in the system database. If B (the owner) wants to share data with C (add C to the list of approved users), the SK will be re-encrypted with the public key of C. When C needs to access data, he/she can decrypt them using its private key. Because only C has access to C's private key, no one else can decrypt the data. The storage process is shown in Algorithm 1.

Algorithm 1. Data storing level.

```

1: Input  $EHR_i$ , Access control, PuK, PrK, SK, SHA-2
2: For each EHR data do
3:   Use SK to encrypt EHR  $C_{EHR} = \text{Enc}_{EHR} (EHR_i (i \in [1 ; 8]), SK)$ .
4:   Use PuK to encrypt SK,  $C_K = \text{Enc}_{Key} (SK, \text{PuK})$ .
5:   Use SHA-2 to create MD on encrypted EHR,  $MD = H(\text{Enc}_{EHR})$ .
6:   Use PrK to sign MD,  $SIG = (MD, \text{PrK})$ .
7:   Store user's Puk in the system's database.
8:   Upload  $C_{EHR}$  to the CS.
9:   Upload  $C_K$  and SIG to the BC.
10: End for;
11: Output  $C_{EHR}$ ,  $C_K$ , SIG.
```

Data sharing: The EHR owner predefines access rights in smart contracts, such as access privileges, access actions and access rights (e.g., read, write), to ensure the secure sharing of EHR.

The smart contract is activated immediately before the access condition is met, which will ensure the validity and fairness of the sharing of data to implement the corresponding procedure. The process of EHR sharing consists of the following two sections:

A. *Blockchain access:*

- *EHR access request:* The EU initiates a blockchain network EHR exchange request (Req) transaction. The access target (ID), access EHR_i and PrK must be included in the request, as shown in Equation (5). The blockchain receives the transaction request and verifies the EU's identification. Only the EU is fair, and the transaction data will be stored in the blockchains.

$$\text{Req} = (\text{ID} \parallel \text{EHR}_i \parallel \text{PrK}); i \in [1; 8]) \quad (5)$$

- *Execution of the smart contract:* If the Req is valid, the SK will be decrypted using the EU's private key and sent to the user, as described in Equation (6).

$$\text{SK} = \text{Dec}_{\text{CK}} (\text{C}_k, \text{PrK}) \quad (6)$$

B. *Cloud storage EHR sharing:*

As seen in Algorithm 2, the EU will recover the EHR_i from the cloud. Then, to achieve the integrity and authenticity of the EHR, the EU creates a hash of the encrypted EHR, MD₂, as shown in Equation (7). Then, he/she uses the EO's public key to decrypt the SIG; the result of the decryption is shown in Equation (8).

$$\text{MD}_2 = \text{H}(\text{C}_{\text{EHR}}) \quad (7)$$

$$\text{Dec}_{\text{SIG}} = (\text{SIG}, \text{PK}) \quad (8)$$

If this decrypted MD matches MD₂, the signature is correct, and the EU will decrypt the EHR and perform its access action, as described in Equation (9). If not, the user can inform the system that the data may have been changed.

$$\text{EHR}_i = \text{Dec}_{\text{CEHR}} (\text{C}_{\text{EHR}}, \text{SK}) \quad (9)$$

Algorithm 2. Data sharing level.

```

1: Input SK, PrK.
2: If Req is not valid then
3:   'return failure'.
4: Else
5:   Decrypt EncKey, SK = DecCK (Ck, PrK).
6:   Retrieve CEHR from the CS.
7:   Create MD2 = H(CEHR).
8:   Decrypt SIG, DecSIG = (SIG, PK) to get the MD.
9:   IF the two MD do not match then
10:    'return failure'.
11:  Else
12:    Decrypt CEHR, EHRi = DecCEHR (CEHR, SK).
13:  End If
14: End If
13: Output EHRi.
```

