Privacy-preserving Blockchain based IoT Ecosystem using Attribute-based Encryption

Yogachandran Rahulamathavan, Raphael C.-W Phan, Muttukrishnan Rajarajan, Sudip Misra, and Ahmet Kondoz

***Abstract*—The Internet of Things (IoT) has penetrated deeply into our lives and the number of IoT devices per person is expected to increase substantially over the next few years. Due to the characteristics of IoT devices (i.e., low power and low battery), usage of these devices in critical applications requires sophisticated security measures. Researchers from academia and industry now increasingly exploit the concept of blockchains to achieve security in IoT applications. The basic idea of the blockchain is that the data generated by users or devices in the past are verified for correctness and cannot be tampered once it is updated on the blockchain. Even though the blockchain supports integrity and non-repudiation to some extent, confidentiality and privacy of the data or the devices are not preserved. The content of the data can be seen by anyone in the network for verification and mining purposes. In order to address these privacy issues, we propose a new privacy-preserving blockchain architecture for IoT applications based on attribute-based encryption (ABE) techniques. Security, privacy, and numerical analyses are presented to validate the proposed model.**

***Index Terms*—Privacy, security, Internet of Things, Blockchain, attribute based encryption.**

I. Introduction

Various modern-day applications such as smart city, smart grid, eHealth, and Industry 4.0 are relying on millions of connected digital devices distributed across people, homes, roads, and communities etc [1]. These sensors measure data such as heart rate, blood pressure, weather conditions, location, traffic conditions, vehicular speed etc based on their characteristics and upload the measured data to their owner/master periodically. This data enables the master to develop intelligent algorithms to monitor the subjects in real time. For instance, if it is a healthcare application then doctors can monitor the patients' health condition dynamically and prescribe medications immediately to avoid any delay and complications.

Even though the IoT technology seems to be exciting and solving various problems in real time, achieving security and privacy in IoT is challenging due to its characteristics i.e., low processing power, distributed nature, and lack of standardization [2]. In order to tackle this shortcoming, researchers from industry and academia are focussing on blockchain fundamentals and customising blockchain based cryptocurrency models for IoT applications.

This work was supported by the UK-India Education Research Initiative (UKIERI). Ref No. UGC-UKIERI-2016-17-019.

Y. Rahulamathavan and A. Kondoz are with the Institute for Digital Technologies, Loughborough University London, London, U.K. (e-mails: y.rahulamathavan@lboro.ac.uk and a.kondoz@lboro.ac.uk).

R. C.-W Phan is with MMU University, Malaysia, US. (e-mail: raphael@mmu.edu.my).

M. Rajarajan is with the Information Security Group, School of Engineering and Mathematical Sciences, City University London, EC1V 0HB, London, U.K. (e-mail: R.Muttukrishnan@city.ac.uk).

S. Misra is with the Department of Computer Science & Engineering Indian Institute of Technology, Kharagpur (e-mail: smisra@sit.iitkgp.ernet.in).

The work in [2] proposed an optimised blockchain technology suitable for IoT and developed an end-to-end solution for smart home application. A hierarchical structure has been proposed in [2] to optimize resource consumption and increase network scalability. The work in [3] focuses on edge and fog computing scenarios where the IoT devices located in the edge of 4G/5G networks can be used to process the sensor data in a distributed manner using blockchain.

The recent work of [4] exploits blockchain technology in IoT to avoid the need of a centralised server. Instead of storing IoT sensor data in centralised servers, the blockchain technology supports the devices and users to maintain a distributed database where sensor data can be managed by individuals similar to crypto currencies [4]. There are few more recent works proposing similar techniques for different applications [5], [6]. However, these techniques do not protect the privacy of data in the transactions. The main reason for this is that all the techniques rely on symmetric key encryption for data encryption. If the data is encrypted by symmetric key schemes such as AES, the key must be shared together with the data to enable the miners of the blockchain to verify the content and update the blockchain. This means privacy and confidentiality of the data generated and shared by the IoT network is not protected [7].

In this paper, we exploit the state-of-the-art attribute-based encryption (ABE) technique to address the privacy and confidentiality of the data shared in blockchain based IoT ecosystems. ABE has been known for its simplicity where a single encryption provides both confidentiality and access control and has been identified as a potential technology for data sharing in decentralised networks [8]. To the best of our knowledge, this is the first work that restructures the blockchain protocol to absorb ABE and provide end-to-end privacy-preserving blockchain based IoT ecosystem.

II. System model

Let us describe the IoT network model considered in this paper. Similar to the work in [2], we also consider a hierarchical approach where there will be a *cluster head* for a given set of IoT sensors. The cluster head is assumed to be more powerful than IoT devices and performs computationally intensive operations such as data processing and encryption. The data recorded by IoT devices are transmitted to the cluster head for processing and transmission. As shown in Figure 1, there are a number of *blockchain miners* who verify transactions and contribute to the blockchain. These miners could be service providers or even cluster heads. In order to provide ABE there will be a number of *attribute authorities* (AAs) part of this network. Let us briefly define the concept of ABE and how it will be used for blockchain in the next subsection followed by the blockchain transaction architecture.

