J Med Syst (2012) 36:4005–4020
DOI 10.1007/s10916-012-9873-8

ORIGINAL PAPER

Secure Dynamic Access Control Scheme of PHR in Cloud Computing

Tzer-Shyong Chen · Chia-Hui Liu · Tzer-Long Chen · Chin-Sheng Chen · Jian-Guo Bau · Tzu-Ching Lin

Received: 14 July 2012 / Accepted: 25 July 2012 / Published online: 28 August 2012

Abstract With the development of information technology and medical technology, medical information has been developed from traditional paper records into electronic medical records, which have now been widely applied. The new-style medical information exchange system "personal health records (PHR)" is gradually developed. PHR is a kind of health records maintained and recorded by individuals. An ideal personal health record could integrate personal medical information from different sources and provide complete and correct personal health and medical summary through the Internet or portable media under the requirements of security and privacy. A lot of personal health records are being utilized. The patient-centered PHR information exchange system allows the public autonomously maintain and manage personal health records. Such management is convenient for storing, accessing, and sharing personal medical records. With the emergence of Cloud computing, PHR service has been transferred to storing data into Cloud servers that the resources could be flexibly utilized and the operation cost can be reduced. Nevertheless, patients would face privacy problem when storing PHR data into Cloud. Besides, it requires a secure protection scheme to encrypt the medical records of each patient for storing PHR into Cloud server. In the encryption process, it would be a challenge to achieve accurately accessing to medical records and corresponding to flexibility and efficiency. A new PHR access control scheme under Cloud computing environments is proposed in this study. With Lagrange interpolation polynomial to establish a secure and effective PHR information access scheme, it allows to accurately access to PHR with security and is suitable for enormous multi-users. Moreover, this scheme also dynamically supports multi-users in Cloud computing environments with personal privacy and offers legal authorities to access to PHR. From security and effectiveness analyses, the proposed PHR access scheme in Cloud computing environments is proven flexible and secure and could effectively correspond to real-time appending and deleting user access authorization and appending and revising PHR records.

Keywords Personal health records · Cloud computing · Access control · Key management · Lagrange interpolation

Introduction

Foreword

Continuing on past developments on Electronic Medical Record Systems, this project is carried out with the purpose

T.-S. Chen (✉) · T.-C. Lin
Department of Information Management, Tunghai University,
Taichung, Taiwan
e-mail: arden@thu.edu.tw

T.-C. Lin
e-mail: zijing@thu.edu.tw

C.-H. Liu
Department of Digital Literature and Arts, St. John's University,
Taipei, Taiwan
e-mail: nancy@mail.sju.edu.tw

T.-L. Chen
Department of Information Management, Taiwan University,
Taipei, Taiwan
e-mail: d97725005@ntu.edu.tw

C.-S. Chen
Department of Statistics, Tunghai University,
Taichung, Taiwan
e-mail: sschen@thu.edu.tw

J.-G. Bau
Department of Biomedical Engineering, Hungkuang University,
Taichung, Taiwan
e-mail: baujg@sunrise.hk.edu.tw

of assisting medical professionals in dispensing medical care by prioritizing patients' health maintenance or management. In addition to patients' rising awareness, with advanced development and popularization of information technologies and the Internet, many studies have been undertaken to overhaul traditional clinical diagnosis by integrating information technology into medical care in order to promote better treatment tracking [1]. Affirmative reports [2–4], and positive feedback from organizations [5] and health care centers and services [6] that expressed support for e-Health tools in assisting patient access management have prompted active development in the restoration of health and medicare care services. With such similar motives, M. Li [7] proposed a patient-centered, Personal Health Record (PHR) exchange architecture. PHR is so-called because it is patients who maintain and manage these health records, that include medical records of professional diagnoses, voluntary health care programs, and other applications and services related to self-health management. As defined by the Markle Foundation report in Connecting for Health [2],"The PHR is an Internet-based set of tools that allows people to access and coordinate their lifelong health information and make appropriate parts of it available to those who need it." The PHR is thus a lifelong health management tool with the primary objective of assisting people in understanding better their own health information.