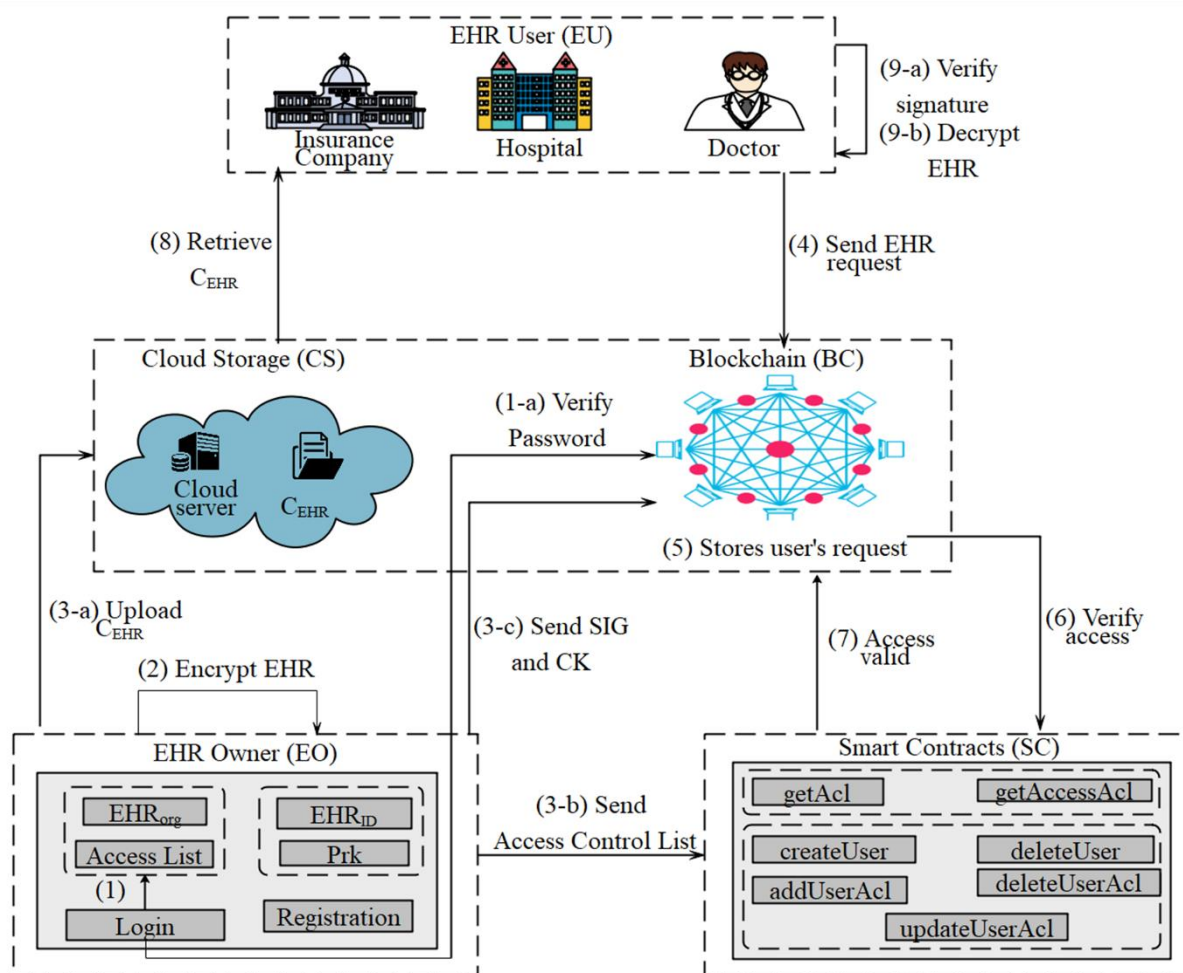



Figure 3. System workflow.

3.5 Smart contract implementation

Smart contracts are reusable and modular pieces of code used to automate any task on the blockchain when certain circumstances are satisfied. The most well-known application of smart contracts is Ethereum. Smart contracts typically use the Solidity programming language, which may be used to program any operation a programmer needs to perform. Programmers can compile the appropriate operations using Ethereum Virtual Machine (EVM) bytecode after programming them (Shahnaz et al., 2019). Furthermore, after constructing them, they may be run and deployed on Ethereum. When the smart contract is implemented, the programmer's bytecode is recorded in the blockchain, and there is a storage address for it.

The smart contract is an important part of the proposal as it performs basic operations. The BAcP-EHR contract allows users access, performs CRUD operations on patients' EHRs and defines roles for gaining access to these features. The `createUser` function is one of the EO's CRUD operations, allowing a new user to be added to the system. `UpdateUser` is a function for updating a user's data in the system. After the EO deletes a user from the system using the `DeleteUser` function, the `removeAclUser` function revokes the user's network authorization. The `addUserToAcl` function is used by the EHR sharing layer to grant a user access to an EHR. The `UpdateUserAcl` method is used to change a user's network role. The `DeleteDoc` method deletes the document from the system and its users and access. The BAcP-EHR smart contract code is given in Figure 4.

```

pragma solidity ^0.4.24;
contract aclService {
    struct User {
        bytes32 password; bytes32 login;
        bytes32 pke2; uint8 createdby;
        uint8 user_id;
    }
    mapping(address => User) private userDictionary;
    struct Acl {
        uint8 doc_id; address account;
        bytes32 crypted_AES;
        bytes32 doc_signature;
        bytes32 accesstype;
    }
    mapping(uint8 => Acl) private AD;
    struct Doc {
        address _owner; uint8 doc_id;
        bytes32 signature;
    }
    mapping(uint8 => Doc) private shareDoc;
    function createUser(address _account,
        bytes32 _password, bytes32 _login,
        bytes32 _pke2, uint8 _createdby,
        uint8 _user_id) external {
        User memory _user = User({password:_password,
            login:_login, pke2:_pke2,
            createdby:_createdby,
            user_id:_user_id});
        userDictionary[_account] = _user;
    }
    function addtouseracl(uint8 _acl_id, uint8 _doc_id,
        address _account, bytes32 _crypted_AES,
        bytes32 _doc_signature,
        bytes32 _accesstype) external {
        Acl memory _acl = Acl({doc_id:_doc_id,
            account:_account, crypted_AES:_crypted_AES,
            doc_signature:_doc_signature,
            accesstype:_accesstype});
        AD[_acl_id] = _acl;
    }
    function getACL(uint8 _acl_id) external view returns
        (uint8 _doc_id, address _account, bytes32
            _crypted_AES, bytes32 _doc_signature,
            bytes32 _accesstype) {
        return (AD[_acl_id].doc_id,
            AD[_acl_id].account,
            AD[_acl_id].crypted_AES,
            AD[_acl_id].doc_signature,
            AD[_acl_id].accesstype);
    }

    function auth(address _account) external view returns
        (bytes32 _password, bytes32 _login, bytes32 _pke2) {
        return (userDictionary[_account].password,
            userDictionary[_account].login,
            userDictionary[_account].pke2);
    }
    function getaccesstype(uint8 _acl_id) external view
        returns (bytes32 _accesstype) {
        return AD[_acl_id].accesstype;
    }
    function updateacl(uint8 _acl_id, uint8 _doc_id, address
        _account, bytes32
            _crypted_AES, bytes32 _doc_signature, bytes32
            _accesstype) external {
        Acl memory _acl = Acl({doc_id:_doc_id,
            account:_account, crypted_AES:_crypted_AES,
            doc_signature:_doc_signature, accesstype:_accesstype});
        AD[_acl_id] = _acl;
    }
    function updateuser(address _account,
        bytes32 _password, bytes32 _login, bytes32
            _pke2, uint8 _createdby,
            uint8 _user_id) external {
        userDictionary[_account] = User({password:_password,
            login:_login, pke2:_pke2,
            createdby:_createdby,
            user_id:_user_id});
    }
    function deleteUser(address _account) external {
        delete userDictionary[_account];
    }
    function deleteAcl(uint8 _acl_id) external {
        delete AD[_acl_id];
    }
    function createDoc(address _owner, uint8 _doc_id,
        bytes32 _signature) external {
        Doc memory _doc = Doc({_owner:_owner,
            doc_id:_doc_id, signature:_signature});
        shareDoc[_doc_id] = _doc;
    }
    function updateDoc(address _owner, uint8 _doc_id,
        bytes32 _signature) external {
        Doc memory _doc = Doc({_owner:_owner,
            doc_id:_doc_id, signature:_signature});
        shareDoc[_doc_id] = _doc;
    }
    function deleteDoc(uint8 _doc_id) external {
        delete shareDoc[_doc_id];
    }
    function getDoc(uint8 _doc_id) external view returns
        (address _owner, uint8 _doc_id, bytes32 _signature) {
        return (shareDoc[_doc_id]._owner, shareDoc
            [_doc_id].doc_id, shareDoc[_doc_id].signature);
    }
}

