IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

SPECIAL SECTION ON SECURITY AND PRIVACY IN EMERGING DECENTRALIZED COMMUNICATION ENVIRONMENTS

Received May 25, 2019, accepted June 18, 2019, date of publication July 2, 2019, date of current version July 17, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2926354

Sensitive and Energetic IoT Access Control for Managing Cloud Electronic Health Records

KHALED RIAD[1,2], RAFIK HAMZA[1,3], AND HONGYANG YAN[1,3]

[1]School of Computer Science, Guangzhou University, Guangzhou 510006, China
[2]Mathematics Department, Faculty of Science, Zagazig University, Zagazig 44519, Egypt
[3]Peng Cheng Laboratory, Shenzhen 518055, China

Corresponding authors: Khaled Riad (khaled.riad@science.zu.edu.eg), Rafik Hamza (rafik.hamza@hotmail.com), and Hongyang Yan (hyang.yan@foxmail.com)

This work was supported by the National Natural Science Foundation of China under Grant 61702125 and Grant 61702126.

ABSTRACT Electronic health records (EHRs) replaced the old paper-based systems to make patient data more accurate, reliable, and more accessible. Yet, the EHRs system requires high transmission cost, energy, and waste of time for both doctors and patients. Furthermore, EHRs security presents a serious issue threatening the patient's privacy. Most of the third-party hosting systems have some issues related to the users' privacy and data security. Hence, it is necessary to restrict the access control policies and develop efficient mechanisms for cloud-based EHRs data. In this paper, a sensitive and energetic access control (SE-AC) mechanism is proposed for managing the cloud-hosted EHRs and providing a fine-grained access control even in critical situations. The proposed mechanism ensures the confidentiality of the patient's data, where only authorized individuals to have permission to be able to edit or review certain of the patient's data. Each EHR data is encrypted by the managing authority before submitting to the cloud storage. The requesting user can get dynamically changing permissions based on authentication and context attributes. In addition, seven major aspects have been quantified to assess the operation of any access control that could be deployed in the Internet-of-Thing (IoT). The security analysis indicates that the SE-AC mechanism is secure and will prevent any unauthorized access. The results show exceptional compatibility and performance with different setups and configuration.

INDEX TERMS Access control, cloud computing, electronic health records, Internet-of-Things, data security.

I. INTRODUCTION

Internet-of-Thing (IoT) is everywhere nowadays, causing several influences on the way of human life. Accordingly, IoT becomes an important topic of discussion between researchers [1]. Nonetheless precisely, IoT can operate and elaborate the human daily life harmoniously. Mainly, IoT describes the techniques of the closely attached systems and tools by ingrained sensing components and other devices. In the last decade, many IoT applications have been presented in different areas including construction and home automation, health and health-care, transportation, production, and environmental monitoring [2].

Nowadays, most of the applications dependent on IoT and cloud computing due to the fact that the features of both of them are complementary. In this regard, IoT exploits the boundless capabilities of the cloud computing environment concerning processing and storage. Accordingly, cloud computing becomes an interesting area and rich with IoT information. There are a lot of reliable cloud service suppliers and without the security keys being saved around the cloud. To ensure users confidentiality in the cloud environment, the most commonly used mechanism is encryption [3]. This technique restricts access to the data upon unauthorized users, allowing only who have the secret keys to decrypt the data [2]. Security issues dominate IoT-based systems, requiring a significant revision of the existing security choices that lead several researchers to evolute modern techniques. One of the current security issues is users access control and guarantee cloud data security. In the following discussions, we highlight some existing contributions about access control mechanisms, data security and privacy in IoT and cloud computing.

Data access is restricted to authorized users in order to prevent unauthorized access to sensitive data or changing data. Obviously, these sensitive data could threat the users'

The associate editor coordinating the review of this manuscript and approving it for publication was Xiaochun Cheng.

privacy in different ways. Whether the users today are inside or outside the organization, a piece of information could be accessible to an authorized third party. This means that users hierarchies should have access depending on their level in the system. These hierarchies allow users to build the labels according to their demands [4]. This feature could be set by the system and gives similar characters, such as the assorted levels of users who have access to information [2].

In the context of IoT health-care applications, a patient should be able to preserve the privacy of his own data. Various information of the patients are constructed and stored in the cloud repository, such as blood pressure, heartbeat, etc., [5]. These data should be kept confidential along with other sensitive information in EHRs such as Social Security Number (SSN), billing info, drug records, and alcohol abuse history. Moreover, there are different requirements that should not be dispensed within access control systems of EHRs. For example, a user of health care must freely layout his own privacy policies and demand to apply it within the scope of the health care system [6]. Healthcare providers (such as general practitioners) should be flexible in order to determine the security of a particular document if required. Therefore, patients should have the right to control their own health records, including granting or denying certain medical practitioners from accessing their own metadata EHRs [7]. Furthermore, patients must be able to have other controls such as delegating their EHRs control to another person under certain conditions (like mental illness), and so on. Accordingly, some researchers proposed various access control models for cloud computing. Most of the proposed mechanisms used the Attribute Based Access Control (ABAC) technique in order to meet the access control requirements in cloud systems.

However, most of the applications cannot adopt Access Control Lists (ACLs) in real time due to the huge number of resources along with their different roles. In such a situation, ACLs appear to be extremely complicated for real-time applications. As a matter of fact, the untrusted systems can introduce a comprehensive composite of personal data of a specific user, especially in cloud storage environments [8]. In specific cases, it is possible that the unauthorized users will gain the access rights same as authorized users.

This could cause a disturbance in the access control like give unauthorized access to some users. For example, the pharmacy manager in a hospital needs only to check the billing information before providing the medicine to the patient. Hence, preserving data privacy is a critical issue which can affect human life undeviatingly. To the best of our knowledge, this issue can be resolved by limiting the uniform access of the data for everyone. In addition, it is essential to keep the privacy of the patients' EHRs during the storage process upon the untrusted cloud servers. Thus, it is necessary to preserve data of the patient using an efficient, dynamic access control policy, and independent of the ACLs. To overcome the aforementioned issues, we propose a new sensitive and energetic access control mechanism, denoted by SE-AC, for cloud-based IoT health-care systems.

