A Privacy-Preserving and Fully Decentralized Storage and Sharing System on Blockchain

Gen Li
University of Tokyo
ligen@satolab.itc.u-tokyo.ac.jp

Hiroyuki Sato
University of Tokyo
schuko@satolab.itc.u-tokyo.ac.jp

***Abstract*—With the blockchain receiving extensive attention in recent years, many storage schemes based on the blockchain have been proposed as alternative means of cloud storage for data outsourcing. However, the conventional access control methods in the current sharing schemes require either individual permission granting via symmetric keys or a trusted central attribute authority for ciphertext-policy attribute-based encryption (CP-ABE). Moreover, due to the transparency of the blockchain, the data query logs are recorded in public ledgers. Nevertheless, the problem of privacy protection for data consumers has not been properly addressed.**

In this paper, we propose a fully decentralized data storage and sharing system on a blockchain by using multi-authority CP-ABE and decentralized multi-authority attribute-based signatures (DMA-ABSs). In our system, every party can securely store and share its data with a set of individuals satisfying a policy with no need to grant separate permissions individually. Additionally, the data owners can fully control their data, know how their data are accessed due to the nature of the blockchain and have the ability to opt-out at any time. The public ledger of the blockchain provides immutable logs of data address pointers, access policies, attribute public keys and data queries. In addition, data consumers' attributes are publicly verifiable through the DMA-ABS scheme without revealing more private information. Finally, the combination of the multi-authority CP-ABE with the blockchain guarantees the integrity, confidentiality, and accessibility of the data without the need for trusted third parties, such as a central authority or a data center.

***Index Terms*—privacy, blockchain, access control, file sharing, CP-ABE, attribute-based signature**

I. INTRODUCTION

In recent years, rapidly developing cloud computing has promoted increasingly more organizations and individuals to outsource their data for storage and sharing due to its low cost and high accessibility. However, this datacenter-based strategy heavily depends on trusted third parties who are sometimes compromised, and therefore there are some risks such as the loss of data and failures in service availability. In addition, once users upload their data to the cloud, they cannot fully control the access to their data.

To address the problem above, blockchain technology has been employed to build auditable and tamper-free storage solutions because of its transparency and immutability [1], [2]. Nevertheless, the efficiency of data sharing in current blockchain-based solutions is limited, since the data owners have to grant permission individually. In many applications, to improve the efficiency, we often need to grant permission for a set of recipients according to a policy without knowing the exact identities of all people who are allowed to access the data [3].

CP-ABE is a well-developed and widely used cryptographic primitive for access control management that allows data owners to encrypt their data and to determine the corresponding access policies for those who can decrypt the data. Therefore, CP-ABE has been applied for blockchain storage to improve the sharing efficiency [4], [5]. However, in these solutions, a trusted third party, central attribute authority, is still necessary. Multi-authority CP-ABE, which is a decentralized CP-ABE approach in which anyone can create an attribute and authorize different users, can be used to further decentralize blockchain storage.

On the other hand, since we are concerned with the users' privacy, the decentralized multi-authority attribute-based signature (DMA-ABS) has been introduced for anonymous authentication before accessing data. Unlike the traditional signature scheme that would reveal the real signers, the attribute-based signature (ABS) releases no more information other than the fact that the signer keeps a set of attributes that satisfy the given predicate (policy) [6].

In this paper, we combine multi-authority CP-ABE and DMA-ABSs with a blockchain to design a fully decentralized storage and sharing system with auditable access data in which no trusted third parties are required and users' privacy is well preserved. The rest of the paper is organized as follows. Section II provides an overview of the related work. Section III describes the concept and details of the system. Section IV analyzes the security. Section V concludes the paper.

II. RELATED WORK

A blockchain is a continuously growing sequence of blocks that are linked through hash values that contain batches of verified transactions with timestamps. Typically, a blockchain is managed by a peer-to-peer network, and its data are replicated and stored at individual nodes in the network. Thus, no trust in third parties is required in this decentralized, distributed, and public system, which also eliminates the risks of datacenter-based solutions that we mentioned before.

Due to these properties, some storage solutions have been proposed based on the blockchain. Zyskind et al. [1] design a decentralized personal data management system in which the data owner can store and control the access of their data.