A. Attribute based encryption scheme

ABE supports both confidentiality and access control via single encryption [8]. There are four parties involved in ABE, namely cluster head (data owners), blockchain miners, attribute authorities (AA) and distributed ledger (or blockchain with blocks of transactions). The cluster head aggregates or processes the data from sensors and encrypts them before the transaction. The cluster head encrypts the data in such a way that the transactions can be seen and verified by particular miners who have the *right attributes*. In healthcare scenarios, for example, the cluster head may define a miner policy such as "DOCTORS" or "NURSES" to its encryption. Hence the miner who has "DOCTORS" attribute or "NURSES" attribute can decrypt and verify the transactions. Moreover, once these transactions are appended in the blockchain (i.e., distributed ledger system), only users who have these "DOCTOR" or "NURSES" attributes can be able to use the data. This will allow the data owner to control the data privacy through fine-grained access control.

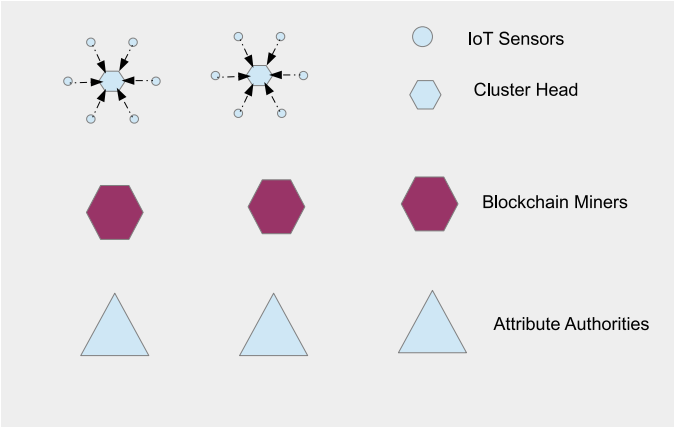


Figure 1: The system model comprises IoT sensors which are connected to corresponding cluster head, blockchain miners and attribute authorities.

It is the AAs' responsibility to verify and issue credentials for different miners and users based on their attributes. There are many centralised and decentralised ABE schemes that have been proposed in the literature [9]–[12]. In this paper, we use a decentralised ABE scheme where more than one AA issues credentials for miners and users. This will allow different sets of attributes to be monitored by different AAs to avoid any single point-of-failure. The decentralised ABE contains the five protocols namely Setup, AA Setup, Key Issuing, Encryption and Decryption. Let us briefly explain the functionalities of each protocol.

Setup: This protocol takes a predefined security parameter as input and outputs the system parameters. These system parameters will be used by AAs who join the system.

AA Setup: Each AA uses the system parameters obtained from the Setup to generate public and private keys for the attributes it maintains.

Key Issuing: Miners/User and AA interact via an anonymous key issuing scheme in order to determine a set of attributes belonging to the user. Then the AA generates decryption credentials for those attributes and sends them to the miners/user.

Encryption: The encryption algorithm will be used by the cluster head. The cluster head takes a set of attributes maintained by AAs and the data from sensors as input. Then it outputs the ciphertext of the data which will be appended in the transaction.

Decryption: The decryption algorithm will be used by miners and users of the blockchain. They take the decryption credentials received from AAs and the ciphertext from the transaction or blockchain as input. The decryption will be successful if and only if the miner/user attributes satisfy the access structure of the ciphertext.

Figure 2 shows the variants of ABE schemes considered for this work [8]. ABE schemes are built based on bilinear pairing, secret sharing and Lagrangian interpolation [8]. Let us briefly explain them below:

Bilinear Pairings: Let $\mathbb{G}_1, \mathbb{G}_2$ be two multiplicative groups of prime order q and let g_1 and g_2 be generators of $\mathbb{G}_1$ and $\mathbb{G}_2$, respectively. Let us denote a bilinear map $e : \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T$. The map has the following three properties:

1 Bilinearity: $\forall x \in \mathbb{G}_1, \forall y \in \mathbb{G}_2$, and $a, b \in \mathbb{Z}_q$, there is $\hat{e}(x^a, y^b) = \hat{e}(x, y)^{ab}$.
2 Non-degeneracy: For $\forall x \in \mathbb{G}_1, \forall y \in \mathbb{G}_2$, there is $\hat{e}(x, y) \neq 1$.
3 Computability: $\hat{e}$ is an efficient computation.

