Fundamenta Informaticae 157 (2018) 167–184
DOI 10.3233/FI-2018-1624
IOS Press

RoFa: A Robust and Flexible Fine-Grained Access Control Scheme for Mobile Cloud and IoT based Medical Monitoring

Yuling Chen
Guizhou Provincial Key Laboratory of
Public Big Data, GuiZhou University
Guizhou Guiyang, P. R. China
61997525@qq.com

Min Lei[*]
Information Security Center
Beijing University of Post and Telecommunications
Beijing, P. R. China
leimin@bupt.edu.cn

Wei Ren[†]
School of Computer Science
China University of Geosciences
Wuhan, P. R. China
weirencs@cug.edu.cn

Yi Ren
School of Computing Science
University of East Anglia
Norwich, UK
E.Ren@uea.ac.uk

Zhiguo Qu
Jiangsu Engineering Center of Network Monitoring
School of Computer and Software, Nanjing University
of Information Science and Technology
Nanjing, P. R. China
qzghhh@126.com

Abstract. Cloud computing paradigm is becoming very popular these days. However, it does not include wireless sensors and mobile phones which are needed to enable new emerging applications such as remote home medical monitoring. Therefore, a combined Cloud-Internet of Things (IoT) paradigm provides scalable on-demand data storage and resilient computation power at the cloud side as well as anytime, anywhere health data monitoring at the IoT side. As both the privacy of personal medical data and flexible data access should be provided,attackers exploit diverse

[*] Also works: Guizhou Provincial Key Laboratory of Public Big Data, GuiZhou University, Jiangsu Engineering Center of Network Monitoring, Nanjing University of Information Science & Technology.

[†] Address for correspondence: School of Computer Science, China University of Geosciences, Wuhan, Guizhou Provincial Key Laboratory of Public Big Data, GuiZhou University, Guizhou Guiyang, P. R. China.

The preliminary result is presented in A Robust and Flexible Access Control Scheme for Cloud-IoT Paradigm with Application to Remote Mobile Medical Monitoring, Proc. of 2015 Third International Conference on Robot, Vision and Signal Processing, Kaohsiung, 2015, pp. 130-133.

social engineering and technology attacks ways, access to personal privacy information stored in the home medical monitoring cloud, with more and more social engineering attacks.Therefore, the data in the Cloud are always encrypted and access control must be operated upon encrypted data together with being fine-grained to support diverse accessibility. Since a plain combination of encryption before access control is not robust and flexible, we propose a scheme referred to as RoFa, with tailored design. The scheme is introduced in a step-by-step manner. The basic scheme (BaS) makes use of cipher-policy attributes based encryption to empower robustness and flexibility. We further propose an advanced scheme (AdS) to improve the computation efficiency by taking the advantages of proxy-reencryption. AdS can greatly decrease the computation overhead on hospital servers due to operation migration. We finally propose an enhanced scheme (EnS) to protect integrity by using aggregate signature. RoFa describes a general framework to solve the secure requirements, and leaves the flexibility of concrete constructions intentionally. We finally compare the robustness and the flexibility of the proposed schemes by performance analysis.

Keywords: Access Control; Cloud Computing; Internet of Things; Fine Grained; Robust and Flexible Security

1. Introduction

Remote mobile medical monitoring is widely regarded as an emerging application of Internet of things (IoT) and is expected to play an important role in future personal health-care services. With such services, patients can be transferred out of hospitals and be further taken care at their own familiar homes, that provide better psychological conditions and may result in faster recovery, looked after by their family members, home nurses and local physiotherapists [1, 2, 3]. To collect data about and around patients [4], medical measurements can be performed anytime, anywhere with the help of body area networks (BANs) [5, 6] through various sensors, such as motions sensors, video sensors, bed sensors and so on.

Consider an example, shown in Fig 1. Each patient wears different wireless sensors collecting vital recordings such as electrocardiography (ECG), respiration rate, saturation of peripheral oxygen (SpO_2), etc. Those sensors are located within a dedicated BAN, and communicate with an access point. From the access point, data are transmitted to database servers in the cloud. Such data are regarded as a personal electronic health records (PEHRs) [7]. Later, such data can be accessed by various users, such as the patient, close family members, health care personnel and doctors, etc.

Storing the data in cloud environment becomes natural and also essential, which have received great attention in today's online business world.[8, 9] However, data stored in the cloud may suffer from malicious use by cloud service providers since data owners have no longer direct control over data. Considering data privacy and security,according to security analysis of information systems taking into account social engineering attacks[10], it is a recommended practice for data owners to encrypt data before uploading onto the cloud.[11] The cloud service providers (CSPs) that keep the data for users may access users sensitive information without authorization.[12] To protect the privacy of sensitive information and combat unauthorized accesses, sensitive data should be encrypted by the data owner before outsourcing[13].

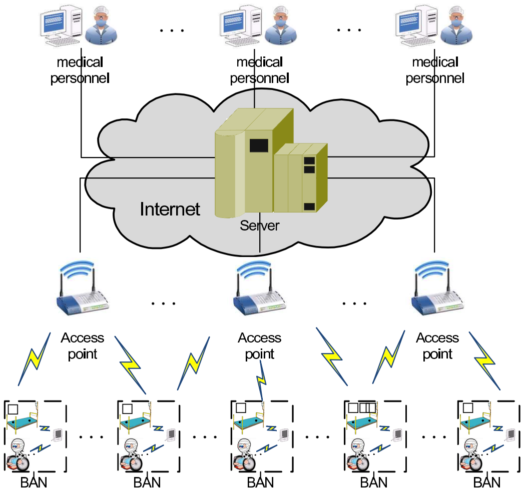

Figure 1. Scenario of remote health monitoring solution, where various users can access patients medical data stored in a Cloud Server with Internet access.

