A data-owner centric privacy model with blockchain and adapted attribute-based encryption for internet-of-things and cloud environment

Youcef Ould-Yahia, Samia Bouzefrane, Hanifa Boucheneb, Soumya Banerjee

A data-owner centric privacy model with blockchain and adapted attribute-based encryption for Internet-of-Things and Cloud environment

Youcef Ould-Yahia* , Samia Bouzefrane

CEDRIC Lab, Conservatoire National des Arts et Métiers, 292 rue Saint Martin 75141, Paris Cédex 03, France
E-mail: youcef.ouldyahia.auditeur@lecnam.net
E-mail: samia.bouzefrane@cnam.fr
* Corresponding author

Hanifa Boucheneb

VeriForm Lab, Ecole Polytechnique de Montréal, P.O. Box 6079, Station Centre-ville, Montréal, Québec, Canada, H3C 3A7
E-mail: hanifa.boucheneb@polymtl.ca

Soumya Banerjee

CEDRIC Lab, Conservatoire National des Arts et Métiers, 292 rue Saint Martin 75141, Paris Cédex 03, France
E-mail: dr.soumya@ieee.org

Abstract: Advances in Internet of Things (IoT) and cloud computing technologies have led to the emergence of new applications such as in e-Health domain bringing convenience for both physicians and patients. However, the development of these new technologies makes users' privacy vulnerable. The threats on private data may arise from service providers themselves voluntarily or by inadvertence. As a result, the data owner would like to ensure that the collected data are securely stored and accessed only by authorized users. In this paper, we propose a novel data-owner centric privacy model in IoT/cloud environment. Our model combines two promising paradigms for data privacy, which are Attribute-Based Encryption (ABE) and blockchain, to strengthen the data-owner privacy protection. We propose a new scheme of ABE that is, in one hand, suitable to resource-constrained devices by externalizing the computing capabilities, thanks to Fog computing paradigm and, in the other hand, combined with a blockchain-based protocol to overcome a single point of trust and to enhance data-owner access control.

Keywords: IoT; Cloud; Privacy; Fog computing; Blockchain; Attribute-Based Encryption; e-Health.

Reference to this paper should be made as follows:.

Biographical notes: Youcef Ould-Yahia pursued his PhD thesis at Conservatoire National des Arts et Métiers, Paris. He holds a master of cyber-security from IMT Atlantique (France), and a computer engineering degree from Polytechnic

Military School, Algiers. His areas of research interest include trust evaluation of the service providers and data privacy in Internet of Things and Cloud computing.
Samia Bouzefrane is Associate Professor and has the accreditation to conduct research (HDR) at the Conservatoire National des Arts et Métiers (CNAM) of Paris. She received her PHD in Computer Science from the University of Poitiers (France) in 1998. After four years at the University of Le Havre (France), she joined in 2002 the CEDRIC Lab of CNAM. She is the co-author of many books (Operating Systems, Smart Cards, and Identity management Systems). Her current research areas cover security in Cloud computing and Internet of Things, resource allocation in Cloud Computing, and new paradigms for networking such as NFV and NDN.
Hanifa Boucheneb is currently a full Professor in the computer and software engineering department at Polytechnique of Montréal. In 1988, she received her Master degree in computer science from USTHB University of Algeria, under supervision of Professor Michel Diaz (LAAS-France). In 1999, she completed her PhD at the USTHB University of Algeria under supervision of Professor Gerard Berthelot (CNAM-France). Her research interests include formal verification techniques and their applications to real time and infinite complex systems. She published more than 100 research papers in international journals, conferences, workshops and books. She has served as a member and chair in several program committees of conferences and workshops.
Soumya Banerjee completed his Ph.D in Computer Science and Engg. from Birla Institute of Technology, Mesra, India on Stigmergic optimization with Hybrid Intelligence in 2009. He has more than 120 international journal publications including 32 book chapters and 46 International top level conference proceedings published from Elsevier Science, IEEE Transactions, ACM, SpringerÂ–Verlag Germany, CRC Press, and Idea Publication USA. He is an active project participant and consultant in IRIDIA (The National Lab of Computational Intelligence), Belgium, and Simula lab.Norway. He was also visiting professor in 2016 (April-August) at CNRS INSA de-Lyon, France; and in 2017, at Cnam and Inria Paris France for 6 months. At present he is a visiting research professor at CEDRIC Lab, Cnam, Paris working on various machine learning and IoT Security projects.

1 Introduction

Cloud computing provides storage for personal-data as well as computational capabilities for connected objects that have limited resources such as in Internet of Things (IoT). Recently to take advantage of the Cloud-computing efficiency, the edge and Fog computing paradigms have been introduced to bring Cloud services closer to end users Bonomi et al. (2012), Yi et al. (2015). These services are leveraged directly at the edge of the network such as access points, routers or cloudlets Premsankar et al. (2018), El-Barbary et al. (2015). One of the core applications that are benefiting from these technologies is the e-health domain. E-Health permits to enable patients to be monitored at any time and anywhere by a spread of ubiquitous connected devices, while sharing his/her data stored in the Cloud, thanks to internet of things paradigm Gope & Hwang (2016) and mobile cloud computing Dinh Thai et al. (2013).

As a consequence of the personal-data outsourcing, their security and privacy have become one of the main concerns in e-Health applications from the perspective of data owner. This is because e-Health applications are not exclusively dedicated to healthcare professionals unlike telemedicine. The e-Health model is rather centered and pivoted on the

consumers of health systems Chiuchisan et al. (2015). As a result, the data owner would like to ensure that collected data are safely computed in the edge/Fog nodes and securely stored in the Cloud server and accessed only by authorized users. However, the data security in a Cloud computing is not guaranteed Malarchelvi et al. (2019), Pussewalage & Oleshchuk (2016), and the threats may arise from service providers themselves, as many companies have significant commercial interests in collecting private health data Lin et al. (2013). Furthermore, complex security algorithms and protocols cannot be used in IoT and mobile devices echosystem due to the limited physical and energy resources Ouada et al. (2016).

For these reasons, in the last few years, a data-owner centric model has emerged in the literature for personal-data protection. According to the existing literature, the attribute-based encryption (ABE) scheme Sahai & Waters (2005) is a viable way to enhance privacy in many domains like e-Health applications Kocabas et al. (2016), Pussewalage & Oleshchuk (2016), Ould-Yahia et al. (2018), smart home data privacy Chowdhury et al. (2017) and data security in Cloud computing Namasudra (2017). In addition, to securing data transmission and storage, ABE provides a fine-grained access control and a flexible data distribution Li et al. (2010), Hemalatha & Manickachezian (2014). Nevertheless, ABE schemes involve computationally intensive pairing operations and exponentiations, and their complexity increases linearly with the number of attributes. Currently, even if an efficient realization of ABE schemes can be implemented using conventional computers (PC, server...), for the limited computational resources devices (ex. sensors, mobile and IoT devices) it remains a challenging task to develop applications using such devices to implement ABE schemes Wang et al. (2014), Ambrosin et al. (2015). As a means of providing a solution for this issue, Asim et al. (2014), Guo et al. (2014), Touati et al. (2014), Zhou & Huang (2012) and Zhang et al. (2018) had proposed an outsourcing-based ABE. However, the main common limitations of these works are ineffective in practice due to there implementation restriction such as a strict multi-authority requirement, the single point of failure represented by a trust authority and the possibility for that entity to access the encrypted personal-data. Thus, the second challenge is to overcome the single point of failure that represents the trusted authority Pussewalage & Oleshchuk (2016). The underlying principle of the decentralized trust, autonomous and self validation, provided by Blockchain paradigm, allows to imagine new applications for access control and personal data sharing such as in Sukhodolskiy & Zapechnikov (2018), Azaria et al. (2016), Zyskind et al. (2015) and Hashemi et al. (2016).

To overcome these challenges, we propose a Fog-Computing Cipher-Policy Attribute-Based Encryption (FCCP-ABE), which is a new ABE scheme that is suitable for resource-constrained devices, enriched with a blockchain access-authorization record (blockchain A2R). This work is motivated by the fact that the privacy protection includes the data ownership control of encryption and access process Zhang & Liu (2010) in order to achieve a data-owner centric privacy protection. The main contributions of this paper are as follows:

1. We provide a new attribute-based encryption scheme that securely outsources encryption from IoT devices to Fog nodes, without the possibility for the Fog node to recover the unencrypted data. This scheme reduces the computational load of the resource-constrained devices like sensors and mobile devices. Unlike the existing solutions, our scheme design incurs less network load by reducing both the messages exchange and the length of the ciphered data generated, stored and transited by the constrained devices. The security of our proposed scheme is proven under the decisional bilinear Diffie Hellman assumption.