978-1-7281-2607-4/19/$31.00 ©2019 IEEE
DOI 10.1109/COMPSAC.2019.10289

The authors use the outputs in transactions to transmit data and their address pointers (the hash values of the data) from the owners to the blockchain. The data are then routed to the off-chain storage while only the pointers are left in the public ledger. This system uses an access policy and digital signatures to ensure the ownership of the data and that they are accessed by properly authenticated consumers. Additionally, the introduction of the blockchain solves the issue of data transparency and auditability, since the data access records will be left in the public ledger. However, the access control in the system is limited because every data consumer needs to be authorized separately. The reason is that the data are encrypted using a symmetric cipher that requires a shared secret key between an owner and a consumer, which raises key and policy management challenges. In addition, Do and Ng [2] introduce a private keyword search into blockchain storage that allows authorized data consumers to search for keywords of encrypted data. However, individual permission granting is still required.

CP-ABE is an advanced access control method that was proposed by Bethencourt et al. [3] for efficient outsourced data sharing. In CP-ABE, the ciphertext determines the access policy structure while the keys are generated according to the attributes (or credentials). An access policy is a boolean formula over some attributes. Only the persons whose attributes match the policy of the data can decrypt the ciphertext.

To address the efficiency problem of data sharing in the blockchain, Yuan et al. [4] combine CP-ABE with blockchain storage. Even though this scheme enables data sharing depending on attributes, it heavily relies on a central authority for the attribute assignment. Additionally, Jemel and Serhrouchni [5] introduce the valid time to CP-ABE-based blockchain storage where revocation would be unnecessary. Nevertheless, this system still requires trust in a third party for prior distributed attributes. In a blockchain based decentralized storage system, we would like to avoid using a central attribute authority that the normal CP-ABE needs and placing absolute trust in it.

Multi-Authority CP-ABE is a system that was designed by Lewko and Waters [7] to decentralize attribute-based encryption. In the system, anyone can create an attribute by generating a public key and authorize consumers by issuing "customized" private keys. Data consumers can simply combine private keys from different authorities to match the attribute policy of the ciphertext for decryption. This cipher works correctly if all users agree on a set of global parameters. Additionally, collusion is prevented by tying every users' global ID to their private keys to introduce distinct noise. The noise will not be canceled out if someone attempts to combine the private keys from different users.

Rahulamathavan et al. [8] propose an Internet of Things (IoT) application based on decentralized multi-authority CP-ABE and the blockchain. In the application, sensor data are aggregated and encrypted at cluster heads and transmitted to the blockchain through transactions. Although this application supports the data integrity, confidentiality and non-repudiation without a centralized server, there are still two main problems. First, all data are stored in the blockchain public ledger that will be duplicated by nodes. As a consequence, the maintenance will be costly. Furthermore, transactions are verified by users who have the correct attributes to access the data rather than being publicly verified. This dramatically reduces the number of miners and can possibly result in malicious collusion.

The attribute-based signature (ABS) is a new and versatile digital signature in which a signer can obtain a set of attributes from authorities and then sign a message along with a predicate. The signature hides both the identity of its signer and the possessed attributes while it only claims that the signer's attributes satisfy a given predicate, which preserves the signers' privacy and anonymity. Early ABS schemes rely on a central authority for attribute distribution in systems. Okamoto and Takashima [9] present a decentralized multi-authority attribute-based signature (DMA-ABS) with no central authorities. Similar to decentralized multi-authority ABE in [7], a signer can obtain private keys from authorities and generate a signature to a message with a predicate by using the private keys of the associated attributes.

III. System Model

In this section, we describe the architecture of the system to show how blockchain technology, multi-authority CP-ABE and DMA-ABS are combined to build a fully decentralized storage system.

A. Overview

Like other blockchain based storage solutions [1], [2], there are the three following roles for users.

1) **Data Owners**: users who own data and want to outsource them for storage and sharing.
2) **Data Consumers**: users who are interested in some data and want to access them.
3) **Blockchain Nodes**: entities that maintain the blockchain and a discrete hash table for storage.
   On the basis of these roles, we introduce a new role for attribute distribution:
4) **Attribute Authority**: users who create attributes and provide authorization.