```

Figure 4. BAcP-EHR smart contract code.

4 Performance

The performance of the proposed scheme is analysed by comparing it to the current works of Li et al. (2018), Thwin and Vasupongayya (2019), Wang et al. (2018), Sahoo and Baruah (2018), Khalaf et al. (2020) and Chen et al. (2019). All the works consist of several components, such that the tests are divided into three main categories: a cryptographic test, the costs of the smart contract and a security analysis.

4.1 Experimental setup

To test the performance of the proposed framework, we have implemented it on the Windows 10 system on an ADM Ryzen 3 2300 CPU @ 2 GHz, and 4.00 GB RAM using Java, and Solidity is the programming

language of Ethereum smart contracts. Our application programming interfaces communicate with Ethereum as the blockchain technology and Google Cloud Platform Storage (GCP), which serves as the cloud storage server in our implementation. We conducted these experiments on public datasets downloaded from the US Department of Health and Human Services (HealthData.gov, 2022).

4.2 Experimental results

The proposed system was assessed for the following cases:

4.2.1 Encryption and decryption time

We first tested the time consumption of encryption and decryption of blockchain works for different sizes of EHRs. The times required for encryption and decryption process times of Thwin and Vasupongayya (2019), Wang et al. (2018) and BAcP-EHR are shown in Table 2. As seen in Figure 5, the encryption and decryption times of the BAcP-EHR scheme are proportional to the EHR size. As the EHR size increases, the encryption and decryption times also increase. Furthermore, it can also be seen in Figure 5 that when the size of the EHR is less than 128 MB, the time spent on the encryption and decryption process is less than 1 s. Even if the EHR size is expanded to 128 MB, the additional time spent is around 1s.

We compared the test findings in Figure 5 with the encrypting and decrypting time consumption in Thwin and Vasupongayya (2019) and Wang et al. (2018). The results of the comparison are shown in Figure 6 and Figure 7. The time consumed by the encryption and decryption process is less than in the other works. When the EHR is large, our encryption and decryption efficiencies are significantly higher than those of Thwin and Vasupongayya (2019) and Wang et al. (2018). The EHRs contain many large image files, such as X-rays and CT scans, and depending on these comparisons, our scheme is more suitable than previous health record encryption and decryption work.

Table 2. Encryption and decryption comparison.