2. In addition, we propose a data-owner centric security model that focuses on privacy protection based on a sound combination of the provided attribute-based encryption scheme for IoT devices and a well-tested and implemented blockchain paradigm, that ensures the integrity and the non-repudiation of access control event. In the proposed design, the new attribute based encryption scheme ensures data security in IoT-Cloud environment by achieving encryption and fine-grained access control capabilities. In the meanwhile, the blockchain access-authorization record provides a decentralized ledger for access-control messages by transforming digital assets as access-right credentials. Furthermore, to enhance privacy, the Cloud verifies the authenticity of the access request without knowing the requester identity.

The remainder of this paper is organized as follows. We start first by the related work in Section 2 before introducing the necessary background in Section 3. Section 4 is dedicated to our novel ABE design and its security proof. In Section 5, we present our proposed system framework followed in Section 6, by the theoretical and experimental analysis of the proposed model. Then, we conclude the paper with the future scope and directions of our work in terms of data protection.

2 Related work

Attribute-based encryption schemes are typically computationally intensive and not adapted for resource-constrained devices like sensors and smartphones, as stated by Wang et al. (2014) that evaluated the Java implementation of ABE on an Android smartphone. In Ambrosin et al. (2015), a C implementation of ABE was performed on Android. In Ambrosin et al. (2016), the same evaluation is performed for IoT devices for a realistic use case (healthcare remote monitoring) using up to 10 attributes and an 80-bit security level. The conclusion is that even if ABE is feasible on constrained devices, the feasibility is strongly dependent on the application requirements mainly the security levels and the data transmission rates. Thus, it remains that the main drawback is that the performance depends on the number of attributes on the access policy. For our concern, we focus below on the encryption operation that is performed on constrained devices.

Regarding the usage of ABE in constrained devices like sensor networks, the authors of Guo et al. (2014) propose a solution for resource-constrained devices but they focus only on the key generation efficiency. The common model proposed in the literature is to have constrained devices that implement a lighweight encryption and to outsource the heavy computational load to devices that are unconstrained. This kind of solution is proposed by Touati et al. (2014) where the constrained devices involve unconstrained trusted neighbours which can support ABE computing. The authors propose to split mathematicaly the secret shared parameter into n parts for each attribute in the access policy and share pairwise keys with the n trusted unconstrained nodes. Since we need multi-support devices, implementing this scheme is limited in the real world. The solution presented in Asim et al.Asim et al. (2014) aims to outsource the heavy computational operations to the cloud. However, it requires access to a third trusted party proxy/cloud and needs to establish and to maintain the connection between IoT devices and the proxy, which can generate an overload on message exchanges.

Using Fog or edge computing paradigm is one of the best ways to implement attribute-based encryption in IoT ecosystem to reduce the overhead on resource-constrained devices, by outsourcing the computational operations. Zhou and Huang Zhou & Huang

(2012) propose a privacy preserving CP-ABE scheme by outsourcing heavy computational operations to the encryption service provider. The complexity of the encryption algorithm is irrelevent to the number of attributes in access structure, however, it depends on the number of attributes used by the data owner to encrypt with. To the best of our knowledge, Zhang et al. (2018) propose a scheme with the most efficient outsourcing encryption capacity. Nevertheless, in their protocol, the data owner first transmits to the edge/Fog node the access structure that generates an intermediary ciphertext which is returned to the IoT node to compute the final ciphertext before being uploaded to the Cloud service provider. Furthermore, the ciphertext length generated by the IoT node depends on the number of attributes used in access policy, which leads to memory and energy consumption.

To manage the access control, a recent interest of the scientific community to use blockchain to improve privacy is observed Azaria et al. (2016), Zyskind et al. (2015). The characteristics of the blockchain allow to target privacy concerns while focusing on personal data. A blockchain is a ledger and not a fully database that can deal with a huge data, the authors of Zyskind et al. (2015) combine blockchain and off-blockchain storage to build a personal data management platform for data privacy. The authors Azaria et al. (2016) and Sukhodolskiy & Zapechnikov (2018) use the smart contract, based on blockchain, and developed on the Ethereum platform, to manage patient access and data sharing. Authors of Hashemi et al. (2016) and Ouaddah et al. (2017) propose a solution for a particular IoT environment. Hashemi et al.'s solution for the IoT data management is centered on their owner. The general framework of the proposed solutions aims to separate the data storage from the data management. This management is done with a blockchain, which is a decentralised solution that overcomes a single point of trust and failure, whereas the data themselves are stored in an off-blockchain like a Cloud/server.

3 Preliminary and background

3.1 Blockchain

The blockchain is a distributed tamper-proof ledger and indestructible which avoids the use of trusted central authorities. The first description of blockchain is proposed by Satoshi Nakamoto Nakamoto (2008) for financial purposes. It allows different parties to transact safely without a need for trusted third parties to ensure verification and compliance. More recently, applications in other fields are explored such as supply chain management and product tracking Anjum et al. (2017), healthcare Mettler (2016), Cloud storage access control Sukhodolskiy & Zapechnikov (2018), Kocabas et al. (2016) and secure sharing of IoT datasets Banerjee et al. (2017).

The main idea of blockchain is to build a stand-alone, autonomous, self-verification and validation of applications that do not rely on a centralized trusted authority. This allows to perform trusted transactions in untrusted networks. One factor that drives the interest in blockchain is the ease with which it can be added to existing workflows Anjum et al. (2017).

Basically, in the blockchain, all the transactions are logged including different useful information like the date, the time, the participants' addresses and the transaction subject (business, record-keeping, access control, etc.).

The typical process of the blockchain system begins when a user broadcasts a transaction. This transaction is gathered into a block. Once a node validates this block by a process called "mining", it broadcasts it to the network. Finally, a consensus method between participant

nodes is applied to select the block to be added to the blockchain. Interested readers can refer to Nakamoto (2008) and Anjum et al. (2017) for more details on blockchain principle.

3.2 Attribute-Based Encryption

Attribute-based encryption (ABE) is a public key one-to-many encryption scheme, introduced by Sahai & Waters (2005). It is a pairing based encryption scheme, using bilinear map.

The two main variants are Ciphertext Policy Attribute-Based Encryption (CP-ABE) Bethencourt et al. (2007) and Key Policy Attribute-Based Encryption (KP-ABE) Goyal et al. (2006). For our concern, we focus on CP-ABE, which is one of the most fine-grained cryptographic access control techniques Zhang & Liu (2010). As we can see in Figure 1a, CP-ABE work-flow consists of four algorithms, as described in Table 1.

Table 1 List of CP-ABE algorithms

Algorithm	**Input/Output**	**Description**
Setup	*Input*:	Security parameter.
	Output:	The public key for encryption Pk and a master secret key Msk to generate decryption key.
Encrypt	*Input*:	Message m, public key Pk and policy P.
	Output:	Ciphertext "c".
KeyGen	*Input*:	Master Secret Key Msk and a set of attributes a_i.
	Output:	The decryption secret key Sk.
Decrypt	*Input*:	The cipher text c, The decryption secret key Sk.
	Output:	If attributes a_i satisfy the policy P then m else $\perp$.

In CP-ABE the access policy is embedded in the ciphertext, and the secret keys are generated with a set of attributes. Only the secret key with a set of attributes that satisfies the previous access policy can retrieve the clear text. ABE allows encrypting the data without any prior knowledge of the identities of the recipients, and provides both cryptographic data protection and access control capability. Thus, it provides a scalable key management and a flexible data distribution Hemalatha & Manickachezian (2014), Li et al. (2010). These characteristics seem to be interesting solutions for user privacy concerns Wang et al. (2014) and data-centric control. However, the main challenge to implement the ABE for resource-constrained devices is the computationally cost Ambrosin et al. (2015).

3.3 Symmetric bilinear maps

A symmetric bilinear map application denoted $e : \mathbb{G}_0 \times \mathbb{G}_0 \rightarrow \mathbb{G}_T$ is an application that maps two elements of $\mathbb{G}_0$ to an element in $\mathbb{G}_T$, where $\mathbb{G}_0$ and $\mathbb{G}_T$ are two multiplicative cyclic groups of prime order p with g as a generator of $\mathbb{G}_0$ that satisfies the following properties :

1. Bilinearity: $\forall u, v \in \mathbb{G}_0$ and $a, b \in \mathbb{Z}_p : e(u^a, v^b) = e(u, v)^{ab}$.
2. Non-degeneracy : $e(g, g) \neq 1$
3. Symmetry : $\forall u, v \in \mathbb{G}_0 : e(u, v) = e(v, u)$

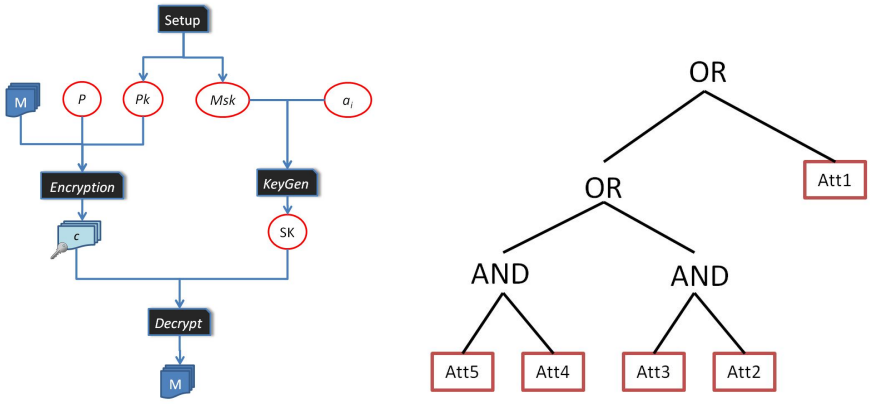
3.4 Access tree structure

Let τ denotes an access tree as illustrated in Figure 1b for a simple policy $P = Att1$ OR $((Att2$ AND $Att3)$ OR $(Att4$ AND $Att5))$. Each leaf node represents an attribute, and each internal node is a logical gate (AND, OR).