There are three kinds of transactions that are allowed in the scheme: $TX_{ATTRIBUTE}$, which is used for attribute distribution; TX_{DATA}, which is used for data storage and the access policy assignment; and TX_{ACCESS}, which is used for retrieval and verification.

B. Multi-Authority CP-ABE and DMA-ABS

The data in the system are encrypted using multi-authority CP-ABE whereas consumers are anonymously verified via DMA-ABS. The cipher is briefly shown as follows. We refer the interested reader to [7] and [9] for more details and the proof.

Global Setup$(\lambda) \rightarrow GP$. This is the initializing algorithm that takes a security parameter λ as the input and generates global parameters for the scheme.

Authority Setup$(GP) \rightarrow (PK_i, SK_i)$. To create an attribute i, this algorithm uses the commonly agreed parameters GP to produce the pair of public PK_i and secret master keys SK_i.

Encryption$(M, (A, \rho), GP, \{PK_i\}) \rightarrow CT$. The algorithm encrypts a message M according to the access structure (A, ρ) and public keys of the relevant attributes to output the ciphertext CT. Additionally, an access structure (A, ρ) defines the authorized set of users who are allowed to access the data where A is an access matrix and ρ maps the rows of A to the attributes.

Key Generation$(GID, i, SK_i, GP) \rightarrow K_{i,GID}$. The private key $K_{i,GID}$ of an attribute i for a user with the identity GID is issued by taking the user's global identity GID, the global parameter and the authority's secret master key SK_i as inputs. Like the Bitcoin system, we that assume every user would have one or more pairs of asymmetric cipher keys while the hash value of public keys are transaction addresses. Thus, we use the public keys of users as global identities.

Decryption$(CT, \{K_{i,GID}\}, GP) \rightarrow M$. If a decryptor holds the private keys of a collection of attributes $\{K_{i,GID}\}$ that satisfies the access structure, then message M can be recovered from the ciphertext CT with the global parameters GP.

ABS.Signing$(\{K_{i,GID}\}, M, (A, \rho), \{PK_i\}, GP) \rightarrow \sigma$. The signature σ of a message M with a predicate (A, ρ) can be generated when the combination of involved attributes satisfies the predicate(A, ρ).

ABS.Verification$(\sigma, M, (A, \rho), \{PK_i\}, GP) \rightarrow \{0, 1\}$. A signature σ on a message M with the predicate (A, ρ) is verified to be ***True***(1) or **False** (0) using the public keys of the relevant attributes $\{PK_i\}$.

A correct multi-authority CP-ABE system ensures that if whenever the global parameter GP is initialized by the *Global Setup* , the ciphertext CT is generated from the message M via the *Encryption* algorithm with the access structure (A, ρ) , $\{K_{i,GID}\}$ is a collection of attributes that are issued through the Key Generation algorithm for the same global identities GID, and their associated attributes satisfy the access structure (A, ρ), then the message M can be recovered through the algorithm *Decryption*$(CT, \{K_{i,GID}\}, GP) = M$. Meanwhile, a correct DMA-ABS scheme ensures that if whenever GP is initialized by the *Global Setup*, the signature σ on the message M with the predicate (A, ρ) is generated through the *ABS.Signing* algorithm using $\{K_{i,GID}\}$, which is issued through the *Key Generation* algorithm for the same global identity GID and for a collection of attributes that satisfies the predicate (A, ρ), the signature is verified to be **True** (*ABS.Verification*$(\sigma, M, (A, \rho), \{PK_i\}, GP) = \{0, 1\}$) using the Verification algorithm [3], [7], [9].

C. Blockchain Protocol

1) Attribute distribution

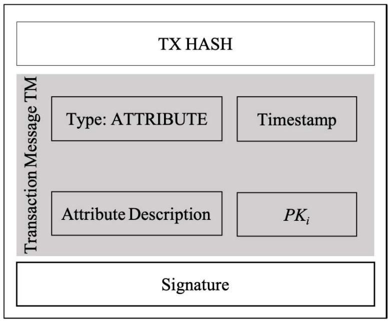

Fig. 1: The structure of transaction attribute $TX_{ATTRIBUTE}$

To create an new attribute i, any user can be the authority AA_i and generate a signed $TX_{ATTRIBUTE}$ illustrated in Fig 1 stating the attribute description (e.g.,the attribute name and the condition to be authorized) and its public key PK_i after running the *Global Setup* and *Authority Setup*. In this paper, TM is used to refer the parts of a transaction excluding the hash value and the signature. Then these authorities are able to authorize users by issuing private keys to them. **Algorithm 1** illustrates the process of the authority setup and authorization.