Once sensors are deployed, enormous quantities of data will be generated and collected. If such data are stored in hospital's server, the storage, computation and administration budget may remarkably increase with the elapse of time.Yan Kong *et al.*[14] proposed a decentralized belief propagation-based method, PD-LBP, for multi-agent task allocation in open and dynamic grid and cloud environments where both the sets of agents and tasks constantly change.Besides, the amount of uploaded data may be growing as the number of sensors allocated to each patient will increase. Furthermore, storing patients' data in different hospitals' servers makes it difficult to establish a patient-centric PEHRs. To deal with aforementioned problems, a natural way is relying on cloud computing that can smoothly reallocate storage and computation requirements into a large scale of resources, so as to provide unprecedented processing capabilities within smoothly increasing budget. Therefore, at the client side, the IoT provides a solution for remote and pervasive health monitoring [15]. The data storage in cloud, together with data collection via IoT, appears to be a promising paradigm, known as cloud-IoT paradigm. Such paradigm improves the scalability of pervasive health related data storage, and it is envisaged as a promising architecture.

As data are stored remotely, related operations (e.g., medical data retrieval) will become inevitable. Since the access of such health data becomes more and more ubiquitous, for example, by personal data assistants (PDAs), smart phones, tablet personal computers (PCs), it thus arises a security related problems, such as access control, etc. As health data are stored as personal records, some regulations such as health insurance portability and accountability act of 1996 (HIPAA) are already mandatorily applied to protect a patient's privacy.

Furthermore, the data may be accessed by different users, in order to follow up patient treatment and to assist in case of emergencies They should thus be granted access rights to different parts of data. Since medical records are personal sensitive data,with the rapid development of social engineering, hackers have used various types of social engineering attacks to get patients sensitive of information. For example, using social engineering dictionaries to carry out violent cracking, using search engines to collect sensitive information, and using auxiliary security problems [16]. strong privacy protection needs to be provided in the actual solutions.

To solve the aforementioned problems,Shen Jian *et al.*[17] proposed an efficient public auditing protocol with global and sampling blockless verification as well as batch auditing.Zhangjie Fu *et al.*[18] proposed a similarity search method for encrypted document based on simhash.Wang Baowei *et al.*[19] proposed an adaptive method is presented aiming at spatialCtemporal efficiency in a heterogeneous cloud environment. Zhang Jian *et al.*[20] proposed optimal cluster-based mechanisms by a modified multi-hop layered model for load balancing with multiple mobile sinks for these problems.Chang Shaohua *et al.*[21] proposed an efficient multilayer authentication protocol and a secure session key generation method for WBANs.we propose a fine grained access control scheme on database server in cloud. Data collected by sensors are encrypted using cipher-policy attributed-based encryption (CP-ABE) [22] and stored in the cloud database servers. In CP-ABE each ciphertext is associated with a set of attributes such as sensor type, owner, location, time, etc., whereas each user is assigned with an access structure that is embedded in her secret key. A user is able to decrypt a ciphertext as long as the attribute associated with a ciphertext satisfies the access structure of the secret key. For example, heart-related data is granted accessible by heart surgeons, and data can be accessed by emergency medical service (EMS) staff using the secret key of EMS. Through detailed security and efficiency analysis, we show that the proposed scheme can achieve fine-grained accessibility, scalability, and data confidentiality in remote health monitoring simultaneously.

The rest of the article is organized as follows. Section 2 gives background information on relevant prior work. In Section 3 we discuss the basic assumption and models used throughout the article. Section 4 introduces cryptographic primitives used in the proposed approach. Section 5 provides detailed description of our proposed schemes. We analyze security and performance aspects of the proposed schemes in Section 6. Finally, Section 7 concludes the paper.

2. Background

Traditional access control architecture usually assumes that data owners and data servers are in the same trusted domain, and that the data servers which store data are considered as trusted party. For example, access control list (ACL) is applied to grant the corresponding access rights to authenticated user according to the ACL permissions. However, in cloud computing architecture, data owners and data servers are in different domains so that the data owners cannot guarantee whether their data are still secure. If the data servers in the cloud are untrustworthy or semi-trusted, the ACL cannot protect the data from the attackers at the cloud side. In addition, to provide fine-grained data access control, per-file ACL should be applied. However, it suffers from scalability. That is, the complexity of the per-file ACL based solution increases dramatically as the number of users in the system increases.

More specifically, the data server owners (e.g., administrators), who take full control of the data server, may be curious about users' data.

To guarantee authenticated users access to patient's data stored on the server, an access control policy should be implemented. To provide a fine grained data access control, a naive approach is to maintain an Access Control List (ACL) in the database server in cloud. Upon receiving data access query from a user, the database server verifies the user's identity within the ACL, and grants the corresponding access rights to that user according to ACL. However, this method has two disadvantages:

1) It is not robust. If authenticity of a role is subverted, for example, password of that role is exposed, then all data can be access by that role will be bleak. If empower a mini access right to a role in a minimal set to avoid above role forgery, it will have to induce a large number of mini roles so that the scalability are sacrificed. As the cloud is untrustworthy or semi-trusted, all data in cloud have to be encrypted before access. It makes the robustness even worse - Once a key is exposed, all data encrypted by this key will be divulged because attackers at cloud side can reveal them.

2) It is not flexible. Access from different locations needs to be verified by a central server which stores access control policy, thus the server must be on-line and have a heavy workload. As data are encrypted, an authorized user needs to possess a lot of keys to decrypt retrieved data.

Many authors [23, 24, 25, 26] proposed various solutions to protect PEHRs. However, most of these solutions rely on a full trusted PEHRs server, and assume that the access control policies are implemented by the patients themselves. A straightforward way is to equip role based access control (RBAC) policy [27, 28] to define which persons are obliged to access the patient's data. However, in cloud-IoT paradigm, RBAC suffers drawbacks as follows. Cloud servers themselves may not be fully trusted, so RBAC itself can be easily bypassed by insider attackers who intrude into servers, revise the RBAC policy or perform as an administrator. On the other hand, unexpected users (e.g., EMS staff) would not have access right; consequently they have to be registered in the database occasionally in order to access the data, whereas in EMS even few minutes may result in a difference between death and alive. Furthermore, taking semi-trusted database server into consideration, unharmful but curious adversaries (e.g., administrators) may try to find out information regarding the health records of certain patients (e.g., movie stars or sport stars, etc.).