For the rest of this paper, we consider the following primitives:

- $parent(x)$ which returns the parent node of node x.
- $num(x)$ is the number of the children of the node x.
- $index(x)$, is defined in the access structure τ, to order between the children of each node by numbering each child from 1 to $num(parent(x))$. $index(x)$ returns such number as identification of node x.

We define also a threshold value $k_x = num(x)$ if x is an AND gate, and $k_x = 0$ if x is an OR gate or a leaf node. Each node x uses k_x as the polynomial degree for the threshold secret sharing scheme Shamir (1979), Benaloh & Leichter (1990).



(a) MD=05% (b) MD=10%

Figure 1: CP-ABE work-flow and simple access tree example

3.5 Decisional Bilinear Diffie-Hellman Assumption

Decisional Bilinear Diffie-Hellman (DBDH) assumption is a computational hardness assumption based on the computational difficulty of discrete logarithms in cyclic groups. We recall the DBDH assumption given by Waters (2011): a challenger selects a group $\mathbb{G}_0$ of prime order p, according to the chosen security parameter and g a generator of $\mathbb{G}_0$. Let $a, b, s \in \mathbb{Z}_p$ chosen randomly. If the challenger gives the adversary (g, g^a, g^b, g^s) then the adversary must not be able to distinguish a valid tuple $e(g,g)^{abs} \in \mathbb{G}_T$ from a random element $Z \in \mathbb{G}_T$. An algorithm $\mathcal{B}(g, g^a, g^b, g^s, T)$, with a challenge as input and with outputs 0 or 1, has advantage ϵ to solve DBDH in $\mathbb{G}_0$ if

$$\mid Pr[\mathcal{B}(g, g^a, g^b, g^s, T = e(g,g)^{abs}) = 0] - Pr[\mathcal{B}(g, g^a, g^b, g^s, T = Z) = 0] \mid \geq \epsilon$$

4 Fog-Computing Cipher-Policy Attribute-Based Encryption (FCCP-ABE)

To achieve a fine-grained access control with a storage security, we propose a new efficient and security proved scheme of cipher-policy attribute-based encryption, called the Fog-Computing Cipher-Policy Attribute-Based Encryption (FCCP-ABE). Being aware that the resource limitation is a major concern in IoT, we propose a design for encryption with a heavy-computational outsourcing. In this section, we present our design as well as the security proof of our scheme in a realistic security context.

4.1 Proposed design

Let $\mathbb{G}_0$ a multiplicative group of prime order p and g a generator. Let also $e : \mathbb{G}_0 \times \mathbb{G}_0 \rightarrow \mathbb{G}_T$ be the bilinear map and the Lagrange coefficient $\Delta_{i,S}$ defined as follows. For $i \in \mathbb{Z}_p$ and a set S of elements in $\mathbb{Z}_p$: $\delta_{i,S}(x) = \prod_{j \in S, j \neq i} \frac{x-j}{i-j}$. Additionally, we consider a hash function $H : \{0,1\}^* \rightarrow \mathbb{G}_0$ and a symmetric encryption primitive $Senc(M, Ks)$ that encrypts/decrypts a message M with a secret key Ks. To enhance the efficiency of our solution, we encrypt the core data with a symmetric encryption and the related key with FCCP-ABE. We define the following sub-algorithms as parts of our proposed Fog-Computing CP-ABE scheme with encryption outsourcing capability and privacy protection:

- **Setup**(λ): The algorithm generates a public key and a master secret key, according to the security parameter λ that determines the size of the group. It chooses randomly two numbers $\alpha, \beta \in \mathbb{Z}_p$ and an output (a public key and a master secret key):

$$Pk = (\mathbb{G}_0, g, h = g^{\beta}, e(g,g)^{\alpha/\beta})$$

$$Msk = (\alpha, \beta)$$

- **EncryptCons**(M, Pk, Ks): this encryption operation is performed by constrained devices and takes as input a message M, a public key for FCCP-ABE encryption Pk and a symmetric key Ks. The algorithm selects an element $t \in \mathbb{Z}_p$ randomly and computes the following:

$$C_1 = Ks \cdot e(g,g)^{\alpha t/\beta}$$

$$C_2 = h^t, C_3 = g^t$$

$$CT_{one} = (\tau, C_1, C_2, C_3, C_4 = SEnc(M, Ks))$$

- **EncryptUncons**(CT_{one}, τ, Pk): This algorithm is performed by unconstrained devices. To build a threshold secret sharing scheme, the algorithm begins by building a polynomial q_x with a degree $d_x = k_x - 1$ for each node x of the access tree τ (k_x is a threshold value defined in 3.4). Starting with the root node R, we select a random element $s \in \mathbb{Z}_p$ and we set $q_R(0) = s$. After that, in order to define completely the polynomial q_R, we choose d_R other points randomly $(u_i, q_R(u_i))$ with i from 1 to d_R.

For other nodes x, we set $q_x(0) = q_{parent(x)}(index(x))$ and we choose d_x other points randomly to finalize the definition of q_x. Let X a set of leaf nodes in τ (corresponding to a set of attributes used in the access policy). The final ciphertext is built by computing CT_{two}:

$$C'_2 = C_2 \cdot h^s = h^{t+s}$$

$$C'_3 = C_3 \cdot g^s = g^{t+s}$$

$$\forall x \in X : C_x = g^{q_x(0)}, C'_x = H(attrib(x))^{q_x(0)}$$

$$CT_{two} = (\tau, C_1, C_3, C'_2, C'_3, C_4, \forall x \in X : C_x, C'_x)$$

Where $attrib(x)$ denotes the attribute associated with the leaf node x.

- **KeyGen(**Msk, C_2, S**)**: This algorithm takes as input a master secret key, C_2 generated with *EncryptCons* and a set of attributes S. It associates a random element $r_i \in \mathbb{Z}_p$ for each attribute $i \in S$. To avoid collusion, a random element $r \in \mathbb{Z}_p$ is generated for each user Sahai & Waters (2005). The secret key is computed as in the following:

$$Sk = (D_1 = g^{(r+\alpha/\beta)}, D_2 = g^r \cdot C_2, \forall i \in S : D_i = g^r \cdot H(i)^{r_i}, D'_i = g^{r_i})$$

- **Decrypt(**CT_{two}, Sk**)** : The decryption algorithm calls the recursive algorithm $DecryptNode(CT_{two}, Sk, x)$ that takes as input a ciphertext, a secret key and a node x:
If the node x is a leaf then $i = attrib(x)$ and if $i \in S$ then

$$\begin{aligned} DecryptNode(CT_{two}, Sk, x) &= \frac{e(D_i, C_x)}{e(D'_i, C'_x)} \\ &= \frac{e(g^r \cdot H(i)^{r_i}, g^{q_x(0)})}{e(g^{r_i}, H(i)^{q_x(0)})} \\ &= \frac{e(g^r, g^{q_x(0)}) \cdot e(H(i)^{r_i}, g^{q_x(0)})}{e(g^{r_i}, H(i)^{q_x(0)})} \\ &= e(g, g)^{r q_x(0)} \end{aligned}$$

Otherwise $DecryptNode(CT_{two}, Sk, x) = \perp$. We recall, in the following, the recursive case as defined by Bethencourt et al. (2007) : when x is a non leaf node then $DecryptNode(CT_{two}, Sk, x)$ calls $DecryptNode(CT_{two}, Sk, z)$ for all the nodes z that are children of x and stores the output as F_z. If at least one $F_z = \perp$ then the node x is not satisfied and $DecryptNode(CT_{two}, Sk, x) = \perp$. Otherwise, let S_x be a $k_x - sized$ set of child nodes z, then we compute F_x using polynomial interpolation as in the following:

Let $i = index(z)$ and $S'_x = \{index(z) : z \in S_x\}$.