Lagrange Interpolation: Shamir's secret sharing uses the Lagrange interpolation technique to obtain the secret from shared-secrets. Suppose that $p(x) \in \mathbb{Z}_p[x]$ is a $(k-1)$ degree polynomial and secret $s = p(0)$. Let us denote $S = \{x_1, x_2, \ldots, x_k\}$ and the Lagrange coefficient for x_i in S as

$$\Delta_{x_i,S}(x) = \prod_{x_j \in S,\ x_j \neq x_i} \frac{x - x_j}{x_i - x_j}.$$

For a given k different number of values $p(x_1)$, $p(x_2)$, ..., $p(x_k)$, the polynomial $p(x)$ can be reconstructed as follows,

$$p(x) = \sum_{x_i \in S} p(x_i) \prod_{x_j \in S,\ x_j \neq x_i} \frac{x - x_j}{x_i - x_j} = \sum_{x_i \in S} p(x_i)\Delta_{x_i,S}(x),$$

hence the secret s can be obtained as:

$$s = p(0) = \sum_{x_i \in S} p(x_i) \prod_{x_j \in S,\ x_j \neq x_i} \frac{0 - x_j}{x_i - x_j}.$$

Now let us assume there are N number of AAs $(A_1, \ldots, A_N)$ and denote the set of attributes for A_k as $\widetilde{A_k} = \{a_{k,1}, \ldots, a_{k,n_k}\}\ \forall k$. Each AA has a value d_k i.e., miner/user must have at least d_k number of attributes of this authority to obtain the private key associated with this AA. Initially, for a given security parameter λ, the Setup algorithm $(\mathcal{S})$ generates the bilinear groups $\mathbb{G}_1$ and $\mathbb{G}_2$ with prime order p i.e., $\{\mathbb{G}_1, \mathbb{G}_2\} \leftarrow \mathcal{GS}(1^\lambda)$. The Authority Setup algorithm $(\mathcal{AS})$ is executed by each AA to randomly generate public keys (PK) and the corresponding private keys (SK). The public-private key pairs for A_k are given as $\{(Y_k, Z_k, [T_{k,1}, \ldots, T_{k,n_k}]), (\alpha_k, \beta_k, [t_{k,1}, \ldots, t_{k,n_k}])\}$.

Let us denote the attribute set belonging to user u as $\widetilde{A_u}$ and the common attribute set between user u and authority k as

$\widetilde{A_u^k}$ i.e., $\widetilde{A_u^k} = \widetilde{A_u} \bigcap \widetilde{A_k}$. The Key Generation ($\mathcal{KG}$) algorithm will be used to issue decryption keys to the user u with a set of attributes $\widetilde{A_u}$.

Denote the set of attributes used to encrypt message m as $\widetilde{A_m}$ and the common attribute set between message m and the authority k as $\widetilde{A_m^k}$ i.e., $\widetilde{A_m} = \{\widetilde{A_m^1}, \ldots, \widetilde{A_m^k}, \ldots, \widetilde{A_m^N}\}$. Denote the index set of authorities involved in the ciphertext of message m as I_c. The encryption algorithm ($\mathcal{E}$) encrypts the message $m \in \mathbb{G}_2$ using an attribute set $\widetilde{A_m}$. To encrypt the message, the message owner randomly generates s and computes the ciphertext as $C = \left[C_1, C_2, C_3, C_{k,j}, \forall a_{k,j} \in \widetilde{A_m^k}\right]$. If the user has decryption keys for the attributes of message m then he can obtain the message m from the ciphertext using the Decryption algorithm ($\mathcal{D}$).

The decentralised ABE scheme in Figure 2 was built on top of the above fundamentals [8], [9]. Miners and users of the blockchain network will follow the protocols in Figure 2. In the numerical analysis section later in this paper, we compare the complexity increase due to the addition of the ABE scheme.

B. Blockchain Architecture

The use of blockchain technology in IoT has three major security advantages: 1) the sensor data generated by the IoT devices will be rigorously verified by a number of miners in the blockchain network for legitimacy before it is accepted. This will mitigate several security attacks including data manipulation attacks by the adversary. 2) once the data is accepted and appended to the blockchain then the data cannot be tampered. 3) there is no central authority nor storage server hence trust of each node will be built by reputation. If any node is malicious and propagating false data it can be identified by miners and the reputation of that node will be damaged. Let us now look at transaction data generated by the cluster head.

1) Blockchain transaction data: Let us consider healthcare applications where patients use various medical IoT devices to measure health parameters such as weight, heart rate, blood pressure, sugar level etc using various sensors in the devices. Doctors will decide on the types of sensors, and how frequent the readings must be taken and uploaded. The patients' smartphone or home router or both could act as a cluster head. During the registration process the cluster head receives a unique identification number. The cluster head exploits public-key cryptography to generate a private and public key pair. The public key will be given to the hospital server where it is stored against the unique identifier. These information (patients unique identifier, public key and types of sensors) can be retrieved by miners and users of the blockchain in the future. The unique identifier (ID) cannot be used by miners or users to identify the patient's privacy sensitive information such as name, address, etc.