3. Problem formulation

3.1. Network model

We consider a typical scenario, as shown in Fig. 1, which consists of multiple BANs. Each BAN consists of a home controller (HC) and multiple sensor nodes. Nodes upload data to HC, which is a sink node in a sensor network. It has much more computational and storage capabilities than sensor nodes. The HC also works as a gateway of BAN and the Internet.

The data are stored in cloud server (CS), and the server has extensive computational and storage resources. The data may be accessed by authenticated users (AU) located in different areas. The cloud computing may induce data migration from one location to another.

We assume that an intermediate hospital server (HS) is allocated between CS and HC. Such approach has multiple advantages: if HS does not exist, the data have to be encrypted by HC or sensors themselves. Consequently, the cipher key will be stored at the HC or sensors. It may be a potential leaking point, and the cipher key will be difficult to update. If an HS exists, the HC can communicate with the HS for negotiating or updating cipher key and cipher-policy. Thus, the access control policy will be more flexible. Moreover, HS has more powerful computational ability to accelerate the crypto-related operations. HC thus can be a normal wireless local area network router to save the budget of customers for smooth deployment.

In summary, the IoT-cloud paradigm addressed in this paper is different from some previous papers on traditional wireless sensor networks or distributed systems. We observe that there exists extreme asymmetry in terms of resources between servers and terminals, which results in asymmetrical design in solutions.

3.2. Attacker model and security requirement

The database server in cloud is considered to be *untrustworthy*. Adversaries (e.g., administrators in cloud) usually are assumed to leak private information of patients potentially. Thus the data in cloud need to be encrypted. The cipher key is generated by HS and the encryption is conducted at HS. As the encrypted data have large volume and they will be accessed via cloud from different locations by various requesters, the access control policies should be fine-grained. Thus the access control should be robust to facilitate fast retrieval in a large scale. Moreover, the encrypted data may be accessed via various terminals, so the access method should be flexible.

HSs are also always trustful because they are located in hospital (it is the security assumption for further discussion). The channel between HS and HC is wired, but may be eavesdropped by attackers, so the data confidentiality in the channel should be protected. HC may be compromised by attackers, so the access control policy should be stored at HS.

In our scheme we focus on the security requirements as follows:

Data confidentiality: The data confidentiality in the cloud server and link between HC and HS should be protected. The encrypted data needs to be compatible with further fine-grained access control.

Fine-grained access control: The access control needs to be fine-grained to facilitate the pervasive access anytime, anywhere for authenticated users. Furthermore, as the data are encrypted and are stored at the untrustworthy or semi-trustworthy cloud server, the access control needs to be operated over the encrypted data.

Robustness and flexibility: The access control should be robust so as to be resilient to access policy exposure, and be flexible so as to perform properly via diverse accessing terminals.

The security requirements needs to be guaranteed, meanwhile extremely asymmetrical computation resources in cloud-IoT paradigm should also be tackled. Hence, design goal of this paper is robust and flexible, with regard to fine-grained access control over encrypted data in cloud-IoT paradigm for e-health application.

4. Preliminaries

In this section, we introduce two cryptographic primitives which will be used in our proposed schemes.

4.1. Ciphertext-policy attribute based encryption

In CP-ABE, a user's private key is associated with a number of attributes expressed as strings. When a message is encrypted by a party, they specify an associated access structure over attributes. As a consequence, a ciphertext can be decrypted by a user only if the user's attributes satisfy the access structure of the ciphertext. One of the important features of CP-ABE is that the ciphertext size and public key linear increase as the number of attributes increases, which is independent to the number of users. Furthermore, CP-ABE is resistant to collusion attacks from unauthorized users. Here, we give a short introduction about CP-ABE. Generally, a CP-ABE scheme consists of five algorithms defined as follows:

Setup. The setup algorithm takes no input other than the implicit security parameter. It outputs the public parameters PK and a master key MK.

Encrypt (PK**,** M**,** $\mathbb{A}$)**.** The encryption algorithm takes as input the public parameters PK, a message M, and an access structure $\mathbb{A}$ over the universe of attributes. The algorithm will encrypt M and produce a ciphertext CT such that only a user that possesses a set of attributes that satisfies the access structure will be able to decrypt the message. The ciphertext implicitly contains $\mathbb{A}$.

KeyGeneration (MK**,** S)**.** The key generation algorithm takes as input the master key MK and a set of attributes S that describes the key. It generates a private key SK.

Decrypt (PK**,** CT**,** SK)**.** The decryption algorithm takes as input the public parameters PK, a ciphertext CT, which contains an access policy $\mathbb{A}$, and a private key SK, which is a private key for a set S of attributes. If the set S of attributes satisfies the access structure $\mathbb{A}$ then the algorithm will decrypt the ciphertext and return a message M.

Delegate (SK**,** $\tilde{S}$)**.** The delegate algorithm takes as input a secret key SK for some sets of attributes S and a set $\tilde{S} \subseteq S$. It generates a secret key $\tilde{SK}$ for the set of attributes $\tilde{S}$.

4.2. Proxy re-encryption

The proxy re-encryption scheme (PRE) was proposed in [29]. The basic idea of PRE is to enable a semi-trusted proxy to convert a ciphertext C_A into another ciphertext C_B without revealing any information about the underlying plaintext, where C_A is a ciphertext encrypted under Alice's public key $\mathcal{PK}_A$ and C_B can be decrypted by Bob's private key $\mathcal{PR}_B$. Generally speaking, the proxy can *translate* Alice's ciphertext into a ciphertext that can be decrypted by Bob's private key. That is, given a proxy re-encryption key $r\mathcal{K}_{A\leftrightarrow B}$, one can translate ciphertext C_A under Alice's public key $\mathcal{PK}_A$ into ciphertext C_B under public key $\mathcal{PK}_B$ and vise versa.