The history of PHR in its implementation and application is rather short. Up till now, many studies largely focused on treatment and health care management record protocols, under which the development of PHR began to take shape and is now closing on its practical application. PHRs are often linked with electronic medical records (EMRs) and electronic health records (EHRs), which are increasingly being used. The increasing use of PHRs has also been driven by the growing digitization of health/medical information. Especially in the healthcare market, where various different medical information systems are becoming better interconnected, the application of PHRs has grown with concomitant increases in health improvement and disease prevention.

Current developed electronic health record exchange standards such as Health Level Seven (HL7), together with Electronic Medical Record (EMR), Healthcare Information System (HIS), and other related healthcare applications, have allowed medical professionals to add, modify, and exchange medical records through computers or mobile devices. The scope of these applications is largely focused on electronic medical record management and data transmission. These are all operated and managed from the part of medical information providers that oversee electronic health records exchange between hospitals. It is to this that M. Li [7] proposed the Personal Health Records (PHR) that is managed by patients, and allows them to collect and monitor over their own medical records such as, health records from different medical institutions, past surgeries, medical treatments, allergic reaction histories, etc. This collected information can then be provided voluntarily by patients to their doctors for diagnosis, which can then be stored for example, as medical insurance reference records.

The PHR developed from patient-owned EMR [8, 9] to construct a collection of individual patient information. Basic information of a PHR include records such as patients' medical history, health insurance information, allergic reactions, vaccinations, medical treatment, surgeries, patients' wishes in case of unconsciousness, unavailability, or absence, among others. These record histories have been influential during the decision-making of clinical diagnoses, lowering medical professionals' risk of misdiagnoses, and also minimizing treatment delay, or ineffectual treatment. In an EMR, diseases are classified according to International Classification of Diseases (ICD), and patients are restricted from access and control. In a PHR however, patients can access their own data without restrictions, as they are themselves responsible for the data input. As such, data reliability is often questioned [10]. Therefore, there is a need for medical professionals to access and verify the inputted data.

In constructing a patient-oriented PHR system, information safety of confidentiality, integrity, and availability (CIA) [11] must be considered:

(1) Confidentiality: The PHR contains several personal information that most medical information systems do not allow patients to maintain, and is instead managed by the information system. If these data is to be protected, it should be attained through information system's safety protocols. To do so, the safety mechanisms of the system should be able to withstand malicious attacks and unauthorized access.
(2) Availability: Medical records play an important part in clinical decisions, as they lower misdiagnosis risks and cuts down on diagnosis time. With a complete access mechanism, medical staffs can access patients' related records, drug information etc., improving overall medical care quality, efficiency, and safety.
(3) Integrity: Personal medical information generally consists of data such as medical images, reports, drug records etc., in various media forms and format involving not only different medical departments, but also doctors, nurses, patients, and other interested parties. Thus, data completeness and integrity is vital and must be safeguarded during access and transmission, including confirmation of data source and content integrity, and accurate update of record. User access to PHR must also be verified to prohibit change to medical information by unauthorized parties to ensure data completeness and consistency.

Research motive and objective

In recent years, the PHR has become a patient-centric health information exchange model. By consolidating all information in the database of a service provider, through web browsers or the Internet, patients can connect, create, manage and control their health profiles, making the PHR model efficient in access, storage, and sharing of medical data. More importantly, because patients with their complete access and control of their medical information can effectively share the information with interested users including medical institutions, health insurance providers, and family and friends, this also improves preciseness and quality of personal health care, lowering health care costs.

With the advent of cloud computing, medical information technology firms and healthcare services have moved their PHR to clouds. Two primary cloud platform providers, Google and Microsoft, offer PHR services on their clouds called Google Health1 [12] and Microsoft Health Vault [13] respectively. In 1996, the Health Insurance Portability and Accountability Act (HIPAA) [14–16] outlined legal privacy and security protection for PHR. But it does not sufficiently address all issues involved, especially because HIPAA only applies to covered entities such as health plans, healthcare clearinghouses, and healthcare providers. Emerging cloud-based PHR service providers like Dossia, Microsoft, and Google are not covered entities. Integration of PHR with cloud service provides the following benefits: (1)Reduced cost, (2)Medical resource sharing and exchange, (3)Dynamic scalability of resources, (4)Enhanced flexibility, (5)Elimination of device limitation.