$$\begin{aligned} F_x &= \prod_{z \in S_x} F_z^{\Delta_{i,S'_x}(0)} \\ &= \prod_{z \in S_x} (e(g,g)^{r \cdot q_z(0)})^{\Delta_{i,S'_x}(0)} \\ &= \prod_{z \in S_x} (e(g,g)^{r \cdot q_{parent(z)}(index(z))})^{\Delta_{i,S'_x}(0)} \\ &= \prod_{z \in S_x} (e(g,g)^{r \cdot q_x(i)}) \cdot \Delta_{i,S'_x}(0) \\ &= e(g,g)^{r \cdot q_x(0)} \end{aligned}$$

Now, we can define **Decrypt**(CT_{two}, Sk) algorithm. The algorithm starts by calling $DecryptNode(CT_{two}, Sk, R)$. If S satisfies the access tree τ then we set:

$$\begin{aligned} F &= DecryptNode(CT_{two}, Sk, R) \\ &= e(g,g)^{r \cdot q_R(0)} \\ &= e(g,g)^{r \cdot s} \end{aligned}$$

And compute :

$$\begin{aligned} B &= \frac{e(D_2, C'_3)}{e(C_3, C'_2)} = \frac{e(g^r h^t, g^{t+s})}{e(g^t, h^{t+s})} \\ &= e(g^r, g^{t+s}) = e(g,g)^{r(t+s)} \end{aligned}$$

$$A = \frac{B}{F} = \frac{e(g,g)^{r(t+s)}}{e(g,g)^{r \cdot s}} = e(g,g)^{rt}$$

The algorithm decrypts the CT_{two} and retrieves Ks:

$$Ks = \frac{C_1 \cdot A}{e(C_3, D_1)}$$

Finally, $M = SEncr(C_4, Ks)$.

4.2 Security model

We assume that the edge or Fog computing and Cloud providers are honest but curious. It means that they are only trusted to execute protocols but they are not allowed to know any private data. An important security propriety required in our system is to resist attacks by collusion between the users, while the users may try to combine there rights in order to increase their privileges. The communication channels are assumed untrusted.

We define security for FCCP-ABE scheme by a game between a challenger and an attacker. Regarding the security objectives and the capacity of the adversary, we adopt the

chosen plaintext security model for our scheme similarly to De & Ruj (2015). We follow the security game defined by Waters (2011).

- **Init:** The probabilistic polynomial time adversary $\mathcal{A}$ chooses a set of attributes to generate a challenge access structure P^* and sends it to the challenger $\mathcal{B}$.
- **Setup:** The challenger $\mathcal{B}$ runs the Setup algorithm to generate the public parameters, PK and gives it to the adversary $\mathcal{A}$.
- **Phase 1:** The adversary makes a repeated secret keys query, each time with a new set of attributes S_i.
- **Challenge:** The adversary submits two equal-length messages m_0 and m_1 and gives a challenge access structure A^* such that none of the sets S_i from Phase 1 satisfies the given access structure. The challenger chooses a random γ, and encrypts m_γ under A^* by performing *EncryptCons* and *EncryptUncons* algorithms. The ciphertext CT^* is given to the adversary.
- **Phase 2:** Phase 1 is repeated with the condition that none set of selected attributes S_j satisfies the access structure provided as a challenge.
- **Guess:** The adversary outputs a guess γ' of γ.

We note that the model can easily be extended to handle chosen ciphertext attacks by allowing decryption queries in Phase 1 and Phase 2 Goyal et al. (2006).

4.3 Security proof

To prove the security of the proposed scheme, the common method used in leterature is to reduce the problem to the decisional bilinear DBDH assumption, defined in 3.5. Bellare (1999).

Definition 4.1: The advantage of an adversary $\mathcal{A}$ in the security game is defined as :

$$Adv_{\mathcal{A}} = Pr[\gamma' = \gamma] - \frac{1}{2}$$

Where $Pr[\gamma' = \gamma]$ is the probability that the adversary $\mathcal{A}$ outputs the right value of γ.

Theorem 1: *If a polynomial time adversary can break our scheme with non-negligible advantage, then a polynomial time simulator can be constructed to distinguish the DBDH tuple from a random one with a non-negligible advantage.*

Proof. Assume that we have a polynomial time adversary $\mathcal{A}$ with a non negligible advantage $Adv_A = \varepsilon$ that can break our scheme. We will demonstrate that we can build a simulator algorithm $\mathcal{B}$ that can play the DBDH problem as a security game with a non negligible advantage as well.

Let the challenger $\mathcal{C}$ set the groups $\mathbb{G}_0$ and $\mathbb{G}_T$ and an efficient computable bilinear map e and a generator g. The challenger $\mathcal{C}$ chooses randomly: $a, b, s \in \mathbb{Z}_p$ and selects randomly $\mu \in \{0, 1\}$. If $\mu = 0$ then the challenge is $(A, B, C, Z) = (g^a, g^b, g^c, e(g, g)^{abc})$, otherwise the challenge is (A, B, C, Z) with Z a random element of $\mathbb{G}_T$. $\mathcal{C}$ sends a challenge to $\mathcal{B}$.

Now, the simulator $\mathcal{B}$ runs $\mathcal{A}$ and proceeds as follows:

- **Init:** $\mathcal{A}$ chooses a set of attributes to generate a challenge access structure P^* and sends it to $\mathcal{B}$.
- **Setup** The algorithm $\mathcal{B}$ simulates the public and secret parameters of the scheme by generating them with a distribution that is identical to the original one. $\mathcal{B}$ sets $\alpha/\beta = ab$ then $e(g,g)^{\alpha/\beta} = e(g,g)^{ab} = e(A,B)$ and $\beta = b$, then $h = g^{\beta} = g^b = B$ and gives the public key PK^* to $\mathcal{A}$.
- **Phase 1:** $\mathcal{A}$ can adaptively submit any attribute set S to $\mathcal{B}$ such that S does not satisfy the access structure P^*, and $\mathcal{B}$ responds with the secret key SK^* corresponding to the submitted set S. To simulate secrets keys, $\mathcal{B}$ selects a random $r' \in \mathbb{Z}_p$ and implicitly computes $r = br' - ab$ then it sets $D_1 = g^{(\beta/\alpha)+r} = g^{ab+br'-ab} = g^{br'} = B^{r'}$ and selects a random $t' \in \mathbb{Z}_p$ in order to set implicitly $t = a - \frac{t'}{b}$. Then, it computes $D_2 = g^r h^t = g^{br'-ab} g^{g_{bt}} = g^{br'+t'} = B^{r'} g^{t'}$. For each $j \in S$, $\mathcal{B}$ selects randomly $r'_j \in \mathbb{Z}_p$ to compute implicitly $H(j)^{r_j} = g^{ab+r'_j}$ and cancels the terms with g^{ab} that we can not simulate in D_j. Finally, $D_j = g^{br'-ab} g^{ab+r'_j} = B^{r'} g^{r'_j}$ and $D'_j = g^{r'_j}$. At last, $\mathcal{B}$ gives the simulated secret key $SK' = \{D_1, D_2, \forall j \in S : D_j, D'_j\}$ to $\mathcal{A}$.
- **Challenge:** $\mathcal{A}$ submits two messages m_0 and m_1 with the same length to $\mathcal{B}$ as a challenge. $\mathcal{B}$ draws randomly $\gamma \in \{0,1\}$ and generates a challenge ciphertext CT^* corresponding to m_γ. Let's $t = c$ then $C_1^* = m_\gamma Z$ with $Z = e(g,g)^{abc}$ if $\mu = 0$, otherwise Z is a random element from $\mathbb{G}_T$. Let's randomly $s' \in \mathbb{Z}_p$ to implicitly evaluate $s = \frac{s'}{c}$ then $C_2'^* = g^{bc} = g^{\frac{s'}{c}} = B^{s'}$ and $C_3'^* = g^c g^s = g^{s'}$. At the end, $\mathcal{B}$ builds a secret sharing δ_i of s and computes $\forall i \in \mathbb{A}^* : c_i^* = g^{\delta_i}, c_i'^* = Hattrib(i)^{\delta_i}$ before returning the ciphertext to $\mathcal{A}$.
- **Phase 2:** The same as Phase 1.
- **Guess:** $\mathcal{A}$ outputs a guess γ' of γ. If $\gamma' = \gamma$ then $\mathcal{B}$ outputs $\mu' = 0$ to indicate that it believes that it was given $(A,B,C,Z) = (g^a, g^b, g^c, e(g,g)^{abc})$ by the challenger $\mathcal{C}$. Otherwise, it will output $\mu' = 1$ which means that it was given a random element Z.

To finalize the security demonstration of the proposed scheme, we estimate the advantage of the simulator $\mathcal{B}$.