One important feature of the PRE is that it provides a way to allocate data providers' computational cost on proxy server in cloud. In the context of this paper, patients (data providers) do not need to suffer large computation cost as a result of changing user's privileges. In the next section, we will utilize the feature of the PRE to design our scheme.

4.3. Aggregate signature

Given n signatures on n distinct messages from n distinct users, it is possible to aggregate all these signatures into a single signature. This single signature (and all n original messages) will convince any verifier that the n users signed the n original messages.

In a general signature aggregation scheme a user, i, signs her message M_i with a signature s_i. Then anyone can use a public aggregation algorithm to combine all n signatures $s_1, \cdots, s_n$ into a single signature s. Moreover, the aggregation can be performed incrementally. That is, signatures

s_1, s_2 can be aggregated into s_{12} which can then be further aggregated with s_3 to obtain s_{123}, and so on [30].

5. Proposed scheme - RoFa

In this section, we propose a scheme called RoFa - RObust and Flexible fine-grained Access control scheme, and it is presented incrementally as the Basic scheme, the Advanced scheme, and the Enhanced scheme for easier understanding of the scheme by a reader. Each consecutive scheme improves the previous one by adding new features.

The notations used in the rest of paper are listed in Table 1 as follows:

Table 1. Notations in the rest of papers

EACP	Encryption and Access Control Paradigm
BaS	Basic Scheme
AdS	Advanced Scheme
EnS	Enhanced Scheme
HS	Hospital Server
SN	Sensor Node
HC	Home Controller
CS	Cloud Server
AU	Authenticated User
RCS	Re-encryption Cloud Server
PK	CP-ABE Public Key
MK	CP-ABE Master Key
Setup	CP-ABE Setup algorithm
Enc	a symmetric encryption algorithm
$DATA$	Sensing Data
PK_{HS}	Hospital Server's Public Key
K	Session key
CPABEnc	CP-ABE encryption algorithm
$\mathbb{A}$	CP-ABE access structure
KeyGeneration	CP-ABE key generation algorithm
S	Authenticated User's attributes
sk	Authenticated User's private key
CPABEDec	CP-ABE decryption algorithm
$K_{HS \rightarrow AU}$	Proxy-Reencryption proxy key
PREnc	Proxy-Reencryption encryption algorithm
PREDec	Proxy-Reencryption decryption algorithm
$Sig()$	digital signature
$AggSign()$	aggregate signature signing algorithm
$ASig$	aggregate signature
$AggVer$	aggregate signature verification algorithm

5.1. Basic scheme (BaS)

In this subsection we present a basic scheme to illustrate our major motivation. To facilitate the compromise resilience, we propose a robust and flexible fine-grained access control scheme. BaS applies CP-ABE to improve robustness and flexibility, comparing with traditional approaches. For example, when the data are encrypted, such kind of access control links the access permission with the ciphertext's properties. E.g., ciphertext with property "(A and B) or C" can be accessed when conditions A and B are both satisfied or condition C is satisfied. Obviously, such scheme enables the conditional expression on access control policy, and can extend the control granularity to a data record or even to a data entry.

The steps of BaS are described as follows:

1) System key generation. HS runs algorithm $Setup$ to generate a public parameter PK and a master key MK for a patient. That is,

$$HS : PK, MK \leftarrow Setup().$$

2) Data collection. Sensor nodes upload sensing data $DATA$ to HC. That is,

$$SN \rightarrow HC : \{DATA\}.$$

3) Data uploading to HS. HC encrypts the sensing data via ordinary encryption scheme, which is out of the scope of this paper. For example, suppose to use hybrid encryption scheme: encrypt a random session key with HS's public key and encrypt data with that session key. That is,

$$HC \rightarrow HS : \{Enc_{PK_{HS}}(K), Enc_K(DATA)\}.$$

4) Data encryption and uploading to CS. HS decrypts the receipt to obtain uploaded $DATA$, encrypts such data again by using the CP-ABE encryption algorithm $CPABEnc$ that takes as input PK, $DATA$, and access control policy $\mathbb{A}$, then uploads the encryption result (CP-ABE encrypted data) to CS. That is,

$$HS : DATA' = CPABEnc(PK, DATA, \mathbb{A}),$$

$$HS \rightarrow CS : DATA'.$$

5) Data access credential granting. An Authenticated User (AU) wants to access a data and then sends request to HS. HS checks the AU's attributes, denoted as S, picks up corresponding patient's MK, invokes $KeyGeneration$ algorithm taking as input S and MK to generate AU's private key SK (as a credential), and sends SK to that AU securely. That is,

$$SK \leftarrow KeyGeneration(S, MK),$$

$$HS \rightarrow_s AU : \{SK\},$$

where $\rightarrow_s$ denotes a secure channel.

6) Data decryption. The AU requests encrypted data from CS and decrypts encrypted data using the CP-ABE decryption algorithm $CP-ABEDec$ that takes as an input her possessing SK and PK. That is,

$$CS \to AU : \{DATA'\},$$
$$AU : DATA = CPABEDec(PK, DATA', SK).$$

Security Analysis. Data are sent from HC to HS to protect data confidentiality. At HS data are encrypted and then uploaded to CS, data confidentiality is maintained. AU must request HS to grant credential to access data in CS. The access control is fine-grained because the access control granularity scales to data length of the encryption. The access credentials are related to user's attributes, and the ciphertext is related to access control policy. The protection scheme is robust, since key exposure only reveals a limited data set and can be averted at HS by re-encrypting data protected by exposed keys.

Performance Analysis. The computational expensive operations, such as system key generation, original data decryption, data encryption before uploading to CS, and data access credential granting, are all conducted by HS. HS is not computationally constraint (only assuming storage constraint exists). The data decryption occurs at AU usually via a personal devices that has enough computation power to complete the $CP - ABEDec$ algorithm.