Work	Wang et al. (2018)		Thwin and Vasupongayya (2019)		BAcP-EHR	
File size	Encryption	Decryption	Encryption	Decryption	Encryption	Decryption
128 KB	0.36646	0.16169	0.0918	0.00319	0.0012	0.0013
512 KB	0.37069	0.17001	0.094	0.0064	0.0158	0.0027
2 MB	0.37585	0.17967	0.101	0.01662	0.0452	0.0157
8 MB	0.42311	0.22602	0.142	0.05919	0.0615	0.048
32 MB	0.59305	0.40503	0.303	0.23833	0.2064	0.20123
128 MB	2.24242	1.95048	1.828	1.81479	1.4149	1.6284

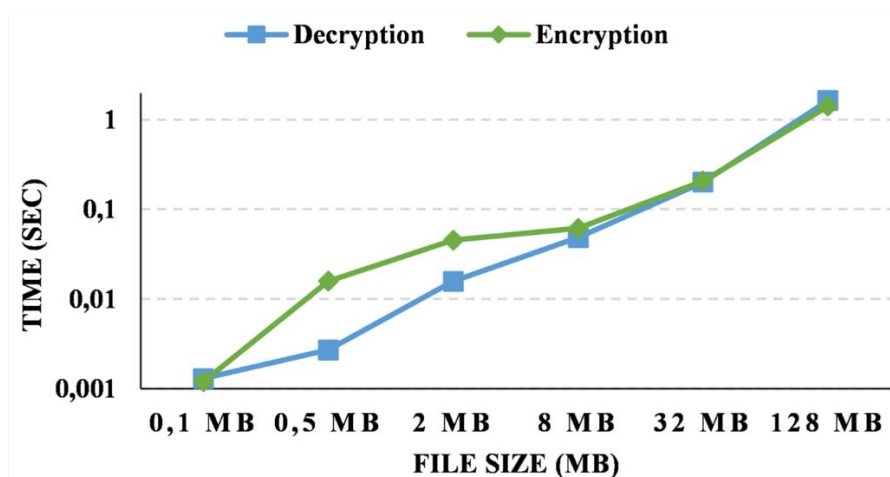


Figure 5. BAcP-EHR scheme encryption and decryption.

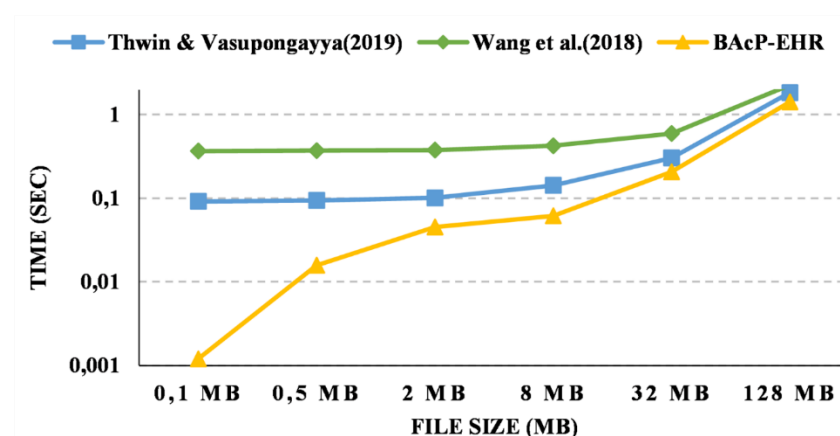


Figure 6. Encryption process comparison.

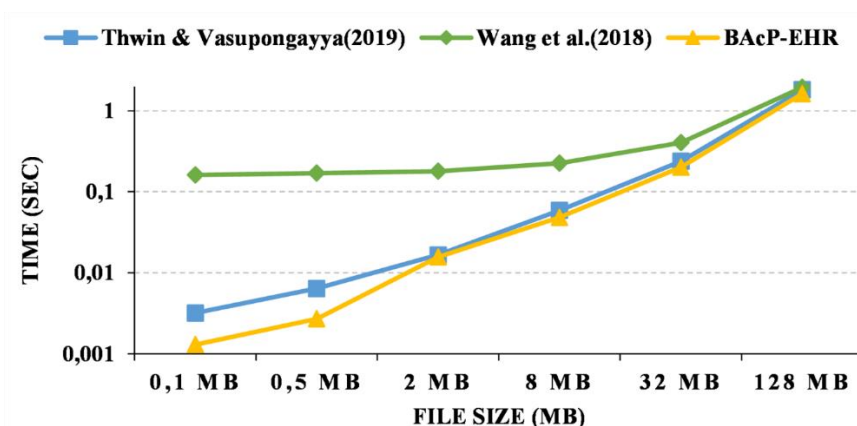


Figure 7. Decryption process comparison.

4.2.2 Smart contract costs

The BAcP-EHR scheme uses Remix as the smart contract development tool, writes the smart contract using the Solidity language, and deploys the compiled contract in the Ethereum Rinkeby test network. We used Metamask and Etherscan to calculate the real gas costs for each function in the smart contract. In the Ethereum blockchain, fees are the gas corresponding to the payment or value of the price needed for each successful transaction or execution of a contract. If a user does not have an account with a valid balance, he/she cannot perform any service and the transaction is considered invalid. Figures 8 and 9 show

respectively the deployment and costs of our proposed contract in Remix IDE and the Metamask software cryptocurrency wallet, as well as the block in the Ganache Ethereum. The smart contract cost results are presented in Table 3.