In consideration to environment security of cloud computing, security mechanisms of information systems must effectively safeguard PHR’s confidentiality and its prudent access. To counter the risk of privacy exposure, service providers of PHR should not only encrypt patients’ data, but also allow patients, the custodians of the PHR to control with whom they want to share records with. Thus, in addition to the traditional mindset of having service providers encrypting messages, the PHR imparts users with access control mechanism [17].

Realizing PHR system in clouds will see multiple-user access that needs substantial mass-number access control, resulting in possible computation overload and data management difficulty from system generation. On the one hand, authorized users may access from all sorts of channels, which include known authorized users and new users applying for authorization through different channels. The demands of such users are usually very large and unpredictable. Allowing all users to manage their own accounts directly could thus make secret key management exceedingly complicated with the massive number of users involved.

On the other hand, as users can manage the stored PHR in the cloud anytime, anywhere, without being limited by having to wait for other users’ response for access approval, the PHR’s accessibility and system availability is unrestricted. With continuous addition and modification to PHR content and the stored PHR data coming from different medical institutions, cloud servers face authorized users making requests for newest updated information at all times. Therefore, spontaneous status updates of PHR in cloud service must be realized.

Though much has been done to encrypt information with various cryptosystems in order to prevent illegal external access to data [17–21], these are mainly single-custodian structured. In a cloud environment, the PHR is no longer sole-owned. An efficient and secure access control mechanism must be considered for such multi-user setups with different access rights. A secure and efficient access control mechanism is needed to safeguard the privacy and security of users’ medical information. In addition, patients should have complete rights over access control which when necessary, can be set to add or remove access rights [7, 22].

In this paper, we propose a dynamic access structure that can impart precise control access to cloud server’s medical record under multi-user setting. To ensure every patient retains maximum control over their medical records, we adopted cryptography based on Lagrange multipliers for encrypting the records. By allowing every custodian to generate his/her own related keys, patients can choose with whom to share their records with. Therefore, central to this paper is the objective of enhancing the encryption of PHR, and improving on user dynamic access policies. To reduce the complexity of key distribution, we overhaul past hierarchical models and created partial order relation to manage users. This reduces key management complexity drastically, and at the same time allows users to not only retain access control of PHR, but one that permits issuance of limited access rights to other users, such as doctors, pharmacists, nurses, researchers etc. This is a very flexible method for multi-user dynamic access control in coordinating the needs for immediate addition, or removal of user access, and also for addition and modification of PHR, making it more suitable for PHR cloud application.

Related work

Electronic medical record

Medical records comprise of detailed information of patients’ past diagnosis such as laboratory results, and diagnosis records that are disparate, and do not allow easy sharing and exchange, resulting in inefficiency and medical resource wastage. As a result, such traditional paper medical records are increasingly being given way to electronic medical record for easier information integration and update.

Electronic medical records is a type of medical record that electronically access, transmit, accept, save, retrieve, connect, and process multimedia information of past, present, and future records of patients' physiological and psychological condition. Definitions of electronic medical record vary, from Computer Patient Record (CPR), and Electronic Medical Record (EMR) in the early days, to recent extended explanations of the Electronic Health Record (EHR).

The Computer-Based Patient Record Institute (CPRI) of the United States defines the CPR as related electronic information of an individual's lifelong health status and health care. In 1997, the Institute of Medicine of the U.S. National Research Council further pointed out that the CPR must provide for complete, accurate data that assist in diagnosis decisions and related medical research. With the popularization of electronic medical records, medical services have gradually diversified. The rise of personal health management issues [23] have also encouraged patients to gather more information, along with better decision options, and better health care plans. Personal Health Record (PHR) overlaps with Electronic Medical Record (EMR), but has its differences. The EHR do not allow patient access or patient control of access to information. The PHR is designed for patients' control and is also unique in that it can be accessed through the Internet from anywhere. PHR also emphasizes on confidentiality or privacy protection, availability, and authenticity, but does not demand EMR's documental properties of non-repudiation and integrity.