5.2. Advanced scheme (AdS)

The BaS illustrates our first step solution, i.e., to achieve robust, privacy-preserving, and flexible fine-grained access control. To further decrease the computational load at the HS and improve the scalability of the proposed scheme, we propose the Advanced Scheme - AdS, as follows.

The idea behind the AdS is inspired from following observation. There are four steps at the HS: system key generation, original data decryption, data encryption before uploading to CS, and data access credential granting. The computational overhead on the HS is slightly high due to decryption and encryption. As the first and last operations cannot migrate to the cloud server side due to security concerns, we concentrate on two steps: original data decryption and data encryption before uploading to CS.

We again utilize CS to conduct these two steps. If these two operations can be avoided at the HS and be migrated to another CS, it can save the budget of the HS and improve the scalability of the proposed scheme. If such operations are migrated to cloud side, data privacy should be protected. Thus the cloud server has to change the ciphertext with key PK_{HS} to a ciphertext with data requester's public key; otherwise, $Enc_{PK_{HS}}(K)$ cannot be decrypted to obtain K, and $\{Enc_K(DATA)\}$ cannot be decrypted with K neither. We thus propose a proxy re-encryption based scheme to further decrease the overhead of HS by avoiding one decryption and again migrating CP-ABE encryption to CS. The scheme improves the scalability and quality of service in terms of response delay of HS. Especially, such migration does not sacrifice security. That is, the intermediate servers in the proxy re-encryption schemes are always assumed to be untrustworthy, and security is thus inherently guaranteed even though intermediate servers may be compromised.

We call the migrated cloud server as Re-encryption Cloud Server (RCS). The steps of AdS are described as follows:

1) System key generation. HS runs algorithm $Setup$ to generate public parameter PK and a master key MK for a patient. That is,

$$HS : PK, MK \leftarrow Setup().$$

2) Data generation. Sensor nodes upload sensing data $DATA$ to HC. That is,

$$SN \rightarrow HC : \{DATA\}.$$

3) Data uploading to HS. HC encrypts the sensing data via a symmetric encryption scheme with key K, namely $Enc_K()$. That is,

$$HC \rightarrow HS : \{Enc_{PK_{HS}}(K), Enc_K(DATA)\}.$$

4) Key Re-encrypting and uploading to CS. HS forwards encrypted data

$$\{Enc_{PK_{HS}}(K), Enc_K(DATA)\},$$

together with proxy-reencryption key $K_{HS \rightarrow AU}$, PK, and $\mathbb{A}$ to RCS. HS can do it because $K_{HS \rightarrow AU}$ and corresponding $\mathbb{A}$ can be retrieved from database indexed by AU.

RCS re-encrypts the data by using Proxy Re-encryption algorithm $PREnc$ that takes as input $K_{HS \rightarrow AU}$ and $Enc_{PK_{HS}}(K)$. RCS also encrypts $Enc_K(DATA)$ by using the CP-ABE encryption algorithm $CPABEnc$ with PK and access control policy $\mathbb{A}$. The result, namely encrypted data $DATA'$, together with received $\{Enc_K(DATA)$ is sent to CS. That is,

$$HS \rightarrow RCS : \{K_{HS \rightarrow AU}, PK, \mathbb{A}, Enc_{PK_{HS}}(K), Enc_K(DATA)\},$$
$$RCS : K' = PREnc(Enc_{PK_{HS}}(K), K_{HS \rightarrow AU}),$$
$$RCS : DATA' = CPABEnc(PK, Enc_K(DATA), \mathbb{A}),$$
$$RCS \rightarrow CS : \{K', DATA'\},$$

where K' and $DATA'$ are encrypted K and $DATA$, respectively.

5) Data access credential granting. When an AU wants to access a data, it sends request to HS. HS checks the AU's attributes, denoted as S, picks up corresponding patient's MK, invokes $KeyGeneration$ algorithm taking as input S and MK to generate AU's private key SK (as a credential). Sends SK to the AU securely. That is,

$$SK \leftarrow KeyGeneration(S, MK),$$
$$HS \rightarrow_s AU : \{SK\},$$

where $\rightarrow_s$ denotes a secure channel.

6) Data decryption. The AU requests encrypted data from CS and decrypts encrypted data using algorithm $CPABEDec$ that takes as input her possessing SK and PK. $PREDec$ could be a normal public key decryption. That is,

$$CS \rightarrow AU : \{K', DATA'\},$$
$$AU : Enc_K(DATA) = CPABEDec(PK, DATA', SK),$$
$$AU : K = PREDec_{SK_{AU}}(K'),$$
$$AU : DATA = Dec_K(Enc_K(DATA)).$$

5.3. Enhanced scheme (EnS)

In BaS and AdS, data integrity is not considered for the sake of simplicity of the description. However, the integrity of $K_{HS \to RCS}$, PK, and $\mathbb{A}$ has to be guaranteed. Consequently we enhance the AdS scheme to EnS that protects data integrity yet maintains low communication overhead. As the previous schemes take the advantages of ID-based cryptography, we continue to rely on an ID-based signature scheme. In particular, the signature length affects communication efficiency; and the shorter length is appealing. Therefore, we thus make use of aggregate signature that maintains manageable length even though the number of signers grows. A scenario of EnS is shown in Fig. 2.

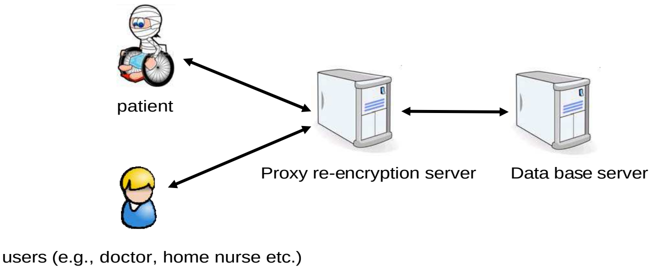
patient
Proxy re-encryption server
Data base server
users (e.g., doctor, home nurse etc.)