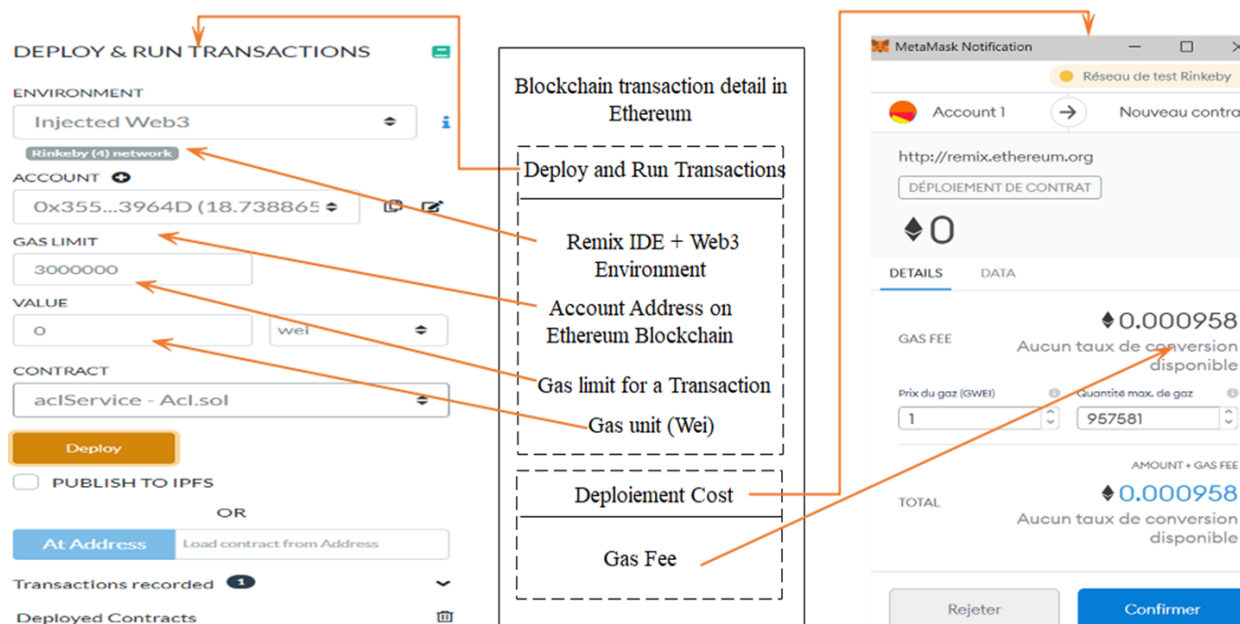


Figure 8. Smart contract costs.

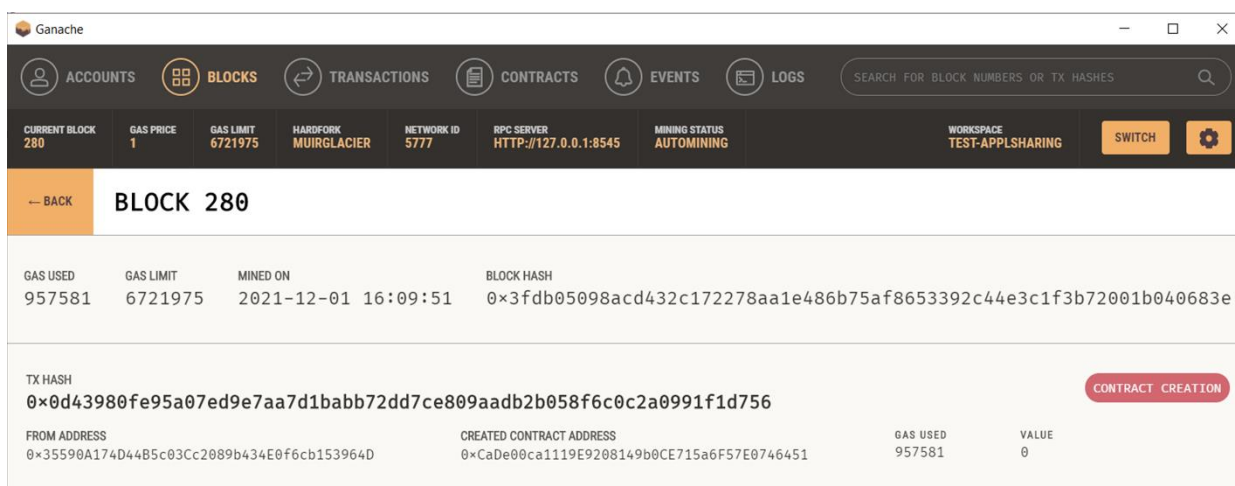


Figure 9. Ethereum smart contract block.

It is not difficult to conclude from the test results in Table 3 that our proposed contract requires fewer costs of deployment and invocation of other functions.

Table 3. Smart contract costs.

CRUD operations	Experimental result		
	Contact functions	Gas used	Cost (ether)
CRUD user	deployContract	957581	0.000957581
	createUser	106231	0.000106
	updateUser	42096	0.000042
	deleteUser	22204	0.000022

<i>CRUD access</i>	addUserToAcl	106285	0.000106
	updateUserAcl	37753	0.000038
	deleteUserAcl	22057	0.000022
<i>CRUD document</i>	addDoc	106285	0.000106285
	updateDoc	27094	0.000027
	deleteDoc	17141	0.000017

5 Discussion

Several authors have proposed blockchain-based healthcare systems to secure the sharing of healthcare data. Unlike Zhao et al. (2019), who encrypted all files and stored them on the blockchain, this article makes use of cloud storage technologies to relieve pressure on blockchain storage and meet real-world deployment requirements. Chen et al. (2019) developed a system in which the patient must provide a private key for the doctor to access data; however, the transmission mechanism cannot guarantee the confidentiality of the private key, and hostile nodes can access it.