When $\mu = 1$, the ciphertext is a random element from the adversary $\mathcal{A}$ point of view, and it gains no information about γ. Therefore, we have $Pr[\gamma' \neq \gamma \mid \mu' = 1] = \frac{1}{2}$. Since $\mathcal{B}$ guesses $\mu' = 1$ when $\gamma' \neq \gamma$, then $Pr[\mu' \neq \mu \mid \mu = 1] = \frac{1}{2}$.

When $\mu = 0$, $\mathcal{A}$ has ε advantage. Hence, by definition we have $Pr[\gamma' = \gamma \mid \mu' = 0] = \varepsilon + \frac{1}{2}$. Since $\mathcal{B}$ guesses $\mu' = 0$ when $\gamma' = \gamma$ then $Pr[\mu' = \mu \mid \mu = 0] = \varepsilon + \frac{1}{2}$. Finally, the overall advantage of $\mathcal{B}$ on the security game is the following:

$$\begin{aligned} Adv_{\mathcal{B}} &= \frac{1}{2}[Pr[\mu' \neq \mu \mid \mu = 1]] + \frac{1}{2}[Pr[\mu' = \mu \mid \mu = 0]] - \frac{1}{2} \\ &= \frac{1}{2}\frac{1}{2} + \frac{1}{2}(\epsilon + \frac{1}{2}) - \frac{1}{2} \\ &= \frac{\epsilon}{2} \end{aligned}$$

Therefore, as ϵ is assumed to be non negligible, $\frac{\epsilon}{2}$ also is non negligible, that proves the Theorem 1.

5 Proposed blockchain-based privacy preserving system

In our proposal, the management of access-control messages is done with a blockchain, which is a decentralised solution that overcomes a single point of trust failure. The fine-grained access control and the security of the storage are achieved thanks to a new scheme of cipher-policy attribute-based encryption (FCCP-ABE). In this section, we will describe the design based on blockchain paradigm and the proposed system framework with a security analysis.

5.1 Blockchain access-authorization record (Blockchain A2R)

The Blockchain is used as a distributed, persistent and tamper-proof database to manage the access control messages. Furthermore, one of the advantages of using blockchain is to give a solution for access right revocation. Before describing our system, we will present our blockchain access-authorization record. For our blockchain, we define a pseudo cryptocurrency as digital assets idx that represents an index to identify a record on the Cloud. Its generation is done by calculating the hash of a bit sequence. For our system, $idx = Hash(CT_{one})$,Where CT_{one} is the intermediate cypher-text seen in 4.1. We also define the following transactions:

1. *idxGenTrans(idx,@st, @src,@dst)*: is the source that generates idx objects. Once idx value is computed by a proxy that has a blockchain address @src, the proxy broadcasts this transaction to transfer the idx to the $data - owner$ account that has a blockchain address $@dst$. The proxy also registers a blockchain address $@st$ on the Cloud data storage.

   token(idx,@st,@rq, @do): is a data structure of credentials to specify an authorization allowed by $@do$ blockchain address owner for the $@rq$ address owner to access a data stored in $@st$ (blockchain address of the storage provider) and identified by idx.

2. *grantTrans(token(idx,@st,@rq, @do), @src,@dst)*: This transaction is used to transfer the $token$ from the blockchain account of one actor to another. In our system, the credential is generated by the data owner and then transferred to the requester account. The requester sends it to the storage provider that will return it to the data owner. This process ensures that each credential is unique and not duplicated.

Figure 2 shows the different interactions within the blockchain. The generation of the idx object is performed by the proxy, called here the Cloudlet or edge, when transferring the encrypted data to the Cloud. The Cloudlet computes $idx = Hash(CT_{one})$ and broadcasts the transaction *idxGenTrans(idx,@st, @src,@dst)*.

5.2 System model

Our proposed system consists of six roles. Figure 3 shows this system model in the context of e-Health:

1. *Data-owner (DO)*: Generates data with his own devices and stores them in the Cloud. The data owner is the only one who has the right to grant access to his data.

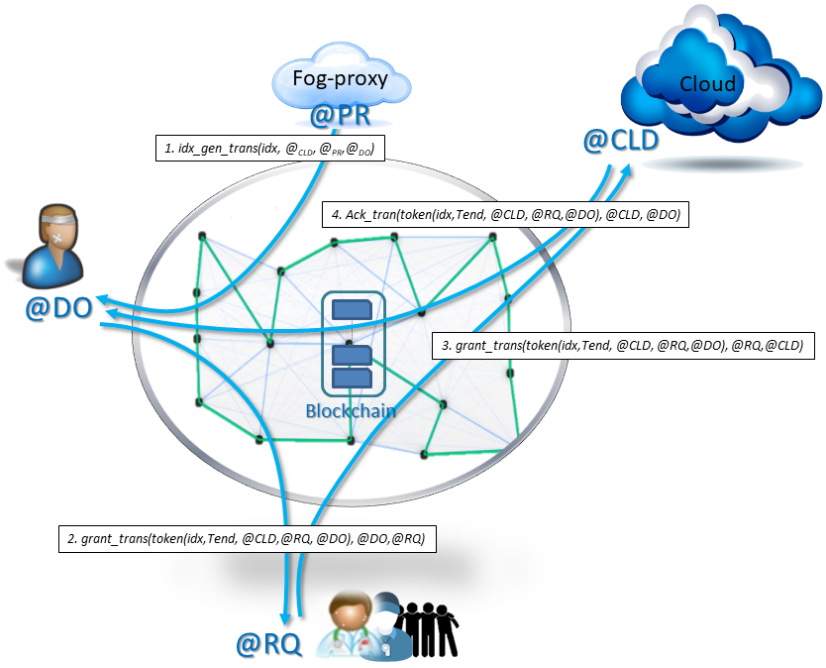


Figure 2: Interactions in the blockchain access-authorization record

2. *Fog-proxy (PR)*: It is the device/gateway/Cloudlet that can be located in the edge network and involved in the encryption process. This proxy is trusted only for performing protocols correctly.

   The proxy executes the *EncryptUncons(CT_{one},τ,Pk)* algorithm without being able to learn any part of the encrypted data.

3. *Data-requester (RQ)*: Is the data consumer who can be a physician or any other medical practitioner that requests access to the personal data of the data owner. To prove the identity of the data requester, we use a PKI infrastructure.

4. *Blockchain-A2R ($BA2R$)*: is the trusted decentralized authority used to ensure the verification and validation of the messages exchanged in untrusted network.

5. *Storage-provider (CLD)*: is instantiated by a Cloud storage service provider. This actor can only check if an anonymous requester can provide evidence, which is allowed by DO to access data.

6. *Data-sources (DS)*: are the data producer devices (sensors or any health devices used to collect measurements). In our system, DS is a resource-constrained device.

   Note that we need also an entity to verify the identity and the attributes of the requester. It can be a PKI infrastructure with a trusted authority. For example, in France, this trusted authority is the French digital health agency (ASIP Santé) that maintains a directory of the health professionals.

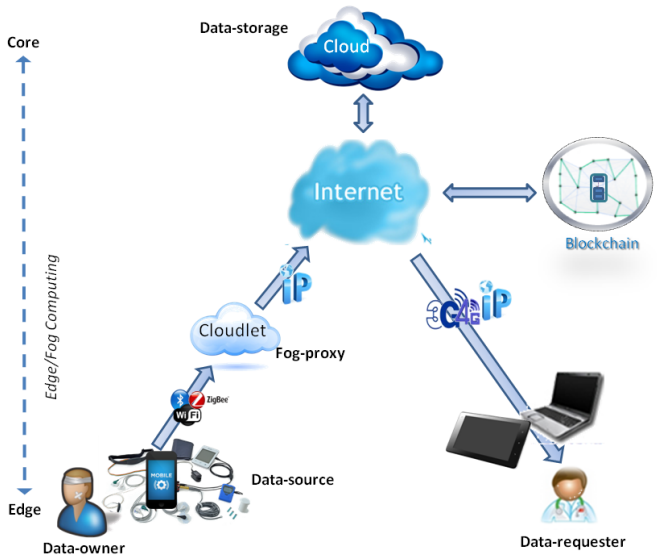


Figure 3: Our system architecture in e-Health context

5.2.1 Assumptions

Before detailing our proposed solution, we make the following assumptions:

1. We use perfect symmetric-encryption primitives $SEncr(M, K)$. It means that to obtain M from $SEncr(M, K)$, we must know K.
2. Physical attacks carried out on the IoT devices (Data source) to get the stored secret keys are out of scope of this proposal.
3. There is sufficiently a large number of honest nodes in the network to maintain the blockchain and to ensure that it is tamper free.
4. Each actor X of our system maintains a blockchain address noted $@X$ used to broadcast transactions within the blockchain.
5. The user manages her/his blockchain and PKI keys in a secure manner.
6. Common PKI infrastructure is used to identify and authenticate the actors when needed in our protocol.