Figure 2. Scenario of Enhanced Scheme: combination of proxy re-encryption server and database server.

Next, we describe the steps of EnS in details as follows:

1) System Key generation. HS run algorithm $Setup$ to generate public parameter PK and a master key MK for a patient:

$$HS : PK, MK \leftarrow Setup().$$

2) Data generation. SN uploads sensing data to HC:

$$SN \to HC : \{DATA\}.$$

3) Data uploading to HS. HC encrypts the sensing data via a symmetric encryption scheme with key K, namely $Enc_K()$. That is,

$$HC \to HS : \{Enc_{PK_{HS}}(K), Enc_K(DATA), Sig(Enc_{PK_{HS}}(K), Enc_K(DATA))\}.$$

4) Key Re-encrypting and uploading to CS. HS forwards encrypted data

$$\{Enc_{PK_{HS}}(K), Enc_K(DATA)\},$$

together with proxy-reencryption key $K_{HS \to AU}$, PK, and $\mathbb{A}$ to RCS. HS can do it because $K_{HS \to AU}$ and corresponding $\mathbb{A}$ can be retrieved from database indexed by AU.

RCS re-encrypts the data by using the $PREnc$ algorithm that takes as input $K_{HS \to AU}$ and $Enc_{PK_{HS}}(K)$. RCS also encrypts $Enc_K(DATA)$ by using CP-ABE with PK and access control policy $\mathbb{A}$. The result, namely encrypted data $DATA'$, together with received $Enc_K(DATA)$ is sent to CS. That is,

$$HS : ASig \leftarrow AggSign(Sig(Enc_{PK_{HS}}(K), Enc_K(DATA)), Sig(K_{HS\rightarrow AU}, PK, \mathbb{A})),$$

where $ASig$ is an aggregate signature conducted by HS and aggregated with HC's signature; $AggSign()$ is an aggregate signature algorithm [30, 31].

$$HS \rightarrow RCS : \{K_{HS\rightarrow AU}, PK, \mathbb{A}, Enc_{PK_{HS}}(K), Enc_K(DATA), ASig\},$$

$$RCS : K' = PREnc(Enc_{PK_{HS}}(K), K_{HS\rightarrow AU}),$$

$$RCS : DATA' = CPABEnc(PK, Enc_K(DATA), \mathbb{A}),$$

$$RCS \rightarrow CS : \{K', DATA', ASig\},$$

where K' and $DATA'$ are encrypted K and $DATA$, respectively.

5) Data access credential granting. An AU wants to access a data and then sends request to HS. HS checks the AU's attributes, denoted as S, picks up corresponding patient's MK, invokes $KeyGeneration$ algorithm taking as input S and MK to generate AU's private key SK (as a credential). Sends SK to that AU securely. That is,

$$SK \leftarrow KeyGeneration(S, MK),$$

$$HS \rightarrow_s AU : \{SK\},$$

where $\rightarrow_s$ denotes a secure channel.

6) Data decryption. The AU requests encrypted data from CS and decrypts encrypted data using algorithm $CP - ABEDec$ that takes as input keys SK and PK. $PREDec$ could be a public key decryption. That is,

$$CS \rightarrow AU : \{K', DATA', ASig\},$$

$$AU : Enc_K(DATA) = CPABEDec(PK, DATA', SK),$$

$$AU : K = PREDec_{SK_{AU}}(K'),$$

$$AU : DATA = Dec_K(Enc_K(DATA)),$$

$$AU : AggVer(Enc_{PK_{HS}}(K), Enc_K(DATA), ASig),$$

where $AggVer$ is verification algorithm in aggregate signature scheme.

Note that, our final proposed scheme RoFa is indeed scheme EnS, which incorporates all features in BaS and AdS. Hereby the proposed scheme is presented in an incremental manner for better understanding.

6. Scheme analysis

6.1. Security analysis

In this section, we analyze the security aspects of EnS (or RoFa). As CP-ABE and PRE schemes are proven to be secure in [22] and [29], the data confidentiality of the data stored on the cloud server and link between HC and HS is protected. In addition, since data are encrypted by CP-ABE, they are specified an associated access structure over attributes. The ciphertext can be decrypted if and only if the users' attributes satisfy the access structure of the ciphertext. This implies that confidentiality and fine-grained access control are provided through data encryption scheme.

Traditionally, confidentiality and access control are placed into two distinct domains. Confidentiality relies on encryption; while access control relies on authentication and authorization.

Firstly, we describe a new paradigm, called encryption and access control paradigm (EACP). In EACP, each data item is encrypted, and then each authenticated user is assigned the accessibility of a group of encrypted data items.

Secondly, we consider fine-grained access control. Suppose there are s tables in the database, denoted as $T_1, T_2, \cdots, T_s$. A table $T_i, i \in [1, s]$, consists of items $I[i, j, k], i \in [1, s], j \in [1, m], k \in [1, n], t = mn$. Suppose y users will access the data items. Fine-grained access control is defined as follows: Each authenticated user is granted a read right to data item $I[i, j, k]$. Therefore, the relationship between a user and her accessible items is a one-to-many mapping.

For the whole system, assume that there are x encryption keys, denoted as $K_1, \cdots, K_x$. As in EACP data are encrypted at first, $I[i, j, k]$ is encrypted by a single key. Suppose $I[i, j, k]$ is encrypted by $K_p, p \in [1, x]$, we denoted it as $I[i, j, k] = K_p$. Assume that there are y users in the system, denoted as $U_1, \cdots, U_y$. Each user possesses one credential for authentication and several keys for decryption. Suppose that one user's authorization is specified by one access control rule. Thus the rules have the form as follows: $U_q : AccessSet_{U_q} = \{I[i, j, k] | I[i, j, k] = K_p\}, i \in [1, s], j \in [1, m], k \in [1, n], p \in [1, x], q \in [1, y]$.