Personal health record

According to the definition of Markle Foundation [2], the PHR is a set of computer-based tools that allow people to access and coordinate their lifelong health information and make appropriate parts of it available to those who need it. A patient's PHR can be electronically saved, and translated into standard formats while meeting security standards of medical service providers like HIPAA and HL7. It may also include online educational tools and messages to assist patients make the best decisions to improve their own health care quality and cost.

The PHR system integrates patient health information from disparate sources, including measurement records (blood pressure, diet, exercise habits, etc.), doctors' records (medical orders, doctors' orders, etc.), hospital and laboratory records (ECG, medical imaging etc.), legal documents, letter of proxy, and insurance documents, etc. In addition, the PHR also includes medical reference information, medical treatment, drug use, and other non-medical management information. Parts of the PHR are also derived from the EMR database. But it should be noted that unlike the EMR, the PHR does not demand EMR's documental properties of non-repudiation and integrity.

The primary objective of the PHR is to assist people to gain deeper understanding of their own health through its use as a lifelong health management tool. The value of the PHR is its long-term cumulative record of personal health that promotes personal health and can be consummately referred to in the future when faced with disease occurrence [24]. In 2005, the National Committee on Vital and Health Statistics (NCVHS) [25] outlined properties of the PHR and the PHR system as follows: (1)Scope and nature of content, (2)Source of information, (3)Features and functions, (4)Custodian of the record, (5)Data storage, (6)Technical approaches, (7)Party controlling access to the data.

Medical services and cloud computing

Introduction to cloud computing

Based on the study, Vaquero, LM et al. defined cloud computing as follows [26]: Clouds are a large pool of easily usable and accessible virtualized resources (such as hardware, development platforms and/or services). These resources can be dynamically re-configured to adjust to a variable load (scale), allowing also for an optimum resource utilization. This pool of resources is typically exploited by a pay-per-use model in which guarantees are offered by the Infrastructure Provider by means of customized Service-Level Agreements. According to the National Institute of Standards and Technology, Information Technology Laboratory (NIST) [27], cloud computing is a conceptual model that connect shared resources (such as network, server, storage, applications, and services) through networks to users' demands using minimum management to achieve rapid configuration and distribution. The three fundamental service models are:

(1) Software as a Service (Saas): This service model provides software through the Internet with manufacturers installing applications on a cloud server which can be accessed by clients and operated as per their needs.
(2) Platform as a Service (PaaS): In this model, cloud providers supply a computing platform to its clients where they can deploy applications of its own, program languages of its own, all without having to maintain or control the cloud equipment such as network equipment, server, etc.
(3) Infrastructure as a Service (IaaS): Vendors integrate basic infrastructure such as IT systems and database and then rents them to clients.

Cloud computing application in medical services

As there are numerous advantages to cloud computing, considerable number of personal health records is now being used in the United States, such as Global Lifeguard, and Healthframe. The United States government has also put forward plans for health cloud systems that integrate personal health care information, clinical records, hospital medical care, and telehealth services. The Clinical Informatics Research Group at the University of Washington has developed the Patient-centric Health Record (PcHR), as an example of an online patient-centric personal health record, one that the patient owns and controls. Such cloud application trends is encouraging and assisting in PHR development of a patient-centric health information exchange model on cloud. Our greatest concern with PHR is security and stability. Cloud computing services rely completely on the Internet as a medium. Cloud Security Alliance [28, 29] listed cloud-related security guidelines for key areas in cloud computing supported with analysis and suggestions. In the face of such risks, legal protection has been stipulated in information laws, while administrative regulations have also been passed to protect health care systems on data security and privacy of cloud users, such as the US Health Insurance Portability and Accountability Act (HIPAA) [14] and the Canadian Personal Information Protection and Electronic Documents Act (PIPEDA) [30, 31].