Contrary to Qin et al. (2021) and Xia et al. (2017), we used a hybrid encryption approach to ensure better security and get the benefits of each encryption. The encryption keys used in our system are completely secure. Furthermore, the original EHR is encrypted before being sent to a cloud server. This allows a solution to the blockchain's limited storage capacity as well as a significant reduction in the risk of confidential information posed by original electronic health data being leaked.

Several proposed systems can meet the requirements for privacy and integrity, but not all of them can provide access control, which is a critical security goal in an EHR sharing system. Moreover, smart contracts are only used in a few proposed systems. In contrast to Qin et al. (2021), Khalaf et al. (2020), Li et al. (2018), Sahoo and Baruah (2018), Liang et al. (2017), Ramani et al. (2018) and Abunadi and Kumar (2021), our solution uses the smart contract to securely store the encryption key, preserve the EHR signature, verify authentication and manage the user's access control.

The system we offer protects the patient's privacy by allowing them to specify granular access controls to their EHR data using smart contracts. Furthermore, it is based on a decentralized network topology with a single point of failure. Unlike Li et al. (2018), Sahoo and Baruah (2018) and Chen et al. (2019), our solution relies on defined user roles to protect EHR data from security threats such as unauthorized access. As a result, malicious users will be unable to access EHR data. It also allows quick and secure access to EHR data based on the patient's preferences. It ensures the availability of EHR data elements without the need for third-party validation.

In terms of functions such as personal data security, access control and data integrity, our system outperforms these similar studies. Furthermore, it presents a feasible alternative for updating existing electronic healthcare systems, including hybrid encryption, encryption keys and smart contracts.

5.1 Security analysis

This part demonstrates how the BAcP-EHR scheme effectively ensures security objectives compared to other works. The result of the comparison is presented in Table 4. '✓' and 'X' indicate whether the literature supports this function or not.

Confidentiality ensures that those who are not allowed to know do not share the content. In our scheme, patients' EHRs are encrypted and accessible only to authorized individuals. Data can only be obtained from the cloud server by the entity that obtains authorization, while details are in the EHR sharing subsection.

The blockchain account is anonymous and unlinkable from a true identity by exploiting the blockchain and smart contract encryption capability. Thus, the blockchain privacy can prevent public information from revealing the true identity of individuals. Our access control system ensures data privacy of individuals. Malicious access is blocked by the user identification capability and authorization in smart contracts, restricting access to our cloud servers from future attacks.

The integrity guarantees that patient information is exchanged between approved users without any modifications. The electronic health records are still encrypted in the proposal to prevent modification. Meanwhile, EHR users cannot change transactions signed in smart contracts for EHR sharing, and no individual can alter and modify the quality of transactions registered. Besides, every node (or service node for better efficiency in practice) has a copy of the blockchain data, so if a node connects with other nodes, a shift in a certain block will be easily detected. Most importantly, in our case, EHR users do not have the right to change or update the smart contract and access policies. Therefore, Merkle's root value would change if the EHR is changed, allowing the block content to change. Therefore, EHRs can be maintained securely and correctly by using the blockchain without being tampered with.

The patients must sign their EHR for authentication before adding the associated contracts to the blockchain. Data authenticity can be provided based on the signatures and the integrity provided by the blockchain.

To allow a user to access a patient's EHRs, the user sends the request to what he/she is interested in. If the user can access the data, it will be sent back with the requested EHRs that are encrypted, which ensures access control.

Table 4. Comparison of BAcP-EHR scheme with related works.

Function	Li et al. (2018)	Sahoo and Baruah (2018)	Khalaf et al. (2020)	Chen et al. (2019)	BAcP-EHR
Blockchain-based	✓	✓	✓	✓	✓
Privacy	✓	✗	✓	✓	✓
Integrity	✗	✓	✗	✓	✓
Access control	✗	✗	✓	✗	✓
Authentication	✓	✗	✓	✓	✓
Cryptographic function	✓	✗	✓	✓	✓
Smart contract	✗	✗	✗	✓	✓

5.2 Key features of BAcP-EHR

The key features of the suggested solution to ensure privacy and security of EHRs in the cloud are as follows:

- The security and privacy of health data outsourced to a public cloud are enhanced by using existing security techniques, including asymmetric encryption (RSA), symmetric encryption (AES) and hash algorithms.
- Data owners create security and privacy protection for their data before they are outsourced to the cloud.
- The user's login is verified by the blockchain.
- Access control permissions and data integrity are attached to each health record independently of other records; only the data owners can define and manage these accesses.

- The cloud is used only to store encrypted EHRs, the cloud provider is not required to store or expose any information about the data.
- The data owner is able to use a unique key to encrypt each EHR by securely attaching the key to the data, and authorized users can retrieve this key while applying secure access.
- Only authorized users can verify the integrity and authenticity of medical records.
- The confidentiality and integrity of EHRs remain protected in the cloud environment from any internal or external security attack.
- Any potential adversary who obtains a protected EHR cannot read or reveal its contents.
- The proposed solution is simple and does not require complicated operations.
- Implementation and evaluation of the proposed solution shows that it can be used practically and effectively.