BaS is more robust than EACP. If a user's credential and one of her possessing keys ($K_{leak} \in \{K_1, \cdots, K_x\}$) are exposed, attackers can successfully impersonate that user and access corresponding items. If such items are encrypted by leaked key, privacy will violated. The exposed data items are denoted as $\{I[i, j, k] | I[i, j, k] = K_{leak}\}$. $| * |$ returns the number of members in a set. The percentage of privacy loss of that user is

$$\frac{|\{I[i, j, k] | I[i, j, k] = K_{leak}\}|}{|AccessSet|}.$$

On average smn/x items are encrypted with the same key. If one key is exposed, smn/x items will be in risk. That is, if more credentials are exposed, more data privacy will be lost. In contrast, in BaS if one user's key is exposed, it only influences the privacy of that user's accessible items.

AdS is more robust than BaS is. AdS migrates public key decryption and CP-ABE encryption operations to cloud side. Such operations take place for each sensed data for every monitor patients, so that the computational overhead is consecutive and exponential. Thus, BaS is weaker in terms of availability than AdS.

EnS is more robust than AdS is. EnS protects the integrity of sensing data and authenticate data source of HC and HS. As the channels between HC and HS, between HS and RCS are insecure channels. The whole cloud side is also assumed to be insecure channel. The data integrity and data source authentication can be verified by AU upon the receiption of sensing data. Whereas, AdS does not protect data integrity and data source authentication.

6.2. Performance analysis

The performance evaluation of PRE, CP-ABE, and AggSign is already justified by original cited papers, and our scheme inherently relies on the performance of PRE, CP-ABE, and AggSign. Especially, the AdS scheme conducts PRE and CP-ABE at the cloud side, which usually has no computational ability concerns.

BaS is more flexible than EACP is. Each user has to be assigned p keys securely in EACP; but only one key needs to be securely assigned in BaS. HS has to maintain a large volume of secret keys for each user and index to encrypted data in database in EACP; but only access structure is maintained for each user in BaS.

AdS is more flexible than BaS is. BaS needs to conduct one public key decryption and one CP-ABE encryption at HS; but AdS migrates such operations to the cloud therefore such computational overhead is avoided.

Furthermore, EnS is more flexible than AdS is. In EnS, proxy-reencryption operations can be migrated to further multiple RCSs in cloud due to the aggregate signature; but in AdS, proxy-reencryption operations can only be migrated to the first RCS in the cloud, as further migration may not be trusted by other RCSs in the cloud due to the possibility of data fake in RCS.

The comparisons between different schemes are listed in Table 2.

Table 2. Comparison between different schemes

Scheme	Fine-Grained	Robustness	Flexibility
EACP	N	N	N
BaS	Y	Improved	Improved
AdS	Y	Improved	Improved
EnS	Y	Improved	Improved

7. Conclusions

In this paper, we propose CP-ABE based fine-grained access control scheme BaS as a basic setting. We further improve BaS's robustness and flexibility by using Proxy-Reencryption and social engineering, which is called an AdS scheme. In AdS, decryption and CP-ABE encryption are migrated to the cloud side, which still makes use of the cloud-IoT paradigm. Finally, an enhanced scheme EnS that protects integrity of migrating operations is proposed, which enables the appealing properties and performs as a final proposal - RoFa. Our analysis formally verified its feathers in robustness and reflexibility.

Acknowledgments

This work is supported by the National Key R&D Program of China (2017YFB0802703),the Open Foundation of Guizhou Provincial Key Laboratory of Public Big Data (2017BDKFJJ017, 2017BDKFJJ006), the CCF-Venustech Hongyan Research Initiative (2016-009,2017-004), the PAPD fund, the CICAEET fund, and the Open Funding of Hubei Provincial Key Laboratory of Intelligent Geo-Information Processing (KLIGIP2016A05).

References

[1] Koch S. Home telehealth–current state and future trends. *Elsevier Int. Journal of Medical Informatics*, 2006;75(8):565–576. doi: 10.1016/j.ijmedinf.2005.09.002.

[2] Sheppered S, and Iliffe S. Hospital at home versus in-patient hospital care. *Cochrane Database of Systematic Reviews*, 2005, pp. 1–172. doi:10.1002/14651858.CD000356.pub2.

[3] Hebert MA, Korabek B, and Scott RE. Moving research into practice: a decision framework for integrating home telehealth into chronic illness care. *Elsevier Int. Journal of Medical Informatics*, 2006;75(12):786–794. Jochen Moehr Special Issue. doi:10.1016/j.ijmedinf.2006.05.041.

[4] Skubic M, Alexander G, Popescu M, Rantz M, and Keller J. A smart home application to eldercare: Current status and lessons learned. *Technology and Health Care*, 2009;17(3):183–201. doi:10.3233/THC-2009-0551.

[5] Jovanov E, Milenkovic A, Otto C, and De Groen P. A wireless body area network of intelligent motion sensors for computer assisted physical rehabilitation. *Journal of NeuroEngineering and Rehabilitation*, 2005;2(1):2–6. URL `https://doi.org/10.1186/1743-0003-2-6`.

[6] Rubel P, Fayn J, Nollo G, Assanelli D, Li B, Restier L, Adami S, Arod S, Atoui H, Ohlsson M, et al.. Toward personal eHealth in cardiology. Results from the EPI-MEDICS telemedicine project. *Elsevier Journal of Electrocardiology*, 2005;38(4):100–106. doi:10.1016/j.jelectrocard.2005.06.011.

[7] Fensli R, Oleshchuk V, Donoghue JO, and Reilly PO. Design requirements for a patient administered personal electronic health record. *Biomedical Engineering, Trends, Researches and Technologies*, 2010. doi:10.5772/12948.

[8] Ren Y, Shen J, Wang J, Han J, and Lee S. Mutual Verifiable Provable Data Auditing in Public Cloud Storage. in *Journal of Internet Technology*, 2015;16(2):317–323. doi:10.6138/JIT.2015.16.2.20140918.

