Attribute-based Multi-Signature and Encryption for EHR Management: A Blockchain-based Solution

Hao Guo[1] Wanxin Li[2] Ehsan Meamari[1] Chien-Chung Shen[1] Mark Nejad[2]
[1]Department of Computer and Information Sciences [2]Department of Civil and Environmental Engineering
University of Delaware, U.S.A.
{haoguo,wanxinli,ehsan,nejad,cshen}@udel.edu

***Abstract*—The global Electronic Health Record (EHR) market is growing dramatically and has already hit $31.5 billion in 2018. To safeguard the security of EHR data and privacy of patients, fine-grained information access and sharing mechanisms are essential for EHR management. This paper proposes a hybrid architecture of blockchain and edge nodes to facilitate EHR management. In this architecture, we utilize attribute-based multi-signature (ABMS) scheme to authenticate user's signatures without revealing the sensitive information and multi-authority attribute-based encryption (ABE) scheme to encrypt EHR data which is stored on the edge node. We develop the blockchain module on Hyperledger Fabric platform and the ABMS module on Hyperledger Ursa library. We measure the signing and verifying time of the ABMS scheme under different settings, and experiment with the authentication events and access activities which are logged as transactions in blockchain.**

***Index Terms*—Attribute-based Multi-Signature, Attribute-based Encryption, Blockchain, Edge Node, Hyperledger Fabric, Hyperledger Ursa.**

I. Introduction

Electronic health records (EHR), although containing private information for patient diagnosis and treatment, need to be frequently shared among different participants, such as healthcare providers, insurance companies, pharmacies, and medical researchers [1]. Therefore, one big challenge to EHR management is to gather, store and share personal healthcare information without violating privacy or compromising security. In the context of EHR data management, although healthcare providers (termed data users), such as doctors, nurses, phlebotomists, medical laboratory scientists, pathologists, etc., need to authenticate the patients (termed data owners), not all of them (for instance, medical laboratory scientists) need to know patients' identity. In addition, patients' healthcare information need to be protected and the patients would define access policy specifying who have the permission to access what information.

To authenticate patients while guaranteeing anonymity, the scheme of Attribute-based Signature (ABS) [2] can be a potential solution. Using ABS, a signature signed with a patient's attributes is attested not to the identity of patient but instead to the attributes possessed by patient. However, ABS is not flexible to update the policy embedded in the signature since it adopts a one-time signature generation process.

978-1-7281-6680-3/20/$31.00

To secure EHRs while facilitating versatile access and sharing among healthcare providers, the method of Attribute-based Encryption (ABE) [3] can be a candidate solution. ABE is a public-key encryption scheme that binds security directly to EHRs and the participants who access it by enforcing attribute-based access control. Instead of encrypting EHRs for a specific data user, ABE allows encrypted EHRs to be accessed by any user with proper attributes satisfying the access policy.

Recently, the technology of Blockchain [4] has been proposed as a solution for EHR management [5], [6], [7], [8]. In this context, a blockchain can be used as a tamper-proof log to record both the authentication events on the patients and the access activities of EHRs by the data users. However, to maximize the capability of blockchain-based EHR management solutions, the following issues need to be addressed. First, the privacy of patients and the security of their EHRs. Due to the decentralized and transparent nature of blockchain, any identity and sensitive information of the patients should not be stored directly into blockchain transactions. Second, the size of blocks in a blockchain. Typically, the size of blocks in a blockchain is too limited to accommodate EHRs containing images of X-ray, CT scans, and MRI, and videos of ultrasound.

This paper proposes the Attribute-based Multi-Signature (ABMS) scheme to authenticate patients anonymously. The paper also describes the adoption of ABE to secure EHRs with policy-based access control. Both schemes are integrated into a hybrid architecture of blockchain and edge computing to facilitate EHR management. Specifically, authentication events of patients and EHR access activities are recorded into the blockchain, i.e., *on-chain*, for traceability and accountability. In collaboration, edge nodes, which function as *off-chain* storage, store ABE-encrypted EHR data, so that only eligible data users satisfying the specific EHR access policies can decrypt and access the EHR data.