6 Conclusion

A blockchain-based access control scheme for preserving privacy of electronic health records (EHR) scheme is proposed in this paper to achieve privacy, confidentiality, integrity and access control. The system uses the Ethereum blockchain technology and the cloud to ensure that electronic records are stored safely by specifying granular user access rules. First, we propose a framework for sharing the EHR among various entities focused on blockchain and cloud storage. In this paper, cloud storage stores the encrypted EHR while EHR signatures are stored on the Ethereum EHR blockchain. Then, the access controls for EHR are presented in the form of Ethereum blockchain smart contracts to ensure efficient access to EHRs in the system. Moreover, we use symmetric and asymmetric encryption to guarantee data confidentiality and achieve data sharing with privacy preservation. Furthermore, we implement the proposal for an Ethereum platform and evaluate the performance of the computation.

Our proposal can meet the defined security objectives according to the performance results, the security analysis and the security proof of the proposed scheme. As future work, we consider several possible extensions or improvements to our system. We are inclined to study a fraud detection system in healthcare based on the blockchain.

Additional Information and Declarations

Acknowledgements: This research is partially supported by the 'Projects de Recherche Formation Universitaire (PRFU)' under the number C00L07UN250220200002.

Conflict of Interests: The authors declare no conflict of interest.


Author Contributions: I.B.: Conceptualization, Methodology, Writing – original draft, Writing – review & editing. K.Z.: Supervision, Writing – review & editing.

References

- AbuKhoua, E., Mohamed, N., & Al-Jaroodi, J. (2012). e-Health Cloud: Opportunities and Challenges. *Future internet*, 4(3), 621–645. <https://doi.org/10.3390/fi4030621>
- Abunadi, I., & Kumar, R. L. (2021). BSF-EHR: blockchain security framework for electronic health records of patients. *Sensors*, 21(8), Article no. 2865. <https://doi.org/10.3390/s21082865>
- Agbo, C., Mahmoud, Q., & Eklund, J. (2019). Blockchain Technology in Healthcare: A Systematic Review. *Healthcare*, 7(2), 56. <https://doi.org/10.3390/healthcare7020056>
- Boumezeur, I., & Zarour, K. (2018). Privacy Preserving Requirements for Sharing Health Data in Cloud. In *International Conference Europe Middle East & North Africa Information Systems and Technologies to Support Learning* (pp. 412-423). Springer. https://doi.org/10.1007/978-3-030-03577-8_46
- Capitalone. (2022). *Capital main page*. <https://www.capitalone.com/>
- Catalini, C., & Gans, J. S. (2020). Some simple economics of the blockchain. *Communications of the ACM*, 63(7), 80–90. <https://doi.org/10.1145/3359552>