Our proposed mechanism consists of four parts: organization central authority, system authorities, cloud storage, and custodian domains. The proposed work has the ability to encrypt the EHRs of each patient before submitting to the cloud storage. The SE-AC distributes the tasks of encryption, token generation, decryption, and so on. In this regard, each authority in the system has its own responsibility for the previous mentioned tasks for the regular patients and users who request access for an EHR. The proposed mechanism is stable and fast due to the fact that no time overhead congestion will be at one authority. For example, during generating the decryption token, each system authority will work as main authority in the current session. This authority will also supply the application servers with the requested ciphertext which contain the requested EHR. Finally, the application server will receive the encrypted EHR data in real time. The proposed mechanism has multiple tasks with different approaches to ensure users' privacy and secure the EHR data. Most of these tasks will be executed in a parallel manner to save time at each stage in the mechanism. Thus, the proposed mechanism is fast with a high level of security. The main contributions are summarized as follows:

- We propose a new access control mechanism (SE-AC) for cloud-based IoT health-care systems.
- SE-AC mechanism guarantees preserving-privacy of the cloud-hosted EHRs using a fast and secure encryption mechanism. The system subsequently can retrieve and decrypt the ciphertext EHRs data in a short time in accordance with the requesting user's identity.
- SE-AC mechanism can dynamically adapt with the granted rights for user role hierarchy. Furthermore, the proposed mechanism give permissions based on the environmental conditions and the requester attributes.
- SE-AC mechanism has the ability to deal with a large number of users belonging to different custodian domains with different roles.
- The proposed mechanism empowers the patients to control their own EHRs data and set self-policies. Once the mechanism set up, the process of authorizing access to the patients EHRs are done without the interference of the patient towards each time there is access request.

Finally, the extensive experimental evaluation of a variety of configurations confirms the effectiveness and efficiency of our SE-AC mechanism.

The rest of this paper is organized as follows: Section II contains some related works and comprehensive discussion. Section III presents the problem formulation and the motivation for an efficient access control mechanism for IoT. Section IV introduces our proposed SE-AC mechanism with its basic four entities. The proposed mechanism implementation is introduced in Section V. The detailed performance

analysis is presented in Section VI. Finally, section VII introduces the conclusion and main results of this paper.

II. RELATED WORK

In this section, we present a comprehensive discussion about some related works. As known, data access security in cloud computing has been considered in many recent works, especially using encryption schemes. The proposed framework in this paper can adapt the granted permissions for each user separately based on the environmental conditions and the requester attributes.

Furthermore, recent state-of-art frameworks investigated the problem of data security taking into consideration receiver corruptions issue. Qin *et al.* in [9] introduced a reversible data hiding technique in encrypted image. Progressive decryption is employed to achieve a full quality of the decrypted image. Comparing to these related works, our proposed mechanism can keep the confidentiality of the cloud-hosted EHR data using the encryption process and homogeneous to the role-based access control. After authenticating the requesting user identity based on the possessed attributes and the custodian domain, the decryption process will be accomplished in a short time.

In recent years, different pieces of research have considered data preserving-privacy mechanisms and their application in IoT using several secure protocols. For instance, Boudia *et al.* [10] propose a secure data aggregation technique for wireless sensor networks. The work provides efficient end-to-end security based on a stateful public key encryption scheme. The solution does not require any bound on the aggregation functions, which illustrate its features compared with other state-of-art mechanisms.

Li *et al.* [11] proposed a new multi-authority access control system based on ciphertext-policy attribute-based encryption. The presented scheme supports any monotone access policy based on the provable security technique in the standard model with an efficient attribute-level user revocation approach including minimum computation cost.

Li et al. in [12] proposed an efficient security system for multi-authority cloud storage systems based on the attribute-based encryption. The proposal designed to make use of two factors protections into an integration, instead of two parts with double encryption processes. As a new technique for access control based on quantum, Zhou *et al.* [5] formalized an encryption scheme and protocol for key distribution in the setting of categorical quantum mechanics.

Data security and users privacy become a serious problem nowadays due to the prompt increase in computer networks activities, especially through IoT devices. Indeed, security and privacy in IoT is a challenging problem versus many aspects like the high computations and complexity [13]. Accordingly, some researchers propose solutions for IoT security through various aspects. Generally, the researchers try to build a centralized privacy-preserving for the storage system that supports both file-level and block-level multi-layers processing. Hamza *et al.* [2] applied a fast cipher mechanism based on chaotic systems for IoT E-healthcare systems, preserving the privacy of patients using a probabilistic encryption scheme.

The rapid deployment of IoT Applications deliver more researches adopting different mechanisms ([3], [14]), especially privacy-preserving techniques for E-healthcare data based on IoT and Cloud systems. For example, Chen *et al.* [15] proposed a security framework for EHR sharing and integration system based on Hybrid clouds. Page *et al.* [16] a preserving privacy system for healthcare monitoring techniques with analytic methods. This work is based on fully homomorphic encryption and runs quickly enough with lower complexity compared to some related works. The presented mechanism permits the extraction of related information for the analysis from patient's data and preserving their privacy. Where the patients could ask to restrict the access to specific records to all physicians in the hospital and maintain access only to their primary doctors.

Indeed, some modern techniques based on traditional cryptography schemes can guarantee data security and privacy for users such as homomorphic encryption algorithms and signatures [17]. Besides traditional cryptography solutions for data access control, there are also some recent contributions to protect and guarantee cloud data access. Several researches [18], [19] have established different solutions in order to secure data access and preserving-privacy for users. For example, Riad *et al.* [20] formalized Attribute Based Access Control (ABAC) and proposed a new access control model. The framework was denoted as Attribute-Rule ABAC (AR-ABAC), and works mainly for cloud computing to meet critical access control requirements in clouds. Furthermore, Riad and Yan [21] have proposed a new Trust-Based Access Control (TB-AC) model, which supports dynamically changing the user's assigned permissions based on its trust level.

III. PROBLEM FORMULATION

In general, the modern IoT access control frameworks are ordinarily referred to as intelligent gate of the IoT systems. The following items listed the access control solutions IoT-based needs.