Algorithm 1 Authorization

```
procedure AUTHORIZATION($\lambda, i, GID$)
    $AA_i$ and $U_{GID}$ form a secure channel
    $AA_i$ executes :
        $GP \leftarrow$ Global Setup($\lambda$)
        $(PK_i, SK_i) \leftarrow$ Authority Setup($GP$)
        $K_{i,GID} \leftarrow$ Key Generation($GID, i, SK_i, GP$)
    $AA_i$ sends $K_{i,GID}$ to $U_{GID}$
     to authorize $U_{GID}$ with attribute $i$
end procedure
```

2) Data Outsourcing

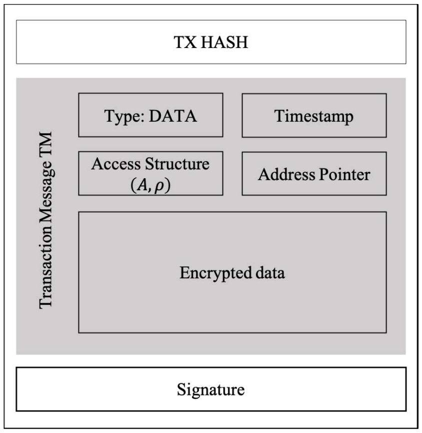

Fig. 2: The structure of transaction data TX_{DATA}

A data owner can select a collection of attributes that they trust and build the access policy over them before encryption. Then the data owner outsources the encrypted data alongwith a normal digital signature (e.g., ECDSA) in a TX_{DATA}. Particularly, if (PK_{sig}, SK_{sig}) is a pair of keys for a correct signature scheme, σ is the signature on the message M that is generated by $\sigma = \textit{Signing}(M, SK_sig)$, and then $\textit{Verification}(\sigma, M, PK_sig) = \boldsymbol{\textit{True}}$. The signature is on the part of the TX_{DATA} excluding the encrypted data; while the integrity of the data is guaranteed by the address pointer. An address pointer is not necessarily a single hash value of data, since some hash scheme(e.g., the Merkle Hash Tree) can be adopted to protect the data integrity. Furthermore, similar to the method of Zyskind et al. [1], the data are transmitted in transactions TX_{DATA} shown in Fig 2 and then routed to the off-chain storage ds while only the remaining parts are left in the public ledger L. The difference in our system is that apart from the address pointers and the data themselves, the access structure (A, ρ) is introduced into transactions, where ρ maps the rows in the access matrix A with the $TX_{ATTRIBUTE}$ hash values of the corresponding attributes. **Algorithm 2** is executed to process the data transactions.

Algorithm 2 Outsourcing Data

```
procedure PROCESSDATATRANSACTION($TM, \sigma, PK_{sig}$)
    $TXType, CT, AP, TS, (A, \rho) \leftarrow \textit{Parse}(TM)$
    if $Verification(TM, \sigma, PK_{sig}) = True$ then
        if $AP = \mathcal{H}(CT)$ then
            $TM' \leftarrow TXType||TS||(A, \rho)||AP$
            $L[\mathcal{H}(TM'||\sigma)] \leftarrow TM'||\sigma$
            $ds[AP] \leftarrow CT$
        end if
    end if
end procedure
```

3) *Access Query*

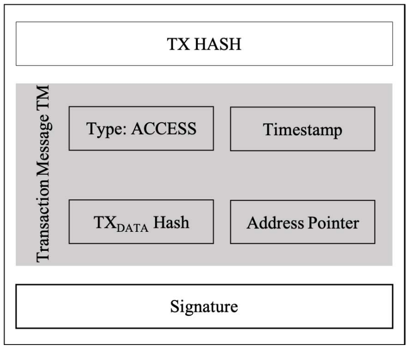

Fig. 3: The structure of transaction access TX_{ACCESS}

When a data consumer attempts to access outsourced data , she needs to send a TX_{ACCESS} whose structure is demonstrated in Fig 3 to the blockchain to provide the TX_{DATA} hash value of the desired data and their address pointers with the ABS, proving that she has the required attributes that satisfy the access structure. Then the blockchain verifies the transaction based on the access policy and signature. If a consumer is successfully verified, the blockchain returns the encrypted data to her. Finally, the plaintext can be recovered by combining the private keys of the relevant attributes and running the Decrypt algorithm. **Algorithm 3** shows how access queries are processed. In addition, data owners can simply operate on their data by providing the signature.