We prototype the designed hybrid architecture by using the Hyperledger Fabric [9] and the Hyperledger Ursa [10]. In addition, the Access Control Lists (ACL) mechanism is used to facilitate attribute-based access control of patient profiles. Using the prototype, we conduct experiments to validate the operations of the smart contracts and policy-based access control. We also evaluate the performance of the signing and verification of multiple signatures of ABMS and blockchain transactions under different application settings.

II. System Architecture

In this section, we describe the hybrid blockchain-edge architecture consisting of the multi-authority ABE and ABMS schemes. By referring to Fig. 1, we define the following entities that take part in the proposed architecture.

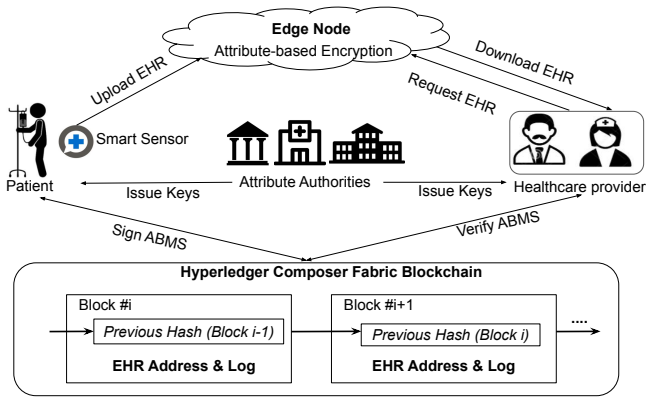

Fig. 1: Proposed System Architecture.

- EHR data: EHR data is a piece of information owned by a patient, and can be accessed by authorized healthcare providers who have the access permission.
- Patient: A patient is the data owner of his/her EHR data, who defines the access policy for the data users (e.g., healthcare providers).
- Healthcare provider: A healthcare provider (e.g., doctor and nurse) is a data user who needs to access the EHR data owned by patients. A healthcare provider actively seeks access permissions from patients.
- Attribute: An attribute is a piece of information (e.g., driver's license) that is associated with a participant.
- Attribute authority: An attribute authority is an entity who manages attributes and issues key pairs to data owners and data users.
- Smart sensor: A smart sensor is a device that collects EHR data from the patients and sends it to the edge node. Smart sensors include imaging equipment such as X-ray, CT scan, MRI, and ultrasound.
- Edge node: An edge node is a computing and storage device, which stores ABE-encrypted EHR data.
- Smart contracts and blockchain: Smart contracts take patients' signatures as input and return EHR addresses. The blockchain serves as a tamper-proof log of patient authentication events and EHR access activities.

A. Multi-authority CP-ABE mechanism design

This subsection describes the use of multi-authority CP-ABE [11] in the context of EHR management.

Initial Setup$(\lambda) \longrightarrow IP$. The initial setup takes a security parameter λ as the input and outputs the initial parameter IP for the system.

Authority Setup$(IP, AT_i) \longrightarrow PK_i, SK_i$. Authority i takes the initial parameter IP and attribute AT_i as the input to generate the public key PK_i and secret master key SK_i.

Key Generation$(AT_i, GID, SK_i, IP) \longrightarrow K_{i,GID}$. This key generation algorithm takes as inputs attribute AT_i, the global identification number GID of a data user, the secret master key SK_i, and the initial parameter IP, and returns the attribute key $K_{i,GID}$ to the data user.

Encryption$(IP, M, \mathbb{A}, \{PK_i\}) \longrightarrow CT$. The encryption algorithm takes as inputs the initial parameter IP, EHR data M, access policy $\mathbb{A}$ defined by a patient, and the set of public keys PK_i. It outputs the ciphertext CT as the encrypted EHR data. Moreover, the access policy $\mathbb{A}$ defines which healthcare providers are allowed to access the EHR data of the patient.

Decryption $(CT, IP, \{K_{i,GID}\}) \longrightarrow M$. The decryption algorithm takes as inputs the initial parameter IP, ciphertext CT, and a collection of attribute keys $\{K_{i,GID}\}$. It outputs the EHR M if the set of attributes keys satisfies the access policy for the ciphertext CT. Otherwise, the decryption fails.