- An access control solution should overcome the IoT scalability obstacles.
- Simplicity in taking care of access control. As a result of the prevalent of IoT tools that inclusive daily tools like individual mobile phones and also devices. Furthermore, the customers with previous experience should be deeply associated with consent tasks compared to in the past.
- An access control solution should support progressed functions. For example, accessibility legal rights delegation and so on.
- An access control solution should support progressed functions. For example, accessibility legal rights delegation and so on.

The proposed mechanism SE-AC provides great flexibility in addressing the aforementioned issues as compared to

traditional Attribute Based Access Control (ABAC)/ Role Based Access Control (RBAC) systems.

The IoT has distinguished features which can lead to some issues like expanding access control administration for unauthorized users. In this regards, there are comparatively new solutions contain multi-layer mechanisms, introducing reliable gain access control to devices through diversified delegation chains. Indeed, both RBAC [22], [23] and ABAC [24], [25] systems have established with stringent rules, do not scale well, and also both techniques tough to utilize and to update [26]. To this end, the data owner should have three basic cases for patient's control over exchanging his/her own EHRs. These three basic cases are listed as follows.

- **Case I: Approving each Request,** the data owner has to approve each request for each custodian. This rarely happens actually, because it will make the organization act like a cobweb which will introduce a great overhead in the communication process. Furthermore, this could include a vast delay in some critical situations while the data owner should be online and available at any time.
- **Case II: Setting Rules for each Authority** the data owner has to fill consent rules with the intended managing authorization directly. This can be executed with some shortcomings such as setting similar rules for different authorities.
- **Case III: Setting Rules for Central Authority** the data owner has to set the consent rules only once to the central managing authority. In this case, the rules could be little-general and the central authority has the ability to re-direct them to each authority in the managing system. Additionally, the data owner should have domination so that he can allow or deny any permission demands according to his requirements.

Our proposed mechanism deals with **Case III** taking into consideration all the above details.

The main goals of the proposed SE-AC are:

1) The proposed work introduces a time efficient access control, which will keep the privacy of their patients' EHRs and will give the requesting users the right permissions.
2) The proposed SE-AC deals with different kind of users with only one access control. Also, the access policy should change according to the requesting user.
3) The work itself runs numerous experiments under different circumstances and configurations for multiple scenarios to ensure the scalability, efficiency, performance, and stability of the introduced SE-AC mechanism.

IV. SENSITIVE AND ENERGETIC ACCESS CONTROL (SE-AC) MECHANISM

The proposed SE-AC mechanism concentrates on managing access control in a hospital with multiple internal departments, such as emergency department, general surgery, pharmacy, human resources, and so on. For example, those departments should have different authorization permissions to their users (physicians or employers), which will guarantee the EHRs privacy of their patients (data owners). Furthermore, some patients exist with HIV, drug or alcohol abuse records. So, those patients could ask to restrict the access to specific records to all physicians in the hospital and maintain access only to their primary doctors.

Figure 1 shows the proposed SE-AC mechanism in details. The proposed model has four major parts each of them has its own construction in the system as the description below shows.

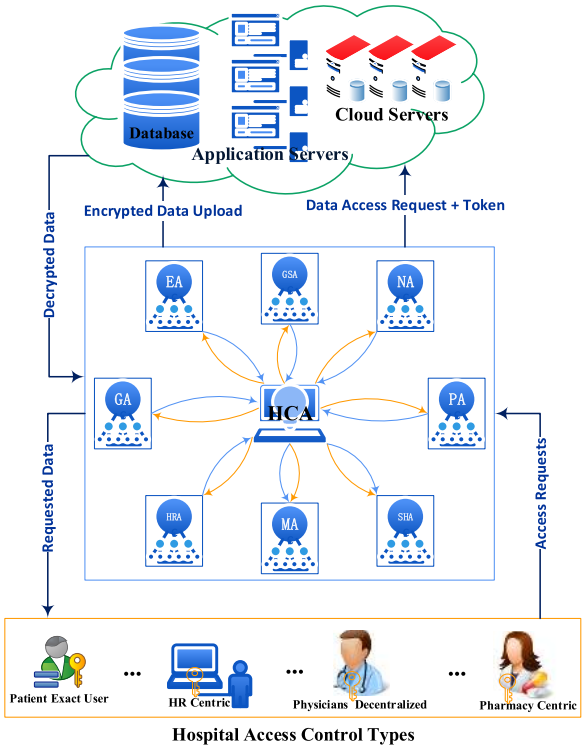

FIGURE 1. SE-AC system model for an organization (hospital).

TABLE 1. The notations considered in our sensitive and energetic mechanism.

Notation	Description
OCA	Organization Central Authority
AID	A unique Authority ID
PK_{AID}	The public key of an authority AID
CID	A unique custodian domain ID
SK_{CID}	The secret key of a custodian domain CID
EHR	Electronic Health Record
DT	Decryption Token
$\mathcal{AC}_{AID}$	Access structure of the managing AID
PID	Patient ID
CT_{AID}^{PID}	Ciphertext from AID to PID
SK_{CID}	Secret key of CID
DT_{CID}	Decryption token for CID

A. Organization Central Authority (OCA)

Here, the authority represents the core managing authority of the hospital organization. In this regards, the main goal is to

balance and distribute the responsibilities of *OCA* to avoid congestion and delays. *OCA* has two tasks:

1) SETUP

First, *OCA* initializes the system authorities with composing the structures. Each authority should be assigned by a unique Authority ID (*AID*). This is an alphanumeric ID that distinguishes those authorities from each other. In addition, each user in a specific custodian domain will assign a unique ID (*CID*). After that, the patient (or data owner) will assign a unique Patient ID (*PID*), which represents the precise patient identity. Accordingly, the *OCA* has to generate two types of keys (master and public keys).

Given a multiplicative group $\mathbb{G}$ and a symmetric paring e from the multiplicative group $\mathbb{G}$ to another multiplicative group $\mathbb{G}_T$, $(e : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T)$. *OCA* randomly chooses three generators $g, g_1, g_2 \in \mathbb{G}$ and $\alpha \in \mathbb{Z}_p^*$, where p is a prime number. *OCA* then computes $g_1 = g^\alpha$. After that, *OCA* selects $q' \in \mathbb{G}$ and $\ell - length$ vector $(q_1, \cdots, q_\ell) \in \mathbb{G}^\ell$. The output is represented in the public parameter $(g, g_1, g_2, q', q_1, \cdots, q_\ell)$ and the master key $(MK = g_2^\alpha)$. The MK is used to generate the secret key for each requesting access user.