Once the initial setup is completed, the cluster head collects the sensor data and generates a transaction to distribute to the peers for validation. As shown in Figure 3, there will be a number of entries in each transaction. The transaction data starts with unique ID, date and time, and sequence number. These are purely used for identification and administrative purposes. Then application types will be appended. The application types i.e., diabetic, cholesterol can be used for easy identification. If the data will be used for research purposes in future, the application type will enable researchers to aggregate the correct type of data.

Then based on the application type, the cluster head will decide on the attributes for encryption. In this healthcare case, the cluster head will choose attributes such as doctors, nurses, hospitals, locations etc and build an access structure. The example shown in Figure 3 has an access structure where miners or users who have obtained credentials for doctors or nurses from the AAs can decrypt, verify and use the data in this transaction. Once the access structure is decided, the cluster head will apply ABE to encrypt the sensor data and append the ciphertext in the transaction as shown in Figure 3. Then the hash value of the transaction data will be signed by the private key of the cluster head to generate a digital signature. The generated digital signature will also be appended in the transaction data. Finally, this transaction data will be announced to the blockchain network by the cluster head. The following subsection describes the verification step.

Setup $\mathcal{S}$ For a given security parameter λ, $\mathcal{S}$ generates the bilinear groups $\mathbb{G}_1$ and $\mathbb{G}_2$ with prime order p as follows: $\{\mathbb{G}_1, \mathbb{G}_2\} \leftarrow \mathcal{GS}(1^\lambda)$. Let $e : \mathbb{G}_1 \times \mathbb{G}_1 \longrightarrow \mathbb{G}_2$ be a bilinear map and g, h and h_1 be the generators of $\mathbb{G}_1$ such that $\forall\, x, y \in \mathbb{G}_1$ and $\forall\, a,\ b \in \mathbb{Z}_p$, $e(x^a, y^b) = e(x, y)^{ab}$. There are N number of authorities $\{A_1, \ldots, A_N\}$: A_k monitors n_k attributes i.e. $\tilde{A}_k = \{a_{k,1} \ldots, a_{k,n_k}\}, \forall k$.

AA Setup $\mathcal{AS}$ Security parameters of A_k: $SK_k = \{\alpha_k, \beta_k,$ and $[t_{k,1}, \ldots, t_{k,n_k}]\} \xleftarrow{R} \mathbb{Z}_p, \forall k$. Public parameters of A_k: $PK_k = \{Y_k = e(g,g)^{\alpha_k},\ Z_k = g^{\beta_k},$ and $[T_{k,1} = g^{t_{k,1}}, \ldots, T_{k,n_k} = g^{t_{k,n_k}}]\}$, $\forall\ k$. A_k specifies m_k as minimum number of attributes required to satisfy the access structure ($m_k \leq n_k$).

Key Generation $\mathcal{KG}$ Collision-Resistant Hash Function $H : \{0,1\}^* \rightarrow \mathbb{Z}_p$ to generate u from the miners' global identity. Attribute set of miner is $\tilde{A}_u$: $\tilde{A}_u \cap \tilde{A}_k = \tilde{A}_u^k\ \forall\ k$. A_k generates $r_{k,u} \in_R \mathbb{Z}_p$ and polynomial q_x for each node x (including the leaves) $\mathbb{T}$. For each node x, the degree d_x of the polynomial q_x is $d_x = k_x - 1$ where k_x – threshold value of that node. Now, for the root node r, set $q_r(0) = r_{k,u}$. For any other node x, set $q_x(0) = q_{parent(x)}(index(x))$. Now decryption keys for the user u are generated as follows: $D = D_{k,u} = g^{-\alpha_k} h^{\frac{\beta_k}{r_{k,u}+u}} h_1^{\frac{r_{k,u}}{\beta_k+u}}, D_{k,u}^1 = h^{\frac{1}{r_{k,u}+u}}, D_{k,u}^j = h_1^{\frac{q_{a_{k,j}}(0)}{(\beta_k+u)t_{k,j}}}, \forall a_{k,j} \in \tilde{A}_u^k$.

Encryption $\mathcal{E}$ Attribute set for the transaction m is $\tilde{A}_m$: $\tilde{A}_m \cap \tilde{A}_k = \tilde{A}_m^k, \forall k$, i.e. $\tilde{A}_m = \{\tilde{A}_m^1, \ldots, \tilde{A}_m^k, \ldots, \tilde{A}_m^N\}$. The cluster head of m randomly chooses $s \in_R \mathbb{Z}_p$, and outputs the ciphertext as follows: $C = C_1 = m. \prod_{k \in I_C} e(g,g)^{\alpha_k s}, C_2 = g^s, C_3 = \prod_{k \in I_C} g^{\beta_k s}$ and $\left\{C_{k,j} = T_{k,j}^s\right\}_{\forall k \in I_C, a_{k,j} \in \tilde{A}_m^j}$ where I_c denotes the index set of the authorities.