5.3 Blockchain and attribute-based privacy preserving protocol

Our system is a data-owner centric oriented, used to protect privacy and empower the data owner. The blockchain is used as a distributed, persistent and tamper-proof database for access control of management messages. FCCP-ABE ensures a fine-grained data access control that can be implemented in resource-constrained devices. The following detailed phases of the protocol are illustrated in Figure 4 and Figure 5.

- **Phase 1, *System initialization***: During this step, a security parameter λ and an attribute universe are chosen, and the procedure $Setup(\lambda)$ is executed to generate public and

private parameters of FCCP-ABE (Msk, Pk). In addition, the *Data-source* devices (DS) are provisioned with symmetric key Ks and *Fog-proxy* with Pk.

- **Phase 2, *Data recording*** (See Figure 4): The *Data-source* (i.e., the resource-constrained device), noted DS, encrypts the data:

  $$EncryptCons(data, Pk, Ks) \rightarrow CT_{one}$$

  And transfers CT_{one} to the *Fog-proxy (PR)* and C_2 parameter to the *Data-Owner*. Once received, (PR) executes

  $$EncryptUncons(CT_{one}, \tau, Pk) \rightarrow CT_{two}$$

  Then computes $idx = Hash(CT_{one})$ and stores the results in *Storage-provider* (CLD). At the same time, PR broadcasts the *idxGenTrans(idx,@CLD, @PR,@DO)* transaction.

- **Phase 3, *Access grant***: When a user (i.e., the *Data-requester* noted RQ) requests data from the $Data - owner$ (DO), he first authenticates to the DO himself and his attributes set S using a PKI or any other authentication technique. Once done, DO executes $KeyGen$ algorithm with corresponding parameters:

  $$KeyGen(Msk, C_2, S) \rightarrow Sk$$

  And sends this secret key to RQ securely. At the same time, DO generates the $token(idx, @CLD, @RQ), @DO, @RQ)$ and broadcasts the $grantTrans$ transaction:

  $$grantTrans(token(idx, @CLD, @RQ, @DO), @DO, @RQ)$$

  When this transaction is approved by the blockchain, it means that DO authorizes RQ to access the data identified by idx and stored in CLD.

- **Phase 4, *Data access***: When, RQ receives the authorization to access the data (*as in Phase 3*), it broadcasts a $grantTrans(token, @RQ, @CLD)$ transaction to transfer the $token(idx, @CLD, @RQ, @DO)$ to the *@CLD* blockchain address. The Cloud can then verify that the $@RQ$ address owner is a legitimate entity that is authorized to access data identified by idx, thanks to the property of blockchain. Finally, after that RQ has proved that it has the secret key related to the $@RQ$ blockchain address (with a simple nonce-challenge protocol, not detailed here), the Cloud sends the ciphertext CT_{two} to RQ and broadcasts *grantTrans* transaction in order to return the $token$ to the $@DO$ and to inform the DO that its data has been accessed. Finally, RQ uses its secret key Ks to retrieve the data.
  Figure 5 shows the exchange protocol of the phases 3 and 4.

6 Theoretical and experimental analysis

In this section, we provide a brief analysis and a discussion about the security of our solution followed by an experimental analysis of the CP-ABE on different platforms and performance evaluation of the proposed FCCP-ABE.

Figure 4: Data recording in Cloud step

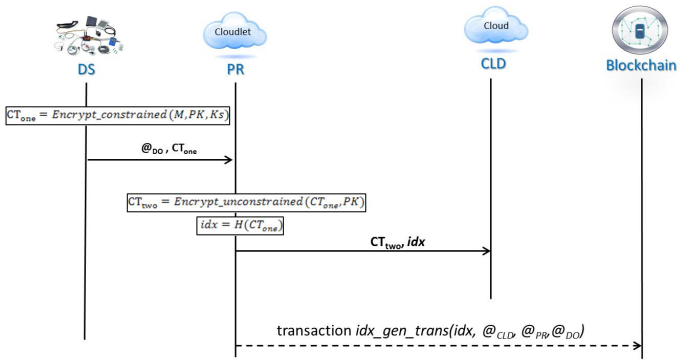
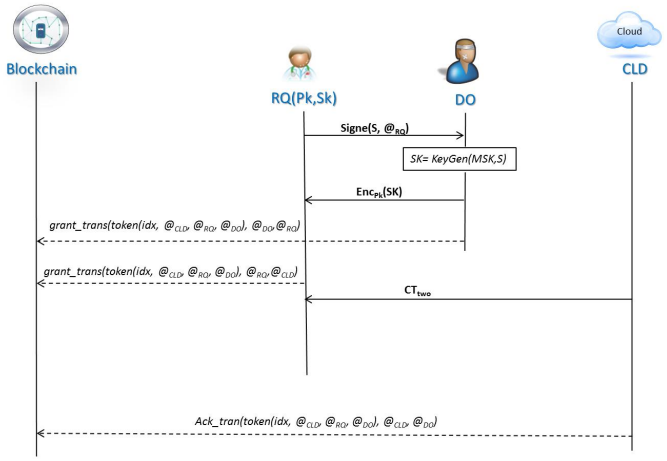


Figure 5: Grant and access steps



6.1 Theoretical Security and privacy analysis

Based on the security intuition of FCCP-ABE, in order to recover a secret message, an adversary who eavesdrops the communication between the data-source device and the Fog-proxy node, must calculate or guess the symmetric secret key Ks. This is not possible due to assumptions 1 and 2. The other possibility is to recover the $e(g,g)^{\alpha t/\beta}$ or the random value t. This is not possible in a polynomial time thanks to the random values used to generate users' secret keys Sk and to encrypted message Ct_{one}. The same reasoning can be applied to the adversary who eavesdrops all the other communications in the system.

Furthermore, collusion attacks did not help since we generate a random value to randomize each user's private key, as proposed by Bethencourt et al. (2007) and formally proved previously. For the same raison, it is important also to notice that the Cloud storage provider as well as the Fog-proxy cannot recover the original secret message. That is because the Cloud is not involved in the data encryption/decryption process and because the partial

part of encryption that the Fog devices compute is not sufficient to recover the value of $e(g,g)^{\alpha t/\beta}$ or t.

Secondly, the proposed system is based on the blockchain paradigm to ensure a stand-alone, self-verification and validation of the access control events. Hence, there is no centralized trusted authority, which overcomes a single point of trust failure. The pseudonymous property of transactions in the blockchain allows to preserve the privacy of the users. The anonymity in the blockchain is based on the fact that users can create any number of anonymous addresses Nakamoto (2008). Our blockchain-based design protects against adversaries that want to compromise the nodes of the system. This requirement is guaranteed thanks to the digitally signed transactions, which ensures that an adversary cannot forge a control message and cannot impersonate a legitimate user. Therefore, based on the decentralized verification and validation process of the blockchain, the only way to corrupt the network is to gain control over the majority of the network resources.

6.2 Experimental analysis

To validate our protocol, we conducted experiments using Raspberry Pi platforms as constrained devices, and a workstation acting as a Fog computing node such as Cloudlet. The experimental simulation of the ABE scheme is done using the pairing-based cryptography library (PBC-Library) Lynn (2007). The workstation runs under 64-bits Ubuntu 16.04 LTS, with Intel(R) Core (TM) i5-4590s 3.00GHz CPU and 8GB RAM. The Raspberry Pi 3 Model B runs a Raspbian operating system, with 1.2GHz 64-bit quad-core ARMv8 and 1GB RAM. To achieve a 128-bit security level, we slightly modify the original PBC-Library Type-A pairing parameters to use a 256-bit elliptic curve group based on the supersingular curve $y^2 = x^3 + x$ over 1536-bit finite field. The number of attributes is $N = \{5, 10, 20, 30, 40\}$. We consider this range to be representative of the real-world applications. To avoid errors, the experimental results are the means of 10 trials.

6.2.1 CP-ABE performance evaluation on different platforms

Regarding the mobile health monitoring use case, we analyse the encryption time because it is the most resourceful operation performed by the data-owner constrained devices. To illustrate the challenge of implementing ABE within resource-constrained devices compared with unconstrained devices, we simulate the original encryption algorithm of CP-ABE scheme Bethencourt et al. (2007) with the number of attributes set from $N = \{5, 10, 20, 30, 40\}$. The results are given in Figure 6a.

Figure 6b shows the measured execution time of the significant computational operations in $\mathbb{G}_0$ and $\mathbb{G}_T$ (see Table 2).

All these experiments are performed both on constrained and unconstrained devices. As we can expect, the cryptographic operations executed in a Raspberry Pi are significantly slower than their execution in a workstation. These experimental results motivate our choice for the outsourcing model to implement the ABE scheme in resource-unconstrained devices.

6.2.2 Performance comparison

In order to demonstrate the validity of the proposed FCCP-ABE, we compare its encryption performance on constrained devices with a selection of the most popular state-of-art models: Asim et al. (2014), Bethencourt et al. (2007), Zhou & Huang (2012) and Zhang et al. (2018).