2) KEY GENERATION

In this algorithm two types of keys (public and private) are generated. Given $(\mathbb{G}, \mathbb{G}_T, p, \ell)|\ell = |p|$. *OCA* randomly chooses $g \in \mathbb{G}$ and $\alpha \in \mathbb{Z}_p^*$. *OCA* then computes $g_1 = g^\alpha$ and $\jmath = e(g, g_1) = e(g, g)^\alpha$. After that, *OCA* randomly selects $x' \in \mathbb{Z}_p^*$ and a random $\ell - length$ vector $(x_1, \cdots, x_\ell)$, where the elements of that vector are randomly selected from $\mathbb{Z}_p^*$. *OCA* calculates $q' = g^{x'}$ and $\{q_i = g^{x_i}\}_{\forall i \in 1:\ell}$. Finally, an injective hashing $H_0 : \mathbb{G} \times \mathbb{G}_T \rightarrow \{o, 1\}^\ell$ is selected. The public key for each authority identified by *AID* is

$$PK_{AID} = (\jmath = e(g, g_1), q' = g^{x'}, q_1 = g^{x_1}, \cdots, q_\ell = g^{x_\ell}, H_o),$$

and the private key for each user in a specific custodian domain identified by *CID*, is

$$SK_{CID} = (g_1 = g^\alpha, x', x_1, \cdots, x_\ell).$$

B. SYSTEM AUTHORITIES

In this part, we introduce some authorities to avoid addressing many responsibilities toward *OCA*. The authorities can manage the requests of specific custodian domain of users. There are some well-known examples of those authorities such as Emergency (EA), General Surgery (GSA), Neurology (NA), Gynecology (GA), Pharmacy (PA), Human Resources (HRA) and so on. Figure 1 shows these authorities where each authority has three basic responsibilities (services).

1) ENCRYPTED DATA UPLOAD

The main task of each authority is encrypting the EHRs data related to the patients before sending them to the cloud hosting. Each authority *AID* in the system use the encryption scheme as shown in Algorithm 1.

Algorithm 1 Encrypting an EHR

Input: leftmargin = 4mm
- *EHR* ▷ The electronic medical record to be encrypted leftmargin = 6.5mm
  - *PID* ▷ The patient ID under the management of the authority *AID* (The EHR owner)

1: Randomly select $s \in \mathbb{Z}_p^*$ ▷ A random encryption exponent
2: Compute $CT_2 = g^s$ and $CT_3 = \jmath EHR = e(g, g_1)^s EHR = e(g, g)^{\alpha s} EHR$
3: Compute $z = H_0(CT_2, CT_3)$ ▷ In the same manner, compute $z_1, \cdots, z_\ell$ that represent the binary expansion of z, where $z_i \in \{o, 1\}$
4: Let $Z \subseteq 1, \cdots, \ell$ ▷ It the set of all i for which $z_i = 1$
5: Compute $CT_1 = (q' \prod_{i=1}^{\ell} q_i^z)^s$
6: Compute the ciphertext:

$$CT_{AID}^{PID} = (AID.CT_1, PID.CT_2, AID^{PID}.CT3)$$
$$= (AID.(q' \prod_{i=1}^{\ell} q_i^z)^s), PID.g^s,$$
$$AID^{PID}.(e(g, g)^{\alpha s} EHR))$$

Output: The ciphertext CT_{AID}^{PID}

Algorithm 1 takes the EHR data as an input to the encryption scheme. EHR data should be elicited from a patient identified by *PID* under the management of a system authority *AID*. As a result, the algorithm will encrypt the EHR data using a random encryption exponent and based on *PID* and *AID*. Obviously, the output of this algorithm will be the encrypted EHR, and these data can be uploaded to the cloud storage servers including the untrusted ones.

2) DATA ACCESS REQUEST

In this part, we illustrate the following scenario of a data access request. The *OCA* receives an access request from a specific custodian domain user (identified by *CID*). Then, the *OCA* forwards the request to the authority in charge (*AID*) of this custodian domain. Finally, the authority *AID* will execute two operations (Data Access Request and Token-Generation) in a *parallel manner*.

3) TOKEN-GENERATION

Furthermore, the *OCA* will send authentication attributes (*AA*) and context attributes (*CA*) to the designated authority (the key attributes) to proceed with the authorization process. The authentication attributes include user identity, (*CID*) attribute as authenticated, Authentication Strength (*AS*), Role (*R*) which contains a vocabulary representing the duties of that user in the organization, Requesting organization (*RO*), and finally Authentication Time (*AT*).

Additionally, the context attributes include Requested Operation (OP) (permission), Purpose of Use (PoU). As well as patients whose records are requested (PID), which means a patient's identity is correlated. Final context attribute includes the Patient's Rules (PR).

Algorithm 2 Token Generation

Input: leftmargin = 4mm

- $AA = \{CID, AS, R, RO, AT\}$ ▷ The authentication attributes as authenticated leftmargin = 6.5mm
- $CA = \{OP, PoU, PID, PR\}$ ▷ The context attributes
- $\mathcal{AC}_{AID}$ ▷ The authority's access structure

1: **if** $AA \cup CA \vDash \mathcal{AC}_{AID}$ **then** ▷ The authentication and context attributes of the requesting user satisfies $\mathcal{AC}_{AID}$
2: **if** AT *within* RO(*work hours*) **then** ▷ The authentication time is within the work hours of the requesting organization
3: Generate $SK_{CID} = (g_1 = g^{\alpha}, x', x_1, \cdots, x_\ell)$
4: Generate $DT_{CID} = \prod_{i=1}^{\ell} e(g, g)^{\alpha s.PID}$
5: **end if**
6: **else** ▷ DT cannot be generated successfully
7: $DT_{CID} = rand(Hex)$ ▷ DT is a random Hexadecimal value
8: **end if**

Output: The Decryption Token DT_{CID}

Algorithm 2 takes the authentication attributes AA as an input along with the context attributes CA, and the access structure $\mathcal{AC}_{AID}$ of the managing authority. The decryption token is generated using Algorithm 2.