Cryptography and encryption systems

Although the transfer of PHR to cloud environment greatly increases security threats, data integrity, confidentiality, and availability cannot be compromised either. Since the primary objective of the PHR system is to grant lawful access to authorized users, we realize this objective through cryptography. Following is a brief introduction to cryptography and encryption systems.

Basic cryptography

Cryptography is a practice and study of techniques (such as mathematical formulas) to randomize messages in order to render them unreadable to other users. By encrypting messages from plaintext into ciphertext, important messages can be protected. Through decryption technology, these ciphertexts can then be translated into plaintext for reading as shown in Fig. 1:

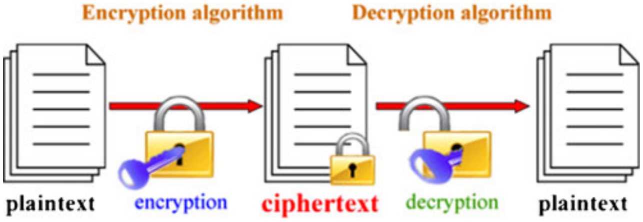


Fig. 1 Encryption and decryption technology

Generally speaking, to oversee system security, a password system must at least have the following four functions: (1)Confidentiality, (2)Authentication, (3)Integrity, (4) Non-repudiation. In accordance with mathematical variances in keys, cryptography systems are divided into two major systems: private key cryptosystem, and public key cryptosystem [32].

Private key cryptosystem

Private key cryptosystem is also known as symmetric cryptosystem or one-key cryptosystem [33–35]. In this system, the plaintext is encrypted and decrypted with one single private key. Prior to sending the message, the sender consults with the receiver over the private key to be used. Following, the sender encrypts the message with the private key into ciphertext and sends it to the receiver. Upon receiving, the receiver uses the same private key to interpret the ciphertext into plaintext for reading. Figure 2 illustrates the process.

Common private key cryptosystems are Data Encryption Standard (DES) [36] and IDEA [37].

Public key cryptosystem

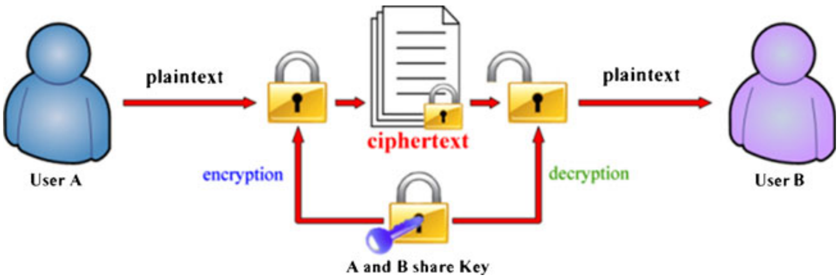
Public key cryptosystem is also known as asymmetric cryptosystem, or two-key cryptosystem [33–35], illustrated in Fig. 3. In this password system, two different keys are used for encryption and decryption, them being the receiver's public key and the corresponding private key respectively. A complex mathematical relation exists between the two keys to ensure no one can derive the private from the public key within a limited time.

The concept of public key cryptography was devised by Diffie and Hellman in 1976 to solve the three said problems. Thus, many current information security systems are designed according to the principles of public key password system. Public key cryptography has the following advantages: (1)Protects information privacy, (2)Simplifies allocation and management of keys, (3)Possess non-repudiation. Although public key password system have the above-mentioned advantages, owing to complex encryption and decryption processes, its efficiency is generally low. Common public key cryptography is the RSA [39], the ElGamal [40], and the Elliptic Curve [38, 41].