B. Multi-authority ABMS mechanism design

By extending [12], [13], the model of multi-authority ABMS is described as below:

Initial Setup$(\lambda) \longrightarrow IP$: The initial setup takes the security parameter λ as input and outputs the initial parameters IP for the system.

Authority Setup$(IP, AT_i) \longrightarrow SIK_i, VK_i$: Each attribute authority executes the setup algorithm with the input of the initial parameter IP and attribute AT_i, and returns the signature key SIK_i and the verification key VK_i for each attribute AT_i.

Extract$(IP, GID, AT_i, SIK_i) \longrightarrow SK_{i,GID}$: This algorithm is executed by the attribute authorities. An attribute authority takes as inputs the initial parameters IP, the owner's unique identity GID, the attribute AT_i, and the authority's signature key SIK_i. In the end, the algorithm returns the signing key $SK_{i,GID}$.

Sign$(H(A_i), IP, SK_{i,GID}) \longrightarrow \sigma_i$: This algorithm is executed n times based on the number of attributes belonging to the data owner (patients). It takes as inputs the hashed attribute value $H(A_i)$, initial parameter IP, and signing key $SK_{i,GID}$, and outputs the signature σ_i.

Verify$(H(A_i), IP, \sigma_i, VK_i) \longrightarrow \{0, 1\}$: This algorithm takes as inputs the hashed attribute value $H(A_i)$, the initial parameter IP, the signature σ_i, and the verification key VK_i. In the end, the algorithm outputs a Boolean value $accept$ or $reject$ to indicate whether the signature from the data singer (patient) with the specific attribute is valid or not without revealing the signer's identity.

In the proposed ABMS scheme, we applied the (t, n) threshold scheme, where $1 \leq t \leq n$, to as shown in Fig. 2. For instance, patient Annie signed three signatures related to her patient ID issued by the hospital, her driver's license issued by DMV, and her insurance ID issued by the insurance company. A medical laboratory scientist, who does test on Annie's blood work, will apply the threshold of *3 out of 3* to authenticate her, while a medical research scientist will apply the threshold of *1 out of 3* to authenticate Annie to access her EHR for data

analysis. For both cases, the aim of ABMS is to authenticate patients anonymously while using different thresholds.

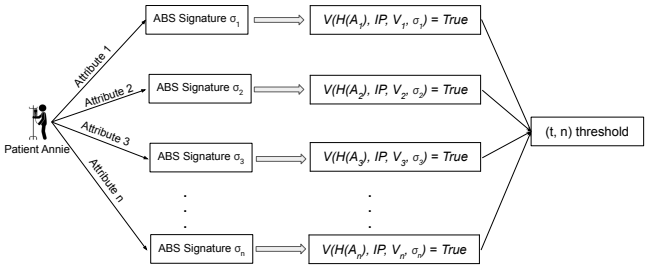

Fig. 2: Multi-signature threshold scheme.

C. On-chain and off-chain hybrid architecture

To overcome the space limitation issue [7], the transaction privacy issues [14], [15], [16], and the issue of lacking proper access control among participants [7], [17] of the blockchain, we propose a hybrid architecture which consists of an on-chain record of ABMS authentication events and EHR access activities (including EHR addresses and other information) as transactions on a blockchain, and an off-chain storage of ABE-encrypted EHR data on the edge nodes, as shown in Fig. 3.

In addition, each patient saves his/her data, including *GID*, name, and ABMS signatures in a private profile. A patient will define access policy for the data users (e.g., healthcare providers) to access his/her profile, which is implemented with the ACL mechanism of the Hyperledger Fabric blockchain.

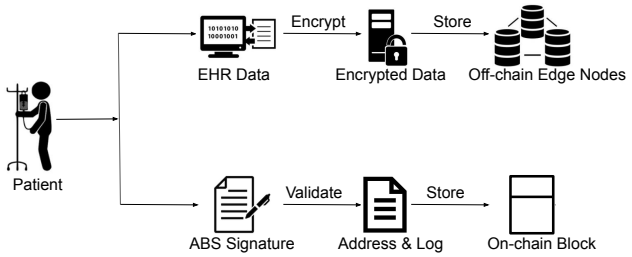

Fig. 3: Hybrid on-chain and off-chain architecture.