The decryption token is successfully generated if the following points have been achieved. First, establish the authentication and authorization process. Second, the context attribute should be identified to overcome the authority's access structure. Third, the authentication time is within the work hours of the requesting organization. Otherwise, the decryption token will receive a random Hexadecimal value.

In this regards, the requesting user organization is not correlated (there is no prior and known relationship between the hospital and the patient) to the PID. Then, the request will be denied. Hereafter, the authority (AID) will generate a decryption token (Algorithm 2) for the requesting user (CID) based on the authentication attributes and the context attributes. Accordingly, the requesting authority AID will receive the requested ciphertext (containing requested EHR) during generating the decryption token.

4) DECRYPTING THE CIPHERTEXT

The secret key (SK_{CID}) and generated token (DT_{CID}) will be used in decrypting the received ciphertext, which contains the requested EHR ($CT_{AID}^{PID} = AID.CT_1, PID.CT_2, AID^{PID}.CT3$) at the authority itself, as follows.

First, the authority will compute

$$z = H_0(CT_2, CT_3).$$

Then, compute

$$z' = x' + \sum_{i=1}^{\ell} x_i z_i \ mod \ p.$$

If $(CT)^{z'} = CT_1$, then the ciphertext can be decrypted using SK_{CID} and DT_{CID}. Otherwise, the ciphertext cannot be decrypted and give the requested EHR.

C. CLOUD STORAGE

In this part, we illustrate the cloud mechanisms related to our work. As known, cloud storage is composed of some application servers, cloud servers with high capabilities, and database servers. Those servers are used together and managed by the application servers to store the encrypted EHRs. Furthermore, in a case where cloud storage receives a data access request from a specific authority, it will respond with the ciphertext that includes the requested EHR through the application servers. This will safeguard the privacy-preserving of the patients' data.

D. CUSTODIAN DOMAINS

Custodian domain is represented by a great collection of custodians (HR Centric, Physicians Decentralized, Pharmacy Centric, and so on) that can request access to a specific EHR for a patient. As well as the exact user, the patient itself has to set and fill his own rules to OCA. These rules will be distributed to the other systems authorities in accordance with each one. This means that each user in a specific custodian domain will be given a numerical CID. This CID is used to identify the user during its session.

V. IMPLEMENTATION

In this section, we present the results of implementing the SE-AC mechanism based on a private cloud environment using OpenStack [27]. Additionally, there are a virtual set of cloud-hosted instances working as application servers, with extensive hardware configuration (8 VCPUS, 160 GB disk space, and 16384 MB of RAM). Those servers outline the legal channels between the cloud services and the organization's authorized authorities. Overall, the application servers receive the data access requests and forward them to the intended cloud storage servers that store the requested encrypted EHR data. Accordingly, the application servers send the requested encrypted data to the designated authority. The application servers are responsible for storing the encrypted data through each authorized authority on the cloud storage serves.

Another significant cluster in our proposed mechanism includes the organization's central authority and other system authorities. The organization's central authority (OCA) is outlined using the central managing authority in the hospital. In this regards, OCA has been implemented in a

virtual instance with a considerable hardware configuration (4 VCPUS, 80 GB disk space, and 8192 MB of RAM). Taking into consideration that the *OCA* has few tasks and easily avoid being a central point of attack and bottleneck, there is no need to give *OCA* higher hardware capabilities. The other system authorities work as the managing authority for each department in the organization. Each of them is implemented as a virtual instance, with X-large configuration (8 VCPUS, 160 GB disk space, and 16384 MB of RAM). In this part, those authorities should be manufactured with excellent hardware configuration larger than the others, due to the fact they have numerous tasks and computations. These tasks include the time sensitive and executing each task using an instance of huge capabilities. Obviously, these tasks will give an effective mechanism to the proposed work.

The proposed mechanism has multiple trusted domains for requesting users and distinct custodian domains. These domains differ from each other due to the diverse permissions according to the custodian domain role. The user belongs to a specified custodian domain. Accordingly, the user will inherit some of the permissions from this domain. The proposed custodian domains contain the data owners (Patient Exact User) and the requesting users (HR Centric, Physician Decentralized, Pharmacy Centric, and so on). The custodian domains possess a private computer which has the ability to communicate with the SE-AC mechanism. It can pass the attributes and the inherited permissions from the custodian domain to the central organization authority.

VI. PERFORMANCE ANALYSIS

In the proposed mechanism, we introduce the performance analysis in two parts. The experiments investigate seven significant measurements. The main idea is to simulate the different access control models in IoT. It should be mentioned that the figures are generated using OriginPro 2018.[1]

A. DATA OWNER AND STORAGE

In this part, we illustrate the details of the communication overhead and time consumed. Starting with introducing the patient exact user his own attributes and rules to *OCA*, until storing the encrypted EHR on the cloud storage servers. To have an extensive discussion, we present four different measures as described in the following sections.

1) PATIENT EXACT USER WITH *OCA*

First, the patient should register to the system and the *OCA* will assign a unique *PID* to the patient. The patient should adjust and provide his own rules in order to access his own EHRs data. Then, the *OCA* will generate a public key for the registered patient and send it to him accordingly. Figure 2 shows the time consumed in this communication scenario. In this test, we aim to find the elapsed time (in milliseconds). Starting from producing the patient-user a request to join the system, until receiving his own public key from the central authority of the organization.

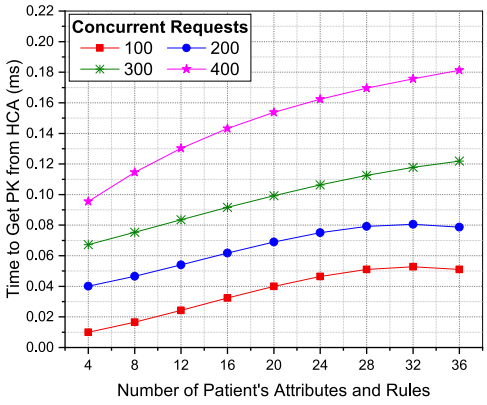

FIGURE 2. The time in milliseconds until the patient exact user receives his own public key from the *OCA*.