- Chen, L., Lee, W.K., Chang, C.C., Choo, K.K. R., & Zhang, N. (2019). Blockchain based searchable encryption for electronic health record sharing. *Future generation computer systems*, 95, 420–429. <https://doi.org/10.1016/j.future.2019.01.018>
- Cheng, E. C., Le, Y., Zhou, J., & Lu, Y. (2018). Healthcare services across China—on implementing an extensible universally unique patient identifier system. *International Journal of Healthcare Management*, 11(3), 210–216. <https://doi.org/10.1080/20479700.2017.1398388>
- FACTOM. (2022). FACTOM main page. <https://www.factom.com/>
- Gem. (2022). Gem main page. <https://gem.com>
- Gordon, W. J., & Catalini, C. (2018). Blockchain Technology for Healthcare: Facilitating the Transition to Patient-Driven Interoperability. *Computational and Structural Biotechnology Journal*, 16, 224–230. <https://doi.org/10.1016/j.csbj.2018.06.003>
- Hardin, T., & Kotz, D. (2019). Blockchain in health data systems: A survey. In 2019 sixth international conference on internet of things: Systems, management and security (pp. 490–497). IEEE. <https://doi.org/10.1109/IOTSMS48152.2019.8939174>
- HealthData.gov. (2022) Washington: Department of health and human services. <https://www.va.gov/bluebutton>
- HealthNautica. (2022). HealthNautica main page. <https://www.healthnautica.com/comppages/index.asp>
- HIMSS. (2020). Digital health. <https://www.himss.org/resources/personal-health-records-electronic-health-records-key-indias-national-digital-health>
- Hölbl, M., Kompara, M., Kamišalić, A., & Nemec Zlatolas, L. (2018). A systematic review of the use of blockchain in healthcare. *Symmetry*, 10(10), 470. <https://doi.org/10.3390/sym10100470>
- Khalaf, O. I., Abdulsahib, G. M., Kasmaei, H. D., & Ogudo, K. A. (2020). A New Algorithm on Application of Blockchain Technology in Live Stream Video Transmissions and Telecommunications. *International Journal of E-Collaboration*, 16(1), 16–32. <https://doi.org/10.4018/ijec.2020010102>
- Li, H., Zhu, L., Shen, M., Gao, F., Tao, X., & Liu, S. (2018). Blockchain-Based Data Preservation System for Medical Data. *Journal of Medical Systems*, 42(8). <https://doi.org/10.1007/s10916-018-0997-3>
- Liang, X., Zhao, J., Shetty, S., Liu, J., & Li, D. (2017). Integrating blockchain for data sharing and collaboration in mobile healthcare applications. In 2017 IEEE 28th annual international symposium on personal, indoor, and mobile radio communications (pp. 1–5). IEEE. <https://doi.org/10.1109/PIMRC.2017.8292361>
- Mayer, A. H., da Costa, C. A., & Righi, R. da R. (2020). Electronic health records in a Blockchain: A systematic review. *Health Informatics Journal*, 26(2), 1273–1288. <https://doi.org/10.1177/1460458219866350>
- Pournaghi, S. M., Bayat, M., & Farjami, Y. (2020). MedSBA: a novel and secure scheme to share medical data based on blockchain technology and attribute-based encryption. *Journal of Ambient Intelligence and Humanized Computing*, 11(11), 4613–4641. <https://doi.org/10.1007/s12652-020-01710-y>
- Qin, Q., Jin, B., & Liu, Y. (2021). A Secure Storage and Sharing Scheme of Stroke Electronic Medical Records Based on Consortium Blockchain. *BioMed Research International*, 1–14. <https://doi.org/10.1155/2021/6676171>
- Rajput, A. R., Li, Q., Ahvanooy, M. T., & Masood, I. (2019). EACMS: Emergency access control management system for personal health record based on blockchain. *IEEE Access*, 7, 84304–84317. <https://doi.org/10.1109/ACCESS.2019.2917976>
- Ramani, V., Kumar, T., Bracken, A., Liyanage, M., & Ylianttila, M. (2018). Secure and efficient data accessibility in blockchain based healthcare systems. In 2018 IEEE Global Communications Conference (pp. 206–212). IEEE. <https://doi.org/10.1109/GLOCOM.2018.8647221>
- Alam, S., Ahmad Reegu, F., Daud, S. M., & Shuaib, M. (2021). *Blockchain-Based Electronic Health Record System for Efficient Covid-19 Pandemic Management*. <https://doi.org/10.20944/preprints202104.0771.v1>
- Sahoo, M. S., & Baruah, P. K. (2018). HBasechainDB – A Scalable Blockchain Framework on Hadoop Ecosystem. In *Asian Conference on Supercomputing Frontiers* (pp. 18–29). Springer. https://doi.org/10.1007/978-3-319-69953-0_2
- Sauermann, S., Frohner, M., Urbauer, P., Forjan, M., Pohn, B., Drauschke, B.A., & Mense, A. (2013). The adolescence of electronic health records: Status and perspectives for large scale implementation. *Acta Informatica Pragensia*, 2(1), 30–38. <https://doi.org/10.18267/j.aip.11>
- Shahnaz, A., Qamar, U., & Khalid, A. (2019). Using blockchain for electronic health records. *IEEE Access*, 7, 147782–147795. <https://doi.org/10.1109/ACCESS.2019.2946373>
- Shuaib, K., Saleous, H., Shuaib, K., & Zaki, N. (2019). Blockchains for secure digitized medicine. *Journal of personalized medicine*, 9(3), Article no. 35. <https://doi.org/10.3390/jpm9030035>
- Shuaib, M., Daud, S. M., Alam, S., & Khan, W. Z. (2020). Blockchain-based framework for secure and reliable land registry system. *Telkomnika*, 18(5), 2560–2571. <https://doi.org/10.12928/TELKOMNIKA.v18i5.15787>
- Siddiqui, S. T., Ahmad, R., Shuaib, M., & Alam, S. (2020). Blockchain security threats, attacks and countermeasures. In *Advances in Intelligent Systems and Computing* (pp. 51–62). Springer. https://doi.org/10.1007/978-981-15-1518-7_5
- Thwin, T. T., & Vasupongayya, S. (2019). Blockchain-Based Access Control Model to Preserve Privacy for Personal Health Record Systems. *Security and Communication Networks*, 2019, 1–15. <https://doi.org/10.1155/2019/8315614>
- Wang, H., & Song, Y. (2018). Secure Cloud-Based EHR System Using Attribute-Based Cryptosystem and Blockchain. *Journal of Medical Systems*, 42(8). <https://doi.org/10.1007/s10916-018-0994-6>

-
- Xia, Q., Sifah, E. B., Smahi, A., Amofa, S., & Zhang, X.** (2017). Bbds: Blockchain-based data sharing for electronic medical records in cloud environments. *Information*, 8(2), Article no. 44. <https://doi.org/10.3390/info8020044>
- Zhang, A., & Lin, X.** (2018). Towards Secure and Privacy-Preserving Data Sharing in e-Health Systems via Consortium Blockchain. *Journal of Medical Systems*, 42(8). <https://doi.org/10.1007/s10916-018-0995-5>
- Zhao, Y., Cui, M., Zheng, L., Zhang, R., Meng, L., Gao, D., & Zhang, Y.** (2019). Research on electronic medical record access control based on blockchain. *International Journal of Distributed Sensor Networks*, 15(11), 155014771988933. <https://doi.org/10.1177/1550147719889330>
-

Editorial record: The article has been peer-reviewed. First submission received on 3 December 2021. Revisions received on 15 January 2022, and 6 February 2022. Accepted for publication on 17 February 2022. The editor in charge of coordinating the peer-review of this manuscript and approving it for publication was Michal Dolezel .

Acta Informatica Pragensia is published by Prague University of Economics and Business, Czech Republic.

ISSN: 1805-4951