Decryption $\mathcal{D}$ In order to decrypt C, the miner u, computes X, Y and Q_k as follows: $X = \prod_{k \in I_C} e\left(C_2, D_{k,u}\right)$, $Y = e\left(C_3, D_{k,u}^1\right)$ and $S_k = \prod_{a_{k,j} \in A_m^k} e\left(C_{k,j}, D_{k,u}^j\right)^{\Delta_{a_{k,j}, \tilde{A}_m^j}(0)}$. Miner then decrypts m as follows: $m = \frac{C_1 X}{Y \prod_{k \in I_C} S_k}$.

Figure 2: The decentralized ABE scheme adopted for the blockchain based IoT Ecosystem [8]

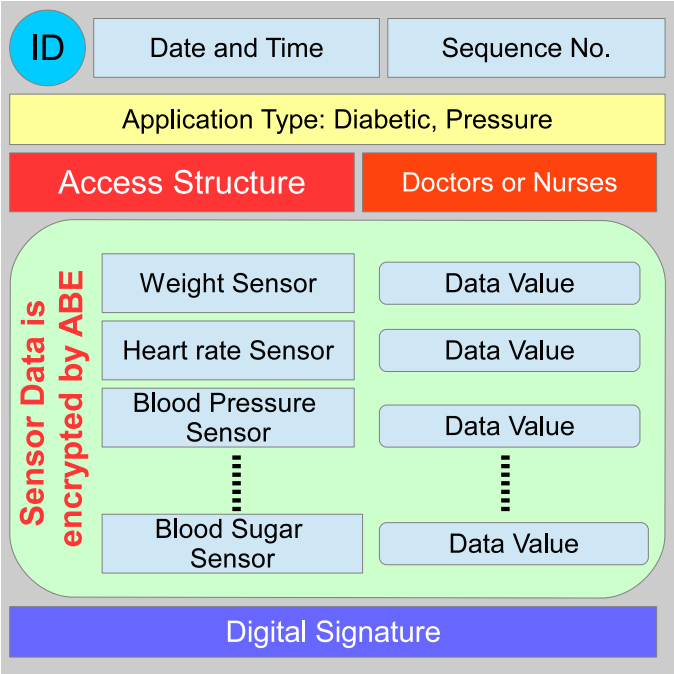


Figure 3: The structure of transaction data generated by the cluster head. This will be shared with peers in the blockchain network for verification followed by storage in the blockchain.

2) Transaction verification: As shown in Figure 1, the miners who are connected directly with the cluster head obtain this transaction data. Then these miners will spread the transaction data to other miners for verification. Eventually all the miners in the blockchain network will receive the transaction data.

Each miner will check if s/he has right attributes to verify the transaction. S/he will not proceed if s/he does not have the right attributes. Otherwise, s/he will retrieve the public key and sensor types' details corresponding to the ID from the hospital server or from the initial block of the blockchain.

S/he will then use his/her credentials obtained from the AAs for the given attributes to decrypt the encrypted sensor data. Then the miner will cross check if the types of sensors used in the transaction is matching with the sensor types obtained from the hospital server. Then s/he will check if the data value for each sensor is within the predefined range. For example blood pressure value must be between *0mmHg* and *300mmHg*. This will avoid out-of-range values.

If there is new sensor information or the range of a given sensor data is out of range, the transaction data will be rejected by the miner. This news will be spread across the blockchain network. If the transaction is verified by most of miners who have right credentials for the attributes then this transaction data will be passed and queued in the pending block.

3) Mining and adding new block to the blockchain: Similar to the blockchain model used in crypto currencies, mining of new blocks in this IoT scenario will also be done periodically. For example, every ten minutes a new block will be mined and appended to the existing blockchain.

All the verified transaction data in the pending block will be used by *any miners* for mining a new block i.e., mining a new block is not restricted by attributes. As per blockchain principles [13], miners will try to find a new hash value for the pending transaction data subject to restrictions e.g. the hash value must contain 50 leading zeros.

The restriction will be increased periodically based on the increment of computing power used by the miners in the network. For example, if the blockchain is updated every 10 minutes and the new hash value for the current block is obtained by miners in five minutes then it means the computing power of miners have doubled. Hence the restriction for generating new hash value will be increased to match the computing power.

4) Reward system for miners: Blockchain based on crypto currency models offer crypto coins to the successful miner who gets the new hash value for the pending transactions. When it comes to IoT ecosystems, the miners must be rewarded to make this model sustainable. In the big data world, data is more valuable and various service providers and research organisations rely on accurate data to provide efficient services. The miners can be rewarded with *tokens* to access the data in the future.