Algorithm 3 Access Query

```
procedure PROCESSACCESSQUERY($TM, \sigma$)
    $TX_{DATA}Hash, AP_{Query}, \leftarrow \textit{Parse}(TM)$
    if $L[TX_{DATA}Hash] \neq \emptyset$ then
        $(A, \rho), AP \leftarrow \textit{Parse}(L[TX_{DATA}Hash])$
        if $AP = AP_{Query}$ then
            while $L[\rho_i] \neq \emptyset$ do
                $\{PK_i\} \leftarrow \textit{Parse}(L[\rho_i])$
            end while
            if $ABS.Verification(TM, \{PK_i\}, (A, \rho), \sigma) = True$ then
        return ds[AP]
            end if
        end if
    end if
end procedure
```

We use an example here to illustrate the workflow of the system in Fig 4. First, Bob and Carol create the attributes ***Prof*** and ***Math*** respectively and send $TX_{ATTRIBUTE}$ to the blockchain nodes. Then Bob authorizes ***User 1*** and ***User 2*** with the private keys $K_{Prof,1}$and $K_{Prof,2}$ respectively, while Carol similarly authorizes ***User 2***, ***User 8***, and ***User 9***. A data owner, Alice, can outsource the document and share it with people who are both ***Prof*** and ***Math*** by encrypting it using the associated public keys that are gathered from the blockchain public ledger. Finally, ***User 2*** who is both ***Prof*** and ***Math*** can query the data by issuing a TX_{ACCESS} in which the required attributes are proved via the ABS. The blockchain verifies the signature to evaluate whether it satisfies the access policy before returning the encrypted data CT.

IV. SECURITY ANALYSIS

We combine the multi-authority CP-ABE and the DMA-ABS with the blockchain for storage and sharing, which solves both the ownership problem of the CP-ABE-based cloud storage and the problem of multiple user authorization in the blockchain based system. In this system, the blockchain provides tamper-free records for the data address pointers, access policies, attribute public keys and data queries; the DMA-ABS ensures that consumers are anonymously verified, and the CP-ABE enables group sharing and protects the data confidentiality.