Lagrange interpolation polynomial

Following is a brief introduction to Lagrange interpolation polynomial, which we have adopted for encryption and decryption processes. In numerical analysis or other

Fig. 2 Private key cryptosystem



applications, many practical problems are represented through functions to express intrinsic relationships or regularity. However, the precise relationship between variable *x* and variable *y* of many functions are extremely complex, and cannot be determined through experiments. The method of Lagrange interpolation enables us to obtain a polynomial which passes through a finite set of points in the x-y plane. The polynomial obtained by this method is called the Lagrange polynomial. Mathematically, the Lagrange interpolation polynomial can obtain a polynomial function which passes through known points of a two-dimensional plane. For example, in a x-y plane, given are *n*+1 known points, (x_0, y_0), (x_1, y_1), …, (x_n, y_n). The method of Lagrange interpolation provides a formula for constructing a unique polynomial of degree *n* which passes through these *n*+1 points. Among them, the Lagrange fundamental polynomial, or interpolation basis function is expressed as follows:

$$\begin{aligned}\ell_j(x) &= \prod_{i=0, i\neq j}^{n} \frac{x-x_i}{x_j-x_i}\\ &= \left(\frac{x-x_0}{x_j-x_0}\right)\cdots\left(\frac{x-x_{j-1}}{x_j-x_{j-1}}\right)\left(\frac{x-x_{j+1}}{x_j-x_{j+1}}\right)\cdots\left(\frac{x-x_n}{x_j-x_n}\right),\\ &\quad 1 \leq j \leq n\end{aligned}$$

The specific point of $\ell_j(x)$ is the derived value 1 from x_j . Values from other points x_i $(i\neq j)$ equals 0, the expression of which is as follows: $\ell_j(x) = \begin{cases} 0, & i\neq j \\ 1, & i=j \end{cases}$

The Lagrange polynomial is $L(x) = \sum_{j=0}^{n} y_j \ell_j(x)$

That is the unique polynomial of degree *n* which passes through the points (x_0, y_0), (x_1, y_1), …, (x_n, y_n). For example, the binomial that passes through (4, 1), (5, 5), and (6, 10) when expressed in Lagrange basic polynomial is as follows:

$$\ell_1(x) = \left(\frac{x-5}{4-5}\right)\left(\frac{x-6}{4-6}\right), \quad \ell_2(x) = \left(\frac{x-4}{5-4}\right)\left(\frac{x-6}{5-6}\right),$$
$$\ell_3(x) = \left(\frac{x-4}{6-4}\right)\left(\frac{x-5}{6-5}\right)$$

By applying Lagrange interpolation polynomial, a single polynomial $L(x)$ can be obtained as expressed below:

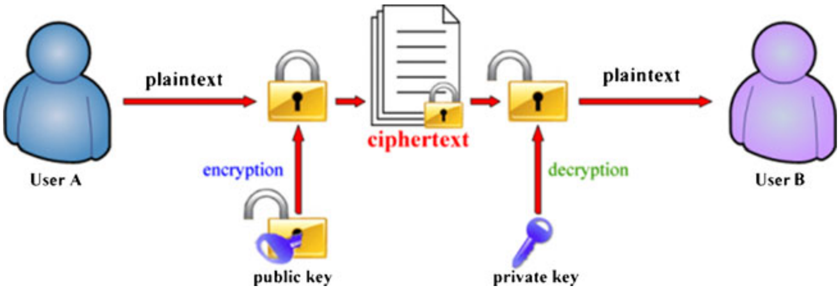
$$\begin{aligned}L(x) &= f(4)l(1) + f(5)l(2) + f(6)l(3)\\ &= 1\times\left(\frac{x-5}{4-5}\right)\left(\frac{x-6}{4-6}\right) + 5\times\left(\frac{x-4}{5-4}\right)\left(\frac{x-6}{5-6}\right)\\ &+ 10\times\left(\frac{x-4}{6-4}\right)\left(\frac{x-5}{6-5}\right)\\ &= \frac{1}{2}x^2 - \frac{1}{2}x - 5\end{aligned}$$

It can be inferred that $f(4) = 1$, $f(5) = 5$, $f(6) = 10$. By applying this formula predicted values can be derived, for example: to derive $f(18)$, substitute $x = 18$ in $L(x)$, and $L(18) = f(18) = 148$ is derived.