D. Workflow of EHR management

Workflow of EHR management is depicted in Fig. 4. First, all the participants register themselves with the EHR management system. Attribute authorities issue ABMS and ABE keys to both patients and healthcare providers based on their attributes. After being granted private keys from multiple attribute authorities, a patient uploads his/her EHR data, which is encrypted via ABE[1], to the edge node. Next, a patient signs his/her hashed attribute values to generate the ABMS signatures, which are stored in his/her profile in the EHR management system. When a data user (e.g., a healthcare provider) needs to access a patient's EHR data, an EHR access request is sent to the blockchain which triggers ABMS signature verification to authenticate the patient. A smart contract is executed to log the authentication event and return a one-time EHR URL to the healthcare provider. After that, the healthcare provider sends the one-time EHR URL to the edge node to retrieve the encrypted EHR data.

[1]In practice, it is not practical for patients to handle (e.g., encrypt) their EHR data personally. To address this issue, smart sensors could be the entities to receive these private keys to ABE-encrypt EHR data before uploading it to the edge node.

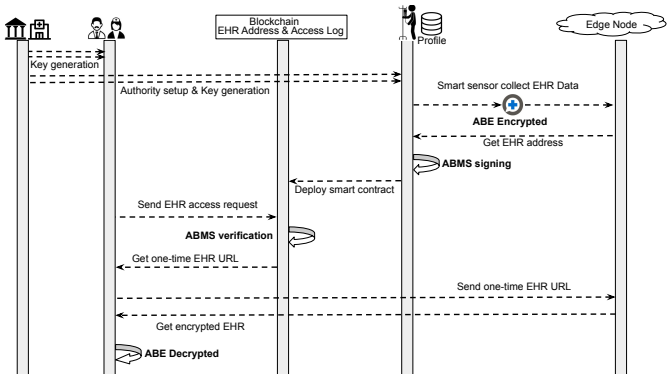

Fig. 4: Workflow of Attribute-based Multi-Signature and Encryption for EHR management.

III. Prototype and Evaluation

A. Prototype

We prototyped the proposed EHR management system and conducted a series of experiments to evaluate its performance. The system consists of two primary modules that interact seamlessly: the ABMS module and the blockchain module.

A.1 ABMS Module

ABMS performs the functionalities of key generation, and signing and verification of attribute-based multi-signatures of the patients. These functionalities are programmed by using Hyperledger Ursa [10], a cryptographic library for blockchain-based applications. Hyperledger Ursa is built on the Rust language and provides APIs for various digital signature schemes.

1) Phase 1 – Initialization: Phase 1 initializes the instances of the participants which include multiple authorities, patients, and doctors. The patient Annie Foster, for instance, has three attributes, which are `patient_id` (value: `0003231`) issued by the hospital, `driver_license` (value: `9907184`) issued by the Department of Motor Vehicles (DMV), and `insurance_id` (value: `1EG4-TE5-MK72`) issued by her health insurance company.

2) Phase 2 – Multi-Authority Key Generation: In Phase 2, each attribute authority generates a key pair for each attribute it issues. The BLS signature scheme [18] was used to build the key-pair generator, which generates the signing key for the data owner and the verification key for the data user, as follows:

```
let generator = Generator::new().unwrap();
let sign_key = SignKey::new().unwrap();
```

```
let verify_key = VerKey::new(&generator,
                 &sign_key).unwrap();
```

3) Phase 3 – Signing ABMS Signatures by Data Owner: In this phase, the data owners use the signing keys to sign their hashed attribute values, and the resulting signatures are saved in data owners' profiles on the blockchain (Section IV-A.2). Using the BLS signature scheme [18], each signature consists of three elements on an elliptic curve. For instance, as shown in Fig. 5, the hashed attribute value of `0003231` from `patient_id` was signed into a combination of three points on an elliptic curve. Our experiments show that the average running time for the signing phase is around 32 ms.

```
(1, 0C7392EE344DFEC0218728C8A2B1D852EF6A82B4CA7521CE53D8E0968FA26C54)
(1, 1573E7399EB1491F1D3A69C88C603F15B29989723087C3E490B064F274948A93)
(2, 095E45DDF417D05FB10933FFC63D474548B7FFFF7888802F07FFFFFF7D07A8A8)
```

Fig. 5: Example of one ABMS signature.