[9] Ma T, Zhou J, Tang M, Tian Y, Al-Dhelaan A, and Al-Rodhaan M. Social network and tag sources based augmenting collaborative recommender system. in *Ieice Transactions on Information & Systems*, 2015;98(4):902–910. URL `http://doi.org/10.1587/transinf.2014EDP7283`.

[10] Kotenko I, Stepashkin M, and Doynikova E. Security analysis of information systems taking into account social engineering attacks, in *Parallel, Distributed and Network-Based Processing (PDP), 2011. 19th Euromicro International Conference on*, 2011 pp. 611–618. doi:10.1109/PDP.2011.62.

[11] Fu Z, Ren K, Shu J, and Sun X. Enabling personalized search over encrypted outsourced data with efficiency improvement. in *IIEEE Transactions on Parallel & Distributed Systems*, 2016;27(9):2546–2559. doi:10.1109/TPDS.2015.2506573.

[12] Fu Z, Sun X, Liu Q, Zhou L, and Shu J. Achieving Efficient Cloud Search Services: Multi-keyword Ranked Search over Encrypted Cloud Data Supporting Parallel Computing. in *IEICE Transactions on Communications*, 2015;98(1):190–200. doi:10.1587/transcom.E98.B.190.

[13] Xia Z, Wang X, Sun X, Wang, Q. A secure and dynamic multi-keyword ranked search scheme over encrypted cloud data. in *IEEE Transactions on Parallel & Distributed Systems*, 2015;27(2):340–352. doi:10.1109/TPDS.2015.2401003.

[14] Kong Y, and Zhang M, and Ye D. A belief propagation-based method for task allocation in open and dynamic cloud environments, in *Knowledge-Based Systems*, vol. 115, 2017 pp. 123–132.

[15] Ren Y, Oleshchuk VA, Li FY, and Sulistyo S. FoSBaS: an efficient key management scheme for body area networks. in *Proc. IEEE Wireless Communications and Networking Conference (WCNC '12)*, Paris, France, Apr. 2012.

[16] Mouton F, and Leenen L, and Venter HS. Social engineering attack detection model: Seadmv2, in *Cyberworlds (CW), 2015 International Conference on*, 2015 pp. 216–223. doi: 10.1109/CW.2015.52.

[17] Shen J, Shen J, Chen X, Huang X, and Susilo W. An Efficient Public Auditing Protocol with Novel Dynamic Structure for Cloud Data, in *IEEE Transactions on Information Forensics and Security*, 2003.

[18] Fu Z, Shu J, Wang J, Liu Y, and Lee S. Privacy-Preserving Smart Similarity Search Based on Simhash over Encrypted Data in Cloud Computing, in *Journal of Internet Technology*, 2015;16(3):453–460. doi:10.6138/JIT.2015.16.3.20140918.

[19] Wang, B, Gu X, Ma Li, and Yan S. Temperature error correction based on BP neural network in meteorological wireless sensor network, in *International Journal of Sensor Networks*, 2017;23(4):265–278. URL `https://doi.org/10.1504/IJSNET.2017.083532`.

[20] Zhang J, Tang J, Wang T, and Chen F. Energy-efficient data-gathering rendezvous algorithms with mobile sinks for wireless sensor networks, in *International Journal of Sensor Networks*, 2017;23(4):248–257. doi:10.1504/IJSNET.2017.083533.

[21] Shen J, Chang S, Shen J, Liu Q, and Sun X. A lightweight multi-layer authentication protocol for wireless body area networks, in *Future Generation Computer Systems*, 2016 doi: 10.1016/j.future.2016.11.033.

[22] Bethencourt J, Sahai A, and Waters B. Ciphertext-policy attribute-based encryption. in *Proc. IEEE Security and Privacy (S&P '07)*, 2007, pp. 321–334. doi:10.1109/SP.2007.11.

[23] Dekker M, and Etalle S. Audit-based access control for electronic health records. in *Proc. Second Int. Workshop on Views on Designing Complex Architectures (VODCA 2006)*, vol. 168, 2007, pp. 221–236. URL `https://doi.org/10.1016/j.entcs.2006.08.028`.

[24] Wilikens M, Feriti S, Sanna A, and Masera M. A context-related authorization and access control method based on rbac. in *Proc. 7th ACM Sym. on Access control models and technologies*, Monterey, CA, USA, Jun. 2002, pp. 117–124. [Online]. URL `http://doi.acm.org/10.1145/507711.507730`

[25] Blobel B. Authorisation and access control for electronic health record systems. *Elsevier Int. J. Medical Informatics*, 2004;73(3):251–257. doi:10.1016/j.ijmedinf.2003.11.018.

[26] Becker M, and Sewell P. Cassandra: distributed access control policies with tunable expressiveness. in *Proc. 5th IEEE Int. Workshop on Policies for Distributed Systems and Networks (POLICY '04)*, Yorktown Heights, New York, USA, Jul. 2004, pp. 159–168. doi:10.1109/POLICY.2004.1309162.

[27] Sandhu R, and Samarati P. Access control: principle and practice. emphIEEE Commun. Mag., 1994;32(9):40–48. doi:10.1109/35.312842.

[28] Sandhu R, Coyne E, Feinstein H, and Youman C. Role-based access control models. *IEEE Computer*, 1996;29(2):38–47.

[29] Blaze M, Bleumer G, and Strauss M. Divertible protocols and atomic proxy cryptography. in *Proc. Advances in Cryptology - EUROCRYPT '98*. Lecture Notes in Computer Science, vol 1403. Springer, Berlin, Heidelberg. 1998, pp. 127–144. URL `https://doi.org/10.1007/BFb0054122`.

[30] Boneh D, Gentry C, Lynn B, and Shacham H. A survey of two signature aggregation techniques *CryptoBytes*, 2003;6(2):1–10.

[31] Ma D, and Tsudik G. Extended abstract: forward-secure sequential aggregate authentication. in *Proc. IEEE Symp. on Security and Privacy (S&P '07)*, Oakland, CA, USA, May. 2007, pp. 86–91. doi: 10.1109/SP.2007.18.