As depicted in Figure 7, we can notice that our scheme execution time is constant and independent from the number of attributes. In addition, when compared with existing

Table 2 List of the significant computational operations in $\mathbb{G}_0$ and $\mathbb{G}_T$

Abbreviations	**Meanings**
Expo G0	Exponentiation in $\mathbb{G}_0$
Expo GT	Exponentiation in $\mathbb{G}_T$
Mul G0	Multiplication in $\mathbb{G}_0$
Mul GT	Multiplication in $\mathbb{G}_T$
Rand G0	Random generation in $\mathbb{G}_0$
Rand GT	Random generation in $\mathbb{G}_T$
Pairing	Pairing map $e(\mathbb{G}_0, \mathbb{G}_0)$

solutions Bethencourt et al. (2007) and Asim et al. (2014), FCCP-ABE is the most efficient in the context of constrained devices.

Our model is slightly better than the one proposed by Zhang et al. (2018) in terms of execution time. However, the design proposed in Zhang et al. (2018) involves two message exchanges between the IoT device and the Fog node, while our scheme needs only to transfer CT_{one} from the IoT device to the Fog (i.e., proxy) node. Furthermore, as shown in table 3, during the encryption process, the ciphertext length generated by the IoT device in Zhang et al. (2018), depends on the number of attributes used in the access structure n, while in our contribution, the length of the ciphertext generated by IoT devices is independent of the attributes number. This leads our proposal to use less memory resources and consumes less energy for data transmission.

Table 3 Ciphertext-size comparison between FCCP-ABE and Zhang et al. (2018) proposed model

Scheme	**Size of the ciphertext**
Zhang et al.	$\mid \tau \mid + \mid SEncr(M, Ks) \mid +2 \mid \mathbb{G}_0 \mid + \mid \mathbb{G}_T \mid$
FCCP-ABE	$\mid \tau \mid + \mid SEncr(M, Ks) \mid +(3+n) \mid \mathbb{G}_0 \mid + \mid \mathbb{G}_T \mid$

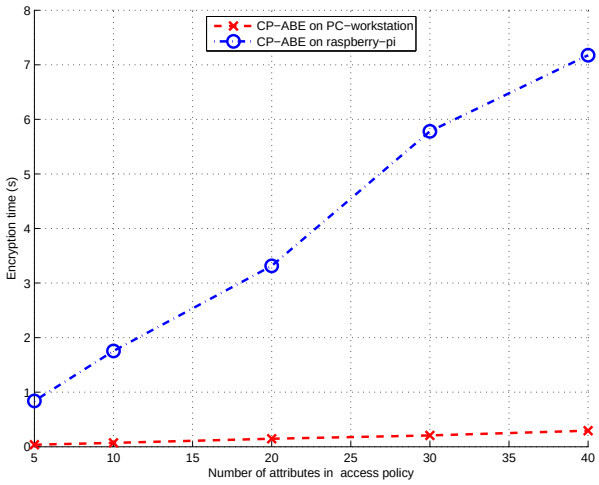
$\mid * \mid$*: Bit length of element in* $*$*.* n*: Number of attributes in leaf nodes of the access structure.*

7 Conclusion

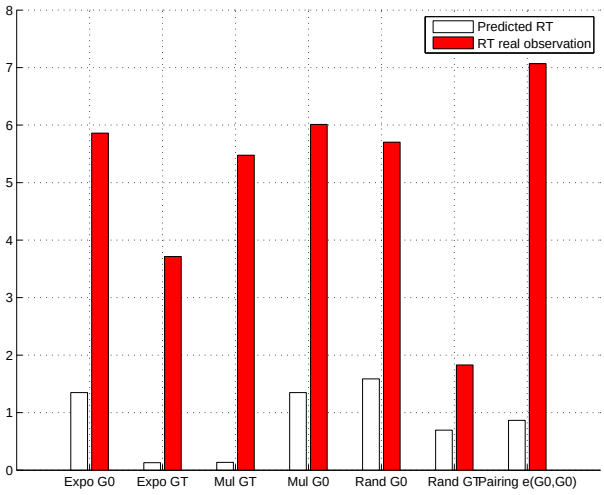
In this paper, we proposed a new user-centric oriented data-privacy model in the context of IoT and cloud. This model is based on a new efficient attribute-based encryption scheme with encryption outsourcing capability on edge/Fog computing, and on a Blockchain-based message exchange. The attribute-based encryption allows to secure personnel data and the Blockchain-based message exchange allows each actor to be able to communicate with each other without requiring a trusted third party, and avoiding a single point of failure. Our proposed blockchain-based privacy preserving design is based on the well-tested technology used in cryptocurrencies, such as Bitcoin, which provides an implementations of the decentralized and secure exchange of information. The formal security proof shows that the proposed FCCP-ABE scheme is secure in respect to the DBDH assumption, and the experimental analysis indicates that the proposed scheme is more efficient than existing

Figure 6: Comparison between the execution-time on PC-workstation and on single-board computers plateforme (raspberry Pi)

(a) Original CP-ABE encryption-time on workstation and on raspberry-pi

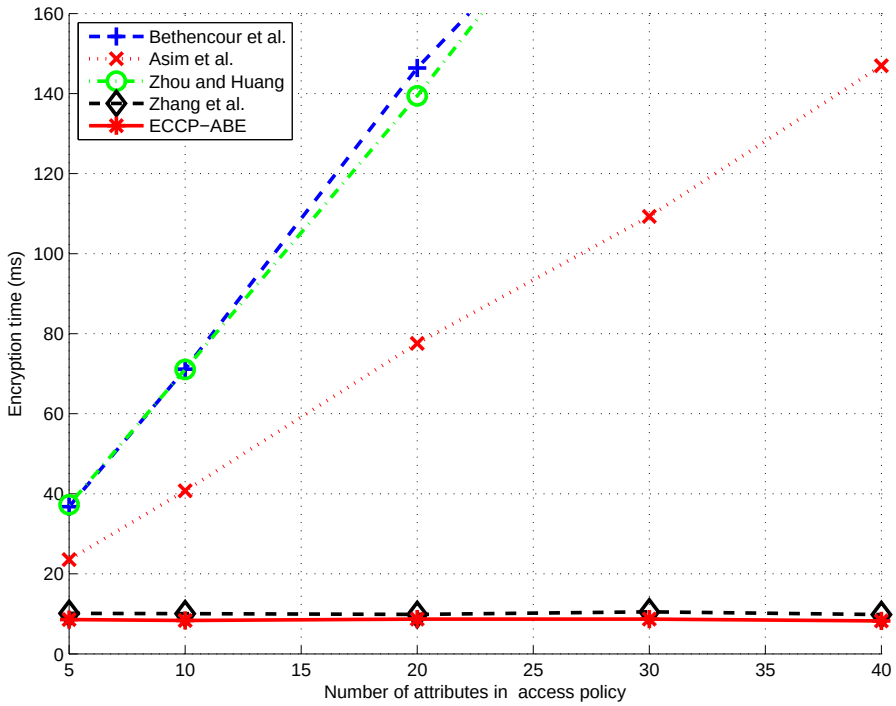


(b) Execution-time of cyclic group operations on workstation and on raspberry-pi



solutions. In this work, we consider the healthcare monitoring as a use case study by focusing only on the encryption process as a critical process. For the other use cases, the proposed scheme can be easily enhanced by outsourcing the decryption process. This is left for a future development.

Figure 7: Encryption performance comparison of FCCP-ABE with other popular schemes



References

Ambrosin, M., Anzanpour, A., Conti, M., Dargahi, T., Moosavi, S. R., Rahmani, A. M. & Liljeberg, P. (2016), On the feasibility of attribute-based encryption on internet of things devices, Vol. 36, pp. 25–35.

Ambrosin, M., Conti, M. & Dargahi, T. (2015), On the Feasibility of Attribute-Based Encryption on Smartphone Devices, *in* 'Proceedings of the 2015 Workshop on IoT Challenges in Mobile and Industrial Systems', IoT-Sys '15, ACM, New York, NY, USA, pp. 49–54.

Anjum, A., Sporny, M. & Sill, A. (2017), Blockchain standards for compliance and trust, *in* 'IEEE Cloud Computing', Vol. 4, pp. 84–90.

Asim, M., Petkovic, M. & Ignatenko, T. (2014), Attribute-based encryption with encryption and decryption outsourcing, *in* 'Proceedings of the 19th Conference on Innovations in Clouds, Internet and Networks', pp. 21–28.

Azaria, A., Ekblaw, A., Vieira, T. & Lippman, A. (2016), MedRec: Using Blockchain for Medical Data Access and Permission Management, *in* '2016 2nd International Conference on Open and Big Data (OBD)', pp. 25–30.