III. Security and Privacy Analysis

Since the proposed solution enforces ABE, the number of miners who qualify for verification is less than the traditional blockchain network. If the cluster head chooses too many attributes or very specific attributes then the number of qualified miners will be reduced. If there are too little miners who can verify the transaction then that may jeopardise the security of the blockchain technology. For example a malicious cluster head may collude with malicious miners to approve falsified transactions by assigning very specific attributes.

To avoid this problem, the blockchain protocol will specify the minimum number of miners for verification. To avoid the case where only few miners are qualified for specific attributes, the AA will be forced to wait until the number of miners for an attribute surpasses the minimum number requirement set by the blockchain protocol.

As we discussed in the introduction, the related works either keep the transaction data in the plain domain or use symmetric encryption like the AES for encryption. If the transaction data is encrypted using AES then the encrypted key will need to be shared with the transaction to enable verification. This means anyone in the blockchain network can *see* the data and there is no advantage of encryption.

In the proposed work, the number of miners who can view the data is limited. Only miners who have the right attributes can see the data. The AAs will scrutinise the miners for their claimed attributes before issuing the credentials. For example, miners associated with hospitals may get the credentials for "DOCTORS" and "NURSES". However, once the verification is performed, any miners in the block chain network can contribute to mining a new block regardless of their attributes. Hence, the blockchain's concept of *proof-of-work (PoW)* is still preserved in the proposed model.

Traditional IoT systems are known for sybil attacks. However, in the proposed blockchain-powered IoT system, the transaction data is verified by a large number of miners before it is accepted. Unless the transaction data are added to the new block, sybil attacks have limited impact. Since the proposed model is built on top of the well researched decentralised ABE and blockchain technology, we can assume that there is no security vulnerability in the rest of the model.

IV. Numerical analyses

Let us first compare the proposed model against AES based blockchain models [2], [5], [6] in terms of scalability and key management. When AES is used for encrypting transactions then the number of symmetric keys used in the system is proportional to the number of cluster heads times the number of unique keys used by cluster heads. This will exponentially increase the complexity of key management. However, when it comes to ABE, the total number keys used in the system is proportional to the number of attributes. In simple terms, if we consider healthcare IoT ecosystems with 10000 patients then we need to manage 10000 AES keys. On the contrary, for the ABE based proposed scheme, for example, we can run the healthcare model with 10000 patients with just 50 attributes hence only 50 keys.

Nevertheless, the inclusion of ABE will increase the computational cost for encryption and decryption in contrast to AES systems. Let us numerically check the computation costs associated with the inclusion of ABE scheme for blockchains. As shown in Figure 2, the ABE scheme has five distinct protocols. However, the miners are only involved in the computation during the decryption step and the cluster heads are involved in the encryption step. Since the other three protocols can be executed off-line, we simply ignore them for this comparison. Also the computational cost for hash functions used in Figure 2 is negligible compared to pairing and exponentiation. Let the computational time (in ms) for one multiplication, one exponentiation, and one pairing be denoted as C_m, C_{ex}, and C_p, respectively.

For comparison, we use the benchmark time values given in the popular pairing-based cryptography library namely jPBC in [14]. Table I compares the time values (in ms) for C_m, C_{ex}, and C_p for two different testbeds: testbed 1 uses Intel(R) Core(TM) 2 Quad CPU Q6600 with 2.40GHz and 3 GB memory running on Ubuntu 10.04 and testbed 2 uses HTC Desire HD A9191 smart phone running on Android 2.2. We can safely assume that the miners use the powerful testbed 1 for decryption while cluster heads use mobile devices similar to the testbed 2 for the encryption.

Table I: Time complexity measures for two different testbeds.

	Testbed 1 (ms)	Testbed 2 (ms)
C_p	14.6	491.2
C_{ex}	2.8	34.1
C_m	1.8	20

Let us denote the number of attributes used for encryption as n per AA. Table II shows the total time required for encryption (by the cluster head) and for decryption (by the miners) for different number of AAs K. In order to graphically visualize the time complexity of the ABE, we plotted the time complexities given in Table II by varying the number of attributes, n, and number of AAs K in Figure 4.

From Figures 4a to 4d, the time complexity increases linearly with the number of attributes per AAs for both cluster head and miners. Time complexity for decryption is almost four times quicker than encryption even though the processing power of Testbed 1 is ten times faster than Testbed 2 in Table I. This is due to the number of operations involved in the decryption process.

It can also be noted from Figures 4a to 4d, that time complexity slightly increases when there more number on AAs for the same number of attributes. For example, if we consider that there are in total ten attributes, then time complexities for encryption and decryption for one AA are *500ms* and *225ms*, respectively. However, when the number of AAs increases to five, the time complexities for encryption and decryption for two attributes per AA are *900ms* and *300ms*, respectively. As per ABE literature, distributing the attributes across AAs will increase the security rather than having a single AA. Hence, the presence of multiple AAs increases the security subject to slight increase in time complexity.