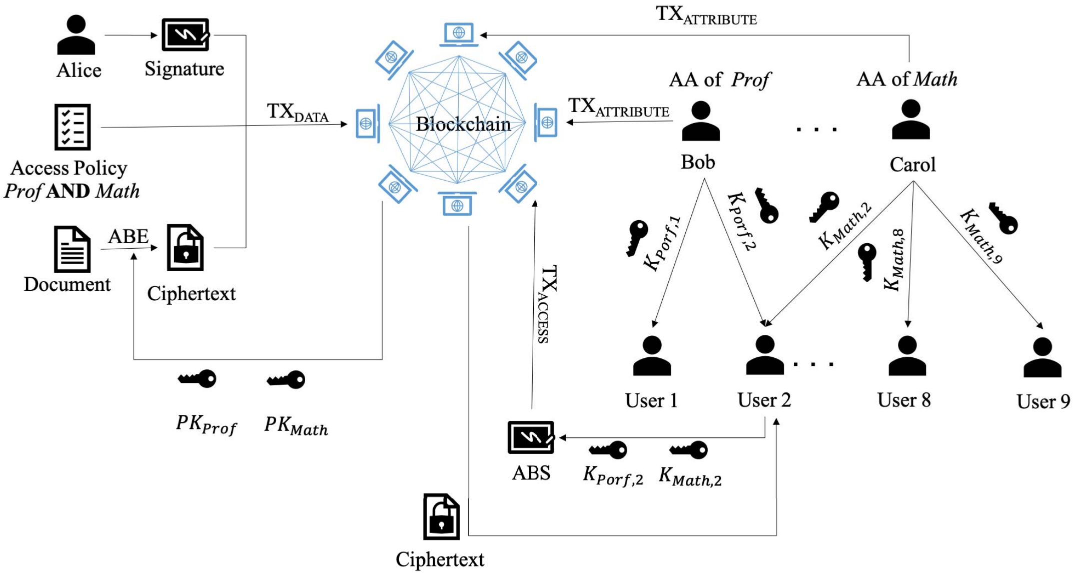

Fig. 4: An example of access control architecture of the system

Similar to cryptocurrency, even though all users are allowed to create their attributes as authorities, only the authorities with good behavior and well-maintained attributes will survive the competition and be trusted by more users if appropriate incentive schemes are designed. Furthermore, attribute authorities who also satisfy the conditions of their own attributes are more likely to be trusted. Thus, our conditional assumptions are as follows.

1) There have been various authorities with well-maintained attributes that are trustworthy. Particularly, the authorities also prove that they meet the conditions of the attributes that they created. An adversary has difficulty being authorized by such authorities if he cannot satisfy the attribute conditions.
2) Discrete hash table (DHT) storage is untrusted.
3) To secure data, owners tend to encrypt over these well-maintained attributes; meanwhile consumers would also like to be authorized with these attributes to access data.

Property 1: Users' identities and the data they queried are protected. In addition, the privacy regarding users' attributes is also preserved. Only the authorities know their own attribute distributions.

(Proof): Data owners and attribute authorities can generate new identities for data outsourcing or attribute creation to break the linkability. Additionally, data consumers are anonymously authenticated using the ABSs that reveals no information regarding users' identities and exact attributes except the fact that the access structures are satisfied. Therefore, little attribute information can be used for an adversary to de-anonymize users.

Property 2: Data owners fully control their data, and only users with the required attributes are allowed to access the data. Furthermore, the data integrity is also preserved.

(Proof): Because the data are encrypted over associated attributes and stored in a discrete hash table, an adversary can obtain no information by controlling the DHT nodes if she does not possess the necessary secret keys for decryption. In addition, the hash values of the data are permanently saved in the tamper-free public ledger, and the hash-based verification ensures that data in the DHT nodes are difficult to be changed when some hash schemes for data integrity are adopted, such as the Merkle Hash Tree.

V. Conclusion

This paper has presented a novel decentralized storage scheme by using multi-authority CP-ABE, DMA-ABS and blockchain technology. This system provides efficient access control and secure data storage with the privacy preserved and no central entities required. We review the main challenges of data outsourcing and then illustrate the architecture and how current problems can be addressed. In our solution, when data owners attempt to change the access policies of their data, re-encryption and new policy assignment are necessary. Therefore, as a future work, we would like to address the problems in revocation problems.

Acknowledgment

This research was partially supported by Japan Society for the Promotion of Science Grants-in-Aid for Scientific Research (JSPS KAKENHI) No. 19K11958.

References

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

[2] H. G. Do and W. K. Ng, "Blockchain-based system for secure data storage with private keyword search," in *2017 IEEE World Congress on Services (SERVICES)*. IEEE, 2017, pp. 90–93.

[3] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *2007 IEEE symposium on security and privacy (SP'07)*. IEEE, 2007, pp. 321–334.

[4] C. Yuan, M. Xu, X. Si, and B. Li, "Blockchain with accountable cp-abe: how to effectively protect the electronic documents," in *2017 IEEE 23rd international conference on parallel and distributed systems (ICPADS)*. IEEE, 2017, pp. 800–803.

[5] M. Jemel and A. Serhrouchni, "Decentralized access control mechanism with temporal dimension based on blockchain," in *2017 IEEE 14th International Conference on e-Business Engineering (ICEBE)*. IEEE, 2017, pp. 177–182.

[6] H. K. Maji, M. Prabhakaran, and M. Rosulek, "Attribute-based signatures," in *Cryptographers' track at the RSA conference*. Springer, 2011, pp. 376–392.

[7] 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.

[8] Y. Rahulamathavan, R. C.-W. Phan, M. Rajarajan, S. Misra, and A. Kondoz, "Privacy-preserving blockchain based iot ecosystem using attribute-based encryption," in *2017 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*. IEEE, 2017, pp. 1–6.

[9] T. Okamoto and K. Takashima, "Decentralized attribute-based signatures," in *International Workshop on Public Key Cryptography*. Springer, 2013, pp. 125–142.