4) Phase 4 – Verifying ABMS Signatures by Data User: Eventually, a data user verifies each ABMS signature retrieved from the blockchain (Section IV-A.2). The verification function takes a signature, a hashed attribute value, a verification key and the corresponding generator as inputs, and utilizes BLS bilinear pairing [19] to verify the signature:

```
let result = Bls::verify($signature,
             patient_id.as_slice(),
             $verify_key, $generator)
             .unwrap();
```

In our experiments, the average running time for verifying each signature is around 243 ms. After that, the data user can authenticate the data owner anonymously by applying the multi-signature threshold mechanism, where the number of verified signatures is compared against the threshold.

A.2 Blockchain Module

The blockchain module is developed on the Hyperledger Fabric and deployed and executed on the Hyperledger Composer Playground [9]. The blockchain module records patients' profiles of GIDs, first and last names, signed ABMS signatures, and one-time self-destructing url addresses [20] for EHR data stored on the edge node. We use `https://1ty.me/` [20] to encode the addresses of EHR data stored on the edge node. Once an `1ty.me` address is accessed, its url link becomes invalid and cannot be used to access the same EHR data again.

To evaluate the access control mechanism between patients and healthcare providers, we conducted the following experiments. When a healthcare provider wants to access a specific patient's EHR data from the off-chain storage, he/she needs to first authenticate the patient by verifying patient's ABMS signatures (Section IV-A.1(4)) against the threshold via a smart contract. Upon executing the smart contract successfully, it returns a one-time self-destructing url address of patient's EHR data stored in the edge node. At the same time, the blockchain will record this access event as a transaction, which includes the event ID and the timestamp as shown in Fig. 6.

Fig. 6: Result of executing the smart contract.

B. Performance Evaluation

To evaluate the performance of the ABMS module, we conducted experiments to analyze the effect of varying the length of attribute values as well as the number of attributes.

1) Varying the length of attributes: We increased the length of attribute from 10 to 100, 1,000, and 10,000 characters, and the results showed that both signing and verification time are independent of the attribute length, and the time remains around 32 ms and 243 ms, respectively, for each attribute.

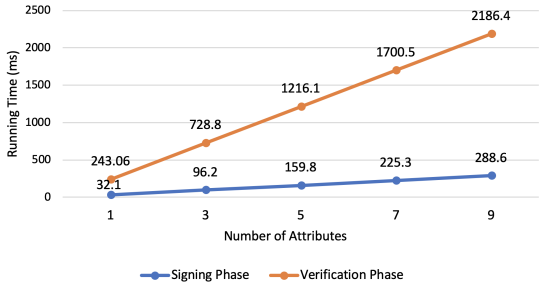

Fig. 7: Run time of Ursa BLS vs. the number of attributes.

2) Varying the number of attributes: We also increased the number of attributes of a patient from 1 to 3, 5, 7 and 9, and measured the running time of the signature signing and verification phases. The total running time showed a linear growth with the increasing the number of attributes (Fig. 7). Compared to the signing phase, the verification phase takes more time because it requires computing two pairings on the elliptic curve [18].

IV. Summary

In this paper, we proposed a hybrid architecture of blockchain and edge nodes, by utilizing the ABMS scheme to authenticate user's signatures without revealing sensitive information and the ABE mechanism to encrypt EHR data which is stored on the edge node. We developed the blockchain module on Hyperledger Fabric platform and the ABMS module on Hyperledger Ursa library. To evaluate the system performance, we designed and conducted experiments for ABMS module. For ABMS module, we measured the signing and verifying time under different settings. For the blockchain module, we experimented with the authentication events and access activities, which were logged as transactions in the Hyperledger Fabric blockchain.