Overall, for ten attributes-based encryption and decryption, the ABE based blockchain model proposed in this paper is nearly $\frac{1}{2}$ a second slower than a blockchain model without any encryption. As can be seen from Table I that the performance comparison is based on four-year-old testbeds and the time complexity values for the current state-of-the-art mobiles and computers must be much faster than the old testbeds. Nevertheless, the proposed model provides privacy assurance to the transaction data of the blockchain not only during the verification but also in the blockchain. The fine-grained nature of ABE only allows certain miners and users with specific attributes to access the data. However, any miners in the network can contribute to the mining process of new blocks regardless of their attributes.

V. Conclusions

This paper proposes a novel blockchain architecture to preserve the privacy of transaction data using the attribute-based encryption technique. This is the first approach that combines the state-of-the-art encryption technique with the blockchain technology. The simplicity and fine-grained nature of attribute based encryption controls who can *see and use* the transaction data. The proposed model slightly changed the blockchain protocol procedure to adopt the attribute-based encryption technique without jeopardising the fundamental security properties of blockchains. We analyse the security and privacy of the proposed model and developed strategies to mitigate some known attacks. The numerical analysis section showed that the blockchain-powered IoT can benefit from attribute based encryption in terms of achieving privacy for minimal computational overhead.

References

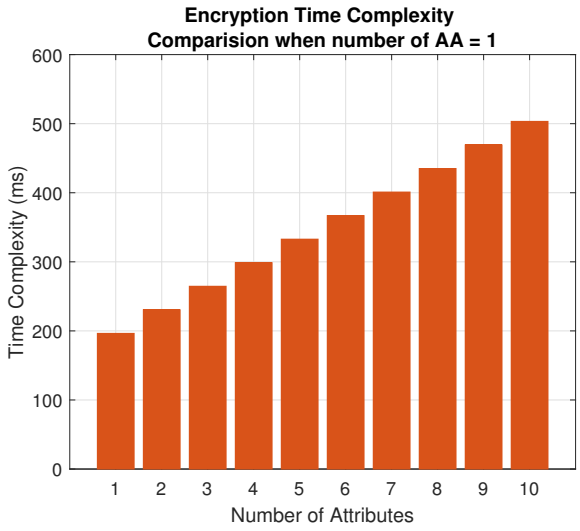
[1] Atzori, L., Iera, A. and Morabito, G., 2010. The internet of things: A survey. Computer networks, 54(15), pp.2787-2805.

[2] Dorri, A., Kanhere, S.S. and Jurdak, R., 2017, April. Towards an Optimized BlockChain for IoT. In Proceedings of the Second International Conference on Internet-of-Things Design and Implementation (pp. 173-178). ACM.

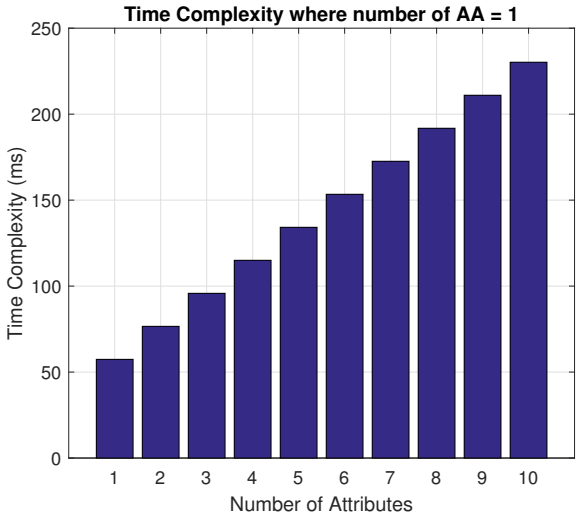
[3] Stanciu, A., 2017, May. Blockchain Based Distributed Control System for Edge Computing. In Control Systems and Computer Science (CSCS), 2017 21st International Conference on (pp. 667-671). IEEE.

[4] Conoscenti, M., Vetrò, A. and De Martin, J.C., 2017, May. Peer to Peer for Privacy and Decentralization in the Internet of Things. In Proceedings of the 39th International Conference on Software Engineering Companion (pp. 288-290). IEEE Press.