Banerjee, M., Lee, J. & Choo, K.-K. R. (2017), A blockchain future to internet of things security: A position paper, *in* 'Digital Communications and Networks'.

Bellare, M. (1999), Practice-oriented provable-security, *in* 'Lectures on Data Security Lecture Notes in Computer Science', Vol. 1396, Springer Verlag, Berlin, p. 1–15.

Benaloh, J. & Leichter, J. (1990), Generalized secret sharing and monotone functions, *in* 'Proceedings on Advances in Cryptology', CRYPTO '88, Springer-Verlag New York, Inc., New York, NY, USA, pp. 27–35.

Bethencourt, J., Sahai, A. & Waters, B. (2007), Ciphertext-Policy Attribute-Based Encryption, *in* 'Proceedings of the 2007 IEEE Symposium on Security and Privacy', SP '07, IEEE Computer Society, Washington, DC, USA, pp. 321–334.

Bonomi, F., Milito, R., Zhu, J. & Addepalli, S. (2012), Fog computing and its role in the internet of things, *in* 'Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing', MCC '12, ACM, New York, NY, USA, pp. 13–16.

Chiuchisan, I., Chiuchisan, I. & Dimian, M. (2015), Internet of things for e-health: An approach to medical applications, *in* '2015 International Workshop on Computational Intelligence for Multimedia Understanding (IWCIM)', pp. 1–5.

Chowdhury, R., Ould-Slimane, H., Talhi, C. & Cheriet, M. (2017), Attribute-based encryption for preserving smart home data privacy, *in* M. Mokhtari, B. Abdulrazak & H. Aloulou, eds, 'Enhanced Quality of Life and Smart Living', Springer International Publishing, Cham, pp. 185–197.

De, S. J. & Ruj, S. (2015), Decentralized access control on data in the cloud with fast encryption and outsourced decryption, *in* '2015 IEEE Global Communications Conference (GLOBECOM)', pp. 1–6.

Dinh Thai, H., Lee, C., Niyato, D. & Wang, P. (2013), A survey of mobile cloud computing: Architecture, applications, and approaches, *in* 'Wireless Communications and Mobile Computing', Vol. 13.

El-Barbary, A. E. H. G., El-Sayed, L. A. A., Aly, H. H. & El-Derini, M. N. (2015), A cloudlet architecture using mobile devices, *in* '2015 IEEE/ACS 12th International Conference of Computer Systems and Applications (AICCSA)', pp. 1–8.

Gope, P. & Hwang, T. (2016), BSN-Care: A Secure IoT-Based Modern Healthcare System Using Body Sensor Network, *in* 'IEEE Sensors Journal', Vol. 16.

Goyal, V., Pandey, O., Sahai, A. & Waters, B. (2006), Attribute-based Encryption for Fine-grained Access Control of Encrypted Data, *in* 'Proceedings of the 13th ACM Conference on Computer and Communications Security', CCS '06, ACM, New York, NY, USA.

Guo, F., Mu, Y., Susilo, W., Wong, D. S. & Varadharajan, V. (2014), CP-ABE With Constant-Size Keys for Lightweight Devices, *in* 'IEEE Transactions on Information Forensics and Security', Vol. 9, pp. 763–771.

Hashemi, S. H., Faghri, F., Rausch, P. & Campbell, R. H. (2016), World of empowered iot users, *in* '2016 IEEE First International Conference on Internet-of-Things Design and Implementation (IoTDI)', pp. 13–24.

Hemalatha, S. R. & Manickachezian (2014), Security Strength of RSA and Attribute Based Encryption for Data Security in Cloud Computing, *in* 'International Journal of Innovative Research in Computer and Communication Engineering'.

Kocabas, O., Soyata, T. & Aktas, M. K. (2016), Emerging security mechanisms for medical cyber physical systems, *in* 'IEEE/ACM transactions on computational biology and bioinformatics / IEEE, ACM', Vol. 13, pp. 401–416.

Li, M., Yu, S., Ren, K. & Lou, W. (2010), Securing Personal Health Records in Cloud Computing: Patient-Centric and Fine-Grained Data Access Control in Multi-owner Settings, *in* S. Jajodia & J. Zhou, eds, 'Security and Privacy in Communication Networks', Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, Springer Berlin Heidelberg, pp. 89–106. DOI: 10.1007/978-3-642-16161-2_6.

Lin, H., Shao, J., Zhang, C. & Fang, Y. (2013), CAM: Cloud-Assisted Privacy Preserving Mobile Health Monitoring, *in* 'IEEE Transactions on Information Forensics and Security', Vol. 8, pp. 985–997.

Lynn, B. (2007), On the implementation of pairing-based cryptosystems, PhD thesis, Stanford University.

Malarchelvi, P. S. K., Manikandasaran, S. & Arockiam, L. (2019), 'Moncrypt: A technique to ensure the confidentiality of outsourced data in cloud storage', *International Journal of Information and Computer Security* **11**(1), 1.

Mettler, M. (2016), Blockchain technology in healthcare: The revolution starts here, *in* '2016 IEEE 18th International Conference on e-Health Networking, Applications and Services (Healthcom)', pp. 1–3.

Nakamoto, S. (2008), 'Bitcoin: A peer-to-peer electronic cash system," http://bitcoin.org/bitcoin.pdf'.

Namasudra, S. (2017), 'An improved attribute-based encryption technique towards the data security in cloud computing', p. e4364.

Ouada, F. S., Omar, M., Bouabdallah, A. & Tari, A. (2016), 'Lightweight identity-based authentication protocol for wireless sensor networks', *International Journal of Information and Computer Security* **8**(2), 121.

Ouaddah, A., Elkalam, A. A. & Ouahman, A. A. (2017), Towards a Novel Privacy-Preserving Access Control Model Based on Blockchain Technology in IoT, *in* 'Europe and MENA Cooperation Advances in Information and Communication Technologies', Springer, Cham, pp. 523–533. DOI: 10.1007/978-3-319-46568-5_53.

Ould-Yahia, Y., Bouzefrane, S. & Boucheneb, H. (2018), Towards privacy and ownership preserving of outsourced health data in iot-cloud context, *in* '2018 International Symposium on Programming and Systems (ISPS)', pp. 1–6.

Premsankar, G., Francesco, M. D. & Taleb, T. (2018), Edge computing for the internet of things: A case study, *in* 'IEEE Internet of Things Journal', Vol. 5, pp. 1275–1284.

Pussewalage, H. S. G. & Oleshchuk, V. (2016), A patient-centric attribute based access control scheme for secure sharing of personal health records using cloud computing, *in* '2016 IEEE 2nd International Conference on Collaboration and Internet Computing (CIC)', pp. 46–53.

Sahai, A. & Waters, B. (2005), Fuzzy identity-based encryption, *in* 'SpringerLink', Springer, Berlin, Heidelberg, pp. 457–473.

Shamir, A. (1979), 'How to share a secret', *Commun. ACM* **22**(11), 612–613.

Sukhodolskiy, I. & Zapechnikov, S. (2018), A blockchain-based access control system for cloud storage, *in* '2018 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus)', pp. 1575–1578.

Touati, L., Challal, Y. & Bouabdallah, A. (2014), C-cp-abe: Cooperative ciphertext policy attribute-based encryption for the internet of things, *in* '2014 International Conference on Advanced Networking Distributed Systems and Applications', pp. 64–69.

Wang, X., Zhang, J., Schooler, E. M. & Ion, M. (2014), Performance evaluation of Attribute-Based Encryption: Toward data privacy in the IoT, *in* '2014 IEEE International Conference on Communications (ICC)', pp. 725–730.

Waters, B. (2011), Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization, *in* 'International Workshop on Public Key Cryptography', Springer, Berlin, Heidelberg, pp. 53–70.

Yi, S., Li, C. & Li, Q. (2015), A survey of fog computing: Concepts, applications and issues, *in* 'Proceedings of the 2015 Workshop on Mobile Big Data', Mobidata '15, ACM, New York, NY, USA, pp. 37–42.

Zhang, P., Chen, Z., Liu, J. K., Liang, K. & Liu, H. (2018), An efficient access control scheme with outsourcing capability and attribute update for fog computing, *in* 'Future Generation Computer Systems', Vol. 78, pp. 753 – 762.

Zhang, R. & Liu, L. (2010), Security Models and Requirements for Healthcare Application Clouds, *in* '2010 IEEE 3rd International Conference on Cloud Computing', pp. 268–275.

Zhou, Z. & Huang, D. (2012), Efficient and secure data storage operations for mobile cloud computing, *in* '2012 8th international conference on network and service management (cnsm) and 2012 workshop on systems virtualiztion management (svm)', pp. 37–45.

Zyskind, G., Nathan, O. & Pentland, A. . (2015), Decentralizing Privacy: Using Blockchain to Protect Personal Data, *in* '2015 IEEE Security and Privacy Workshops', pp. 180–184.