REFERENCES

[1] A. Dubovitskaya, Z. Xu, S. Ryu, M. Schumacher, and F. Wang, "Secure and trustable electronic medical records sharing using blockchain," in *AMIA Annual Symposium Proceedings*, vol. 2017. American Medical Informatics Association, 2017, p. 650.

[2] H. K. Maji, M. Prabhakaran, and M. Rosulek, "Attribute-based signatures: Achieving attribute-privacy and collusion-resistance." *IACR Cryptology ePrint Archive*, vol. 2008, p. 328, 2008.

[3] A. Sahai and B. Waters, "Fuzzy identity-based encryption," in *Annual International Conference on the Theory and Applications of Cryptographic Techniques*. Springer, 2005, pp. 457–473.

[4] A. Narayanan, J. Bonneau, E. Felten, A. Miller, and S. Goldfeder, *Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction*. Princeton University Press, 2016.

[5] A. Ekblaw, A. Azaria, J. D. Halamka, and A. Lippman, "A case study for blockchain in healthcare:"medrec" prototype for electronic health records and medical research data," in *Proceedings of IEEE Open & Big Data Conference*, vol. 13, 2016, p. 13.

[6] J. D. Halamka and A. Ekblaw, "The potential for blockchain to transform electronic health records," *Harvard Business Review*, vol. 3, no. 3, pp. 2–5, 2017.

[7] H. Guo, W. Li, M. Nejad, and C.-C. Shen, "Access control for electronic health records with hybrid blockchain-edge architecture," *in Proceedings of IEEE International Conference on Blockchain, Atlanta, USA*, July 2019.

[8] K. N. Griggs, O. Ossipova, C. P. Kohlios, A. N. Baccarini, E. A. Howson, and T. Hayajneh, "Healthcare blockchain system using smart contracts for secure automated remote patient monitoring," *Journal of medical systems*, vol. 42, no. 7, p. 130, 2018.

[9] Hyperledger Fabric, https://www.hyperledger.org/projects/fabric.

[10] Hyperledger Ursa, https://www.hyperledger.org/projects/ursa.

[11] A. Lewko and B. Waters, "Decentralizing attribute-based encryption," in *Annual international conference on the theory and applications of cryptographic techniques*. Springer, 2011, pp. 568–588.

[12] J. Li, M. H. Au, W. Susilo, D. Xie, and K. Ren, "Attribute-based signature and its applications," in *Proceedings of the 5th ACM Symposium on Information, Computer and Communications Security*. ACM, 2010, pp. 60–69.

[13] S. F. Shahandashti and R. Safavi-Naini, "Threshold attribute-based signatures and their application to anonymous credential systems," in *International Conference on Cryptology in Africa*. Springer, 2009, pp. 198–216.

[14] G. Zyskind, O. Nathan, and A. Pentland, "Decentralizing privacy: Using blockchain to protect personal data," in *2015 IEEE Security and Privacy Workshops*. IEEE, 2015, pp. 180–184.

[15] M. Baza, N. Lasla, M. Mahmoud, and M. Abdallah, "B-ride: Ride sharing with privacy-preservation, trust and fair payment atop public blockchain," *arXiv preprint arXiv:1906.09968*, 2019.

[16] M. Baza, M. Nabil, N. Lasla, K. Fidan, M. Mahmoud, and M. Abdallah, "Blockchain-based firmware update scheme tailored for autonomous vehicles," *Proc. of the IEEE Wireless Communications and Networking Conference (WCNC), Marrakech, Morocco*, April 2019.

[17] H. Guo, E. Meamari, and C.-C. Shen, "Multi-authority attribute-based access control with smart contract," *In Proceedings of 2019 International Conference on Blockchain Technology (ICBCT 2019). ACM, 6 pages. https://doi.org/10.1145/3320154. 3320164*.

[18] D. Boneh, B. Lynn, and H. Shacham, "Short signatures from the weil pairing," in *International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT), Gold Coast, Australia*. Springer, December 2001, pp. 514–532.

[19] D. Boneh, C. Gentry, B. Lynn, H. Shacham *et al.*, "A survey of two signature aggregation techniques," *RSA cryptobytes*, vol. 6, no. 2, pp. 1–10, 2003.

[20] 1ty.me, https://1ty.me/.