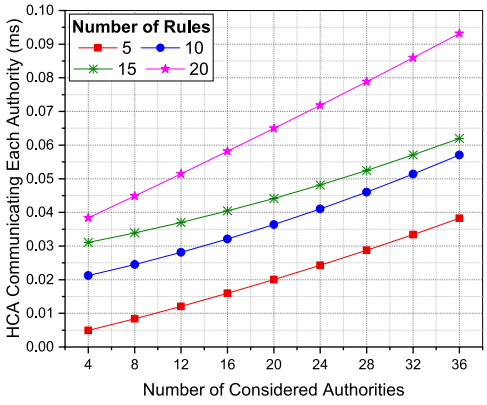

FIGURE 3. The time in milliseconds for sending the patient's rules to each authority from *OCA*.

Figure 2 shows the time elapsed until receiving the patient's public key against the number of patient's attributes and authorization rules. This has been measured according to four different groups (100, 200, 300, and 400) of concurrent requests from different patients to *OCA*. Note that each patient may request a specific authority different from another patient through *OCA* at the same time.

2) *OCA* COMMUNICATION WITH OTHER AUTHORITIES

The central authority has to communicate with each authority in the system and should assign a unique *AID* and a public key for each authority. Furthermore, the patient's rules will be transferred to each authority, due to the fact that each authority is responsible for the process of encrypting the patients EHRs. The time elapsed of these communication steps is shown in Figure 3.

The figure introduces the time elapsed in generating the PK_{AID} for each authority against the number of authorities requesting registration from *OCA* while considering different numbers (5, 10, 15, 20) of rules to be sent for each authority. Obviously, the complexity timing is linearly linked to the number of requesting authorities at the same time.

[1] http://www.OriginLab.com

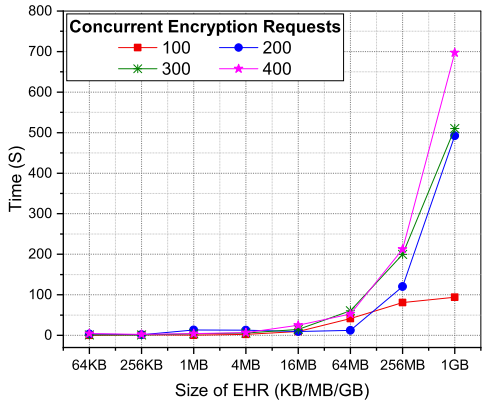

FIGURE 4. The encryption time in seconds against the size of EHR while considering four categories (100, 200, 300, and 400) for the number of concurrent encryptions at the same authority.

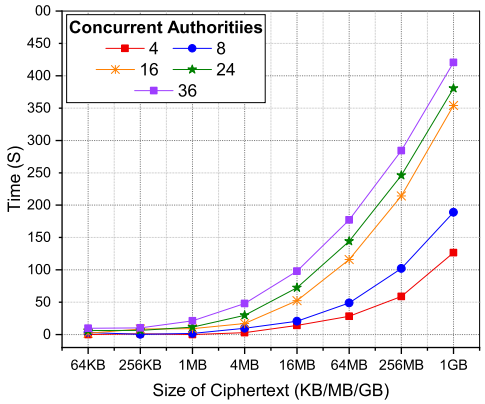

FIGURE 5. The time in seconds elapsed at the cloud storage for storing the ciphertext uploaded by 8, 16, 24, and 36 concurrent authorities based on the size of the ciphertext.

3) ENCRYPTION TIME

The proposed EHR encryption algorithm executes little complex operations to encrypt a specific EHR. Yet, the encryption algorithm overhead is varying based on the EHR data size and the number of concurrent requests (to be encrypted from different users to the same authority at the same time). Figure 4 shows the EHR encryption algorithm considering both EHR size and the number of concurrent encryptions at the same authority at the same time.

The Figure 4 indicates also the time process (in seconds) elapsed from a specific authority to encrypt the EHR data compared to the size of EHR. The results show that consuming 92 seconds for 100 concurrent encryption requests for the same authority while encrypting 1 GB size EHR. Same results with the 400 concurrent encryption request, for the same authority. Where the encryption time is computed 700 seconds with 1 GB EHR data. The timing result is acceptable due to the huge number of concurrent requests. As a result, we can conclude that the proposed mechanism manages to handle all the requests with excellent timing and responses.

4) STORAGE OVERHEAD

In this part, we discuss the storage overhead for the uploading and storing the encrypted data (EHRs) on the cloud storage servers. The storage time in this step is affected by two factors: the ciphertext size and the number of concurrent authorities. In this work, we considered the size of the ciphertext is the same size as the encrypted EHR due to the fact that the encryption algorithm generates ciphertext size equal to original size. Additionally, we tested the mechanism taking into consideration (8, 16, 24, and 36) concurrent authority. These concurrent authorities will upload their ciphertext at the same time. Figure 5 shows the times consumed against the ciphertext size with different concurrent authorities.

B. REQUESTING USER AND DECRYPTION

This part introduces the measurement of communication overhead and time consumed. The process starts from introducing the user belonging to a specific custodian domain his own attributes and certificates to *OCA* until receiving the requested decrypted EHR. Here, we present three different measures.

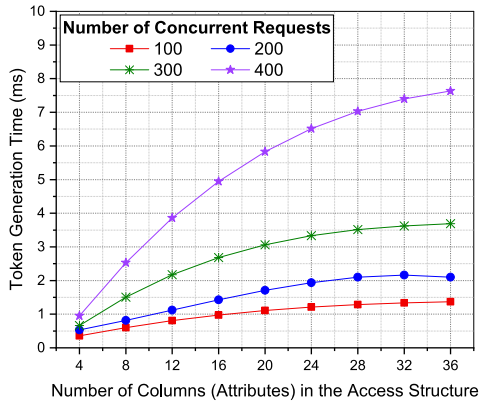

FIGURE 6. The time in seconds elapsed at the authority for generating the decryption token by considering 100, 200, 300, and 400 concurrent token generation request for the same authority, based on the number of columns (ℓ) in the access structure $\mathcal{AC}$.

1) TOKEN GENERATION TIME

When a user requests access from *OCA*, the request will be forwarded to the authority in charge of processing that request. Furthermore, *OCA* will support the authority with the authentication and context attributes. This is very important and key attributes for the authorization process. Consequently, the authority will run Algorithm 2 to produce the decryption token (used for decrypting the ciphertext for the requested EHR). The token generation algorithm is based on the received authentication and context attributes with the ciphertext access structure $\mathcal{AC}_{AID}$. Then, the proposed work will generate SK_{CID} and DT_{CID} for the user with *CID* to access a specific EHR data.