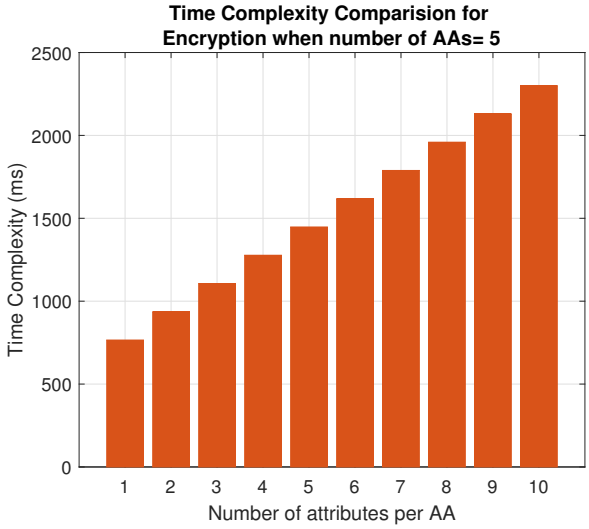
[5] Boudguiga, A., Bouzerna, N., Granboulan, L., Olivereau, A., Quesnel, F., Roger, A. and Sirdey, R., 2017, April. Towards Better Availability and Accountability for IoT Updates by means of a Blockchain. In IEEE Security & Privacy on the Blockchain (IEEE S&B 2017) an IEEE EuroS&P 2017 and Eurocrypt 2017 affiliated workshop.



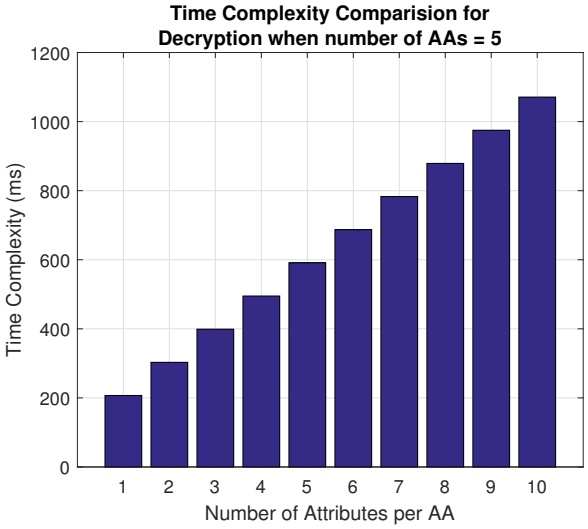
(a) Time complexity for encryption when $K = 1$



(b) Time complexity for decryption when $K = 1$



(c) Time complexity for encryption when $K = 5$



(d) Time complexity for decryption when $K = 5$

Figure 4: Time complexity addition to the blockchain network due to the inclusion of ABE.

Table II: Time complexity comparison when there are K number of AAs in the ABE system

	Time Complexity for encryption and decryption for the ABE scheme in Figure 2
Encryption	$[(2+n)K+1]C_{ex} + (2K+1)C_m + (2K+1)C_m$
Decryption	$[(n+1)K+1]C_p + nKC_e + [3+(2+n)K]C_m$

[6] Dorri, A., Kanhere, S.S., Jurdak, R. and Gauravaram, P., 2017, March. Blockchain for IoT security and privacy: The case study of a smart home. In Pervasive Computing and Communications Workshops (PerCom Workshops), 2017 IEEE International Conference on (pp. 618-623). IEEE.

[7] Sato, M., 2017, April. Long-Term Public Blockchain: Resilience against Compromise of Underlying Cryptography. In Security and Privacy Workshops (EuroS&PW), 2017 IEEE European Symposium on (pp. 42-49). IEEE.

[8] Goyal, V., Pandey, O., Sahai, A. and Waters, B., 2006, October. Attribute-based encryption for fine-grained access control of encrypted data. In Proceedings of the 13th ACM conference on Computer and communications security (pp. 89-98). Acm.

[9] Rahulamathavan, Y., Veluru, S., Han, J., Li, F., Rajarajan, M. and Lu, R., 2016. User Collusion Avoidance Scheme for Privacy-Preserving Decentralized Key-Policy Attribute-Based Encryption. IEEE Transactions on Computers, 65(9), pp.2939-2946.

[10] Li, F., Rahulamathavan, Y., Rajarajan, M. and Phan, R.C.W., 2013, March. Low complexity multi-authority attribute based encryption scheme for mobile cloud computing. In Service Oriented System Engineering (SOSE), 2013 IEEE 7th International Symposium on (pp. 573-577). IEEE.

[11] Li, F., Rahulamathavan, Y., Conti, M. and Rajarajan, M., 2015. Robust access control framework for mobile cloud computing network. Computer Communications, 68, pp.61-72.

[12] Li, F., Rahulamathavan, Y. and Rajarajan, M., 2014, September. LSD-ABAC: Lightweight static and dynamic attributes based access control scheme for secure data access in mobile environment. In Local Computer Networks (LCN), 2014 IEEE 39th Conference on (pp. 354-361). IEEE.

[13] Tschorsch, F. and Scheuermann, B., 2016. Bitcoin and beyond: A technical survey on decentralized digital currencies. IEEE Communications Surveys & Tutorials, 18(3), pp.2084-2123.

[14] Caro, A.D., 2013. The java pairing based cryptography library (jpbc). URL: http://gas. dia. unisa. it/projects/jpbc/, laatst nagekeken op, pp.02-24.