Figure 6 shows the time elapsed in generating the decryption token by the authority in charge of processing. It is based on ℓ, which is the number of columns (attributes) in the ciphertext's access structure while considering multiple

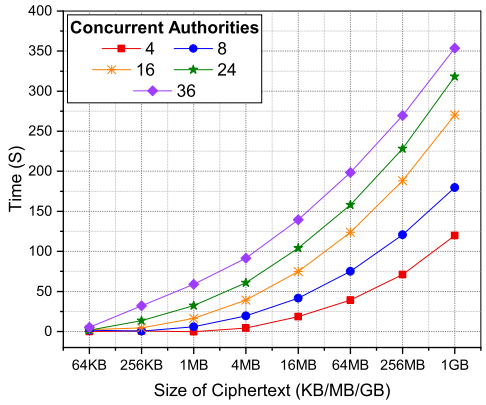

FIGURE 7. The time in seconds elapsed at the cloud storage for storing the ciphertext uploaded by 8, 16, 24, and 36 concurrent authorities based on the size of the ciphertext.

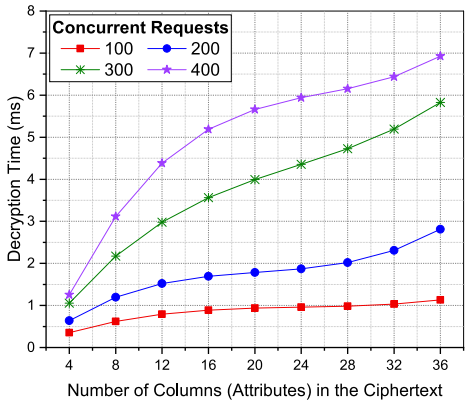

FIGURE 8. The time in seconds elapsed at the authority for the ciphertext decryption by considering 100, 200, 300, and 400 concurrent request from different users for the same authority based on the number of columns (ℓ) in the access structure $\mathcal{AC}$.

(100, 200, 300, and 400) concurrent access requests from different users to the same authority. Note that during the decryption token generation, the authority requests and proceed to receive the ciphertext from the application servers.

2) RECEIVING THE ENCRYPTED CIPHERTEXT

This part is similar to downloading the ciphertext from cloud storage to the requesting authority. Receiving the encrypted data is the contrast of uploading the ciphertext (Section VI-A.4). The authority submits an access request during generating the decryption token to the application servers. Generally, these servers are authoritative of finding the requested ciphertext and sending it back to the requesting authority. A considerable part of this measuring process is based on the throughput of the network itself. Yet, we assume that the network throughput is fixed during our experiments, thus it does not affect the download time.

The time required for downloading the requested ciphertext to the requesting authority is shown in Figure 7. The download time depends on the size of the ciphertext while considering (8, 16, 24, and 36) concurrent authorities requesting ciphertexts from the application servers at the same time.

3) DECRYPTION TIME

In the proposed SE-AC mechanism, the decryption time is independent of neither the considered user's attributes nor the authentication. Note also that context attributes are given by *OCA*. Most of the time required for decrypting the ciphertext has been already achieved during the decryption token generation. As a result, the decryption time slightly depends on the number of columns in the access structure $\mathcal{AC}$ and the number of concurrent decryption requests for the same authority, as shown in Figure 8.

VII. CONCLUSION

In this paper, an efficient access control mechanism (SE-AC) is proposed. The SE-AC builds on the top of a private cloud environment and overcomes the drawbacks of the existing works in the literature. In the proposed mechanism, there are four entities (hospital central authority, system authorities, cloud storage, and custodian domains). The proposed SE-AC mechanism has the ability to process multiple operations in a parallel manner to keep the execution time. The proposed SE-AC has been evaluated through different simulations and the obtained results have demonstrated its efficiency. Several experiments analysis and tests have been performed under varying circumstances and configurations. To ensure the efficiency of the proposed work, we have illustrated seven critical measures in cases study for the access control mechanism. The presented approach ensures consistency of users access and EHR data security with excellent performances. Indeed, the performance analysis demonstrates that the proposed mechanism is efficient and can be implemented under different contexts, especially for IoT health-care systems.

REFERENCES

[1] F. A. Alaba, M. Othman, I. A. T. Hashem, and F. Alotaibi, "Internet of Things security: A survey," *J. Netw. Comput. Appl.*, vol. 88, pp. 10–28, Jun. 2017.
[2] R. Hamza, Z. Yan, K. Muhammad, P. Bellavista, and F. Titouna, "A privacy-preserving cryptosystem for IoT E-healthcare," *Inf. Sci.*, to be published.
[3] T. Song, R. Li, B. Mei, J. Yu, X. Xing, and X. Cheng, "A privacy preserving communication protocol for IoT applications in smart homes," *IEEE Internet Things J.*, vol. 4, no. 6, pp. 1844–1852, Dec. 2017.
[4] L. Wollatz, M. Scott, S. J. Johnston, P. M. Lackie, and S. J. Cox, "Curation of image data for medical research," in *Proc. IEEE 14th Int. Conf. e-Sci. (e-Science)*, Oct./Nov. 2018, pp. 105–113.
[5] L. Zhou, Q. Wang, X. Sun, P. Kulicki, and A. Castiglione, "Quantum technique for access control in cloud computing II: Encryption and key distribution," *J. Netw. Comput. Appl.*, vol. 103, pp. 178–184, Feb. 2018.
[6] B. Martínez-Pérez, I. De La Torre-Díez, and M. López-Coronado, "Privacy and security in mobile health apps: A review and recommendations," *J. Med. Syst.*, vol. 39, no. 1, p. 181, 2015.
[7] B. Yüksel, A. Küpçü, and Ö. Özkasap, "Research issues for privacy and security of electronic health services," *Future Gener. Comput. Syst.*, vol. 68, pp. 1–13, Mar. 2017.
[8] N. Kaaniche and M. Laurent, "Data security and privacy preservation in cloud storage environments based on cryptographic mechanisms," *Comput. Commun.*, vol. 111, pp. 120–141, Oct. 2017.
[9] C. Qin, Z. He, X. Luo, and J. Dong, "Reversible data hiding in encrypted image with separable capability and high embedding capacity," *Inf. Sci.*, vol. 465, pp. 285–304, Oct. 2018.

[10] O. R. M. Boudia, S. M. Senouci, and M. Feham, "A novel secure aggregation scheme for wireless sensor networks using stateful public key cryptography," *Ad Hoc Netw.*, vol. 32, pp. 98–113, Sep. 2015.
[11] Q. Li, J. Ma, R. Li, X. Liu, J. Xiong, and D. Chen, "Secure, efficient and revocable multi-authority access control system in cloud storage," *Comput. Secur.*, vol. 59, pp. 45–59, Jun. 2016.
[12] X. Li, S. Tang, L. Xu, H. Wang, and J. Chen, "Two-factor data access control with efficient revocation for multi-authority cloud storage systems," *IEEE Access*, vol. 5, pp. 393–405, 2016.
[13] K. Muhammad, R. Hamza, J. Ahmad, J. Lloret, H. Wang, and S. W. Baik, "Secure surveillance framework for iot systems using probabilistic image encryption," *IEEE Trans. Ind. Informat.*, vol. 14, no. 8, pp. 3679–3689, Aug. 2018.
[14] M. Frustaci, P. Pace, G. Aloi, and G. Fortino, "Evaluating critical security issues of the IoT world: Present and future challenges," *IEEE Internet Things J.*, vol. 5, no. 4, pp. 2483–2495, Aug. 2018.
[15] Y. Y. Chen, J. C. Lu, and J. K. Jan, "A secure EHR system based on hybrid clouds," *J. Med. Syst.*, vol. 36, no. 5, pp. 3375–3384, 2012.
[16] A. Page, O. Kocabas, T. Soyata, M. Aktas, and J.-P. Couderc, "Cloud-based privacy-preserving remote ECG monitoring and surveillance," *Ann. Noninvasive Electrocardiol.*, vol. 20, no. 4, pp. 328–337, Jul. 2015.
[17] F. Armknecht, C. Boyd, C. Carr, K. Gjøsteen, A. Jäschke, C. A. Reuter, and M. Strand, "A guide to fully homomorphic encryption," IACR Cryptol. ePrint Archive, New York, NY, USA, Tech. Rep. 2015/1192, 2015, p. 1192.
[18] J.-J. Yang, J.-Q. Li, and Y. Niu, "A A hybrid solution for privacy preserving medical data sharing in the cloud environment," *Future Gener. Comput. Syst.*, vol. 43, pp. 74–86, Feb. 2015.
[19] W. Feng, Z. Zhang, J. Wang, and L. Han, "A novel authorization delegation scheme for multimedia social networks by using proxy re-encryption," *Multimedia Tools Appl.*, vol. 75, no. 21, pp. 13995–14014, 2016.
[20] K. Riad, Z. Yan, H. Hu, and G.-J. Ahn, "AR-ABAC: A new attribute based access control model supporting attribute-rules for cloud computing," in *Proc. IEEE Conf. Collaboration Internet Comput. (CIC)*, Oct. 2015, pp. 28–35.
[21] K. Riad and Z. Yan, "Multi-factor synthesis decision-making for trust-based access control on cloud," *Int. J. Cooperat. Inf. Syst.*, vol. 26, no. 4, 2017, Art. no. 1750003.
[22] R. Sandhu, D. Ferraiolo, and R. Kuhn, "The NIST model for role-based access control: Towards a unified standard," in *Proc. 5th ACM Workshop Role-Based Access Control*, Jul. 2000, pp. 47–63.
[23] D. F. Ferraiolo, R. Sandhu, S. Gavrila, D. R. Kuhn, and R. Chandramouli, "Proposed NIST standard for role-based access control," *ACM Trans. Inf. Syst. Security*, vol. 4, pp. 224–274, Aug. 2001.
[24] M. Al-Kahtani and R. Sandhu, "A model for attribute-based user-role assignment," in *Proc. 18th Annu. Comput. Secur. Appl. Conf.*, 2002, pp. 353–362.
[25] V. C. Hu, D. Ferraiolo, R. Kuhn, A. Schnitzer, K. Sandlin, R. Miller, and K. Scarfone, "Guide to attribute based access control (ABAC) definition and considerations," U.S. Dept. Commerce, Nat. Inst. Standards Technol., Gaithersburg, MD, USA, Tech. Rep. 800-162, Jan. 2014.
[26] J. Li and A. H. Karp, "Access control for the services oriented architecture," in *Proc. ACM Workshop Secure Web Services (SWS)*, New York, NY, USA, 2007, pp. 9–17.
[27] *OpenStack*. Accessed: Jul. 7, 2019. [Online]. Available: http://www.openstack.org/

KHALED RIAD received the M.S. degree in computer science from Zagazig University, Egypt, in 2011, and the Ph.D. degree in computer science and technology from the School of Computer and Communication Engineering, University of Science and Technology Beijing, China, in 2017. He is currently a Lecturer of computer science with the Mathematics Department, Faculty of Science, Zagazig University. He is also a Post-Doctoral Fellow with the School of Computer Science, Guangzhou University, Guangzhou, China. He has published multiple articles between top international scientific journals and conferences. He is serving as a Reviewer for well-reputed international journals and conferences. His research interests include cloud security, blockchain, Internet of Things, big data, software-defined networking, cryptography, and access control.

RAFIK HAMZA received the M.Sc. and Ph.D. degrees in computer science from the University of Batna 2, in 2014 and 2017, respectively. He was a Principal Engineer with R&D Sonatrach, Boumerdès, in 2018. He is currently a Researcher with Guangzhou University, where he is involved in machine learning and cryptography. He has published several articles between top international scientific journals and conferences. He is serving as a Reviewer for well-reputed international journals and conferences. His research interests include machine learning, information security, access control, image and video processing, chaos theory, and lightweight cryptography applications.

HONGYANG YAN received the M.S. degree from the School of Mathematics and Information Science, Guangzhou University, in 2016, and the Ph.D. degree from the College of Computer Science, Nankai University, in 2019. Her research interests include secure access control, secure cloud storage, and image and video retrieval.