Proposed scheme

This study proposes a secure and effectively dynamic access scheme which allows users manage, access, or share Personal Health Record (PHR) in Cloud computing environments. In the environment, multi-users can access to PHR for appending, revision, deletion, and inquiry. Such multi-users present distinct access authorities that the access

Fig. 3 Public key cryptosystem



relationship is rather complicated. Patients can append PHR, such as the self-measured temperature and blood pressure. However, after appending the professional diagnosis information of doctors, patients can no longer revise it. In the medical treatment process, each patient might be diagnosed by various doctors because of different illnesses. Based on the professional medical field, the access authorities to patients' PHR would be distinct. Even the doctors in the same department are restricted the access to PHR. In addition to patients and doctors being able to manage PHR, other healthcare personnel could manage it as well. For instance, nurses can update some physiological information, pharmacists could inspect the past medication, cashiers could simply examine the drug record on the day, and other users with low-authorization can merely read some information, such as friends or researchers. In addition to medical personnel in general hospitals, PHR could also be accessed by multi-users for home care, remote care, and health management.

PHR scheme is patient-centered that individuals could maintain and record the health information. Besides, it is required to integrate personal medical information from various medical units that it used to access and provide personal health and medical records through the Internet or portable media. Presently, a lot of online PHR systems offer patients to manage personal medical records. However, as PHR is received from different places and patients could not ensure the contents being instantaneously updated or complete, the application of PHR has gradually transferred to store data in Cloud servers because of the emergence of Cloud computing.

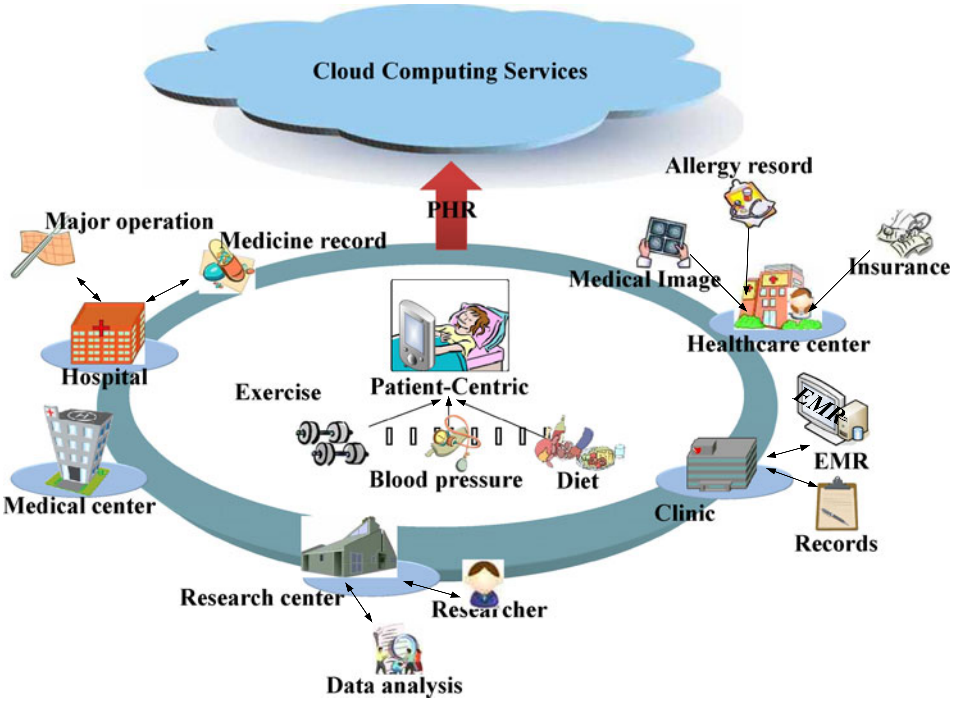
Since there are enormous users and complicated access control schemes in PHR scheme and users cannot ensure the data being immediately updated and complete, this study proposed to have PHR more efficiently provide numerous multi-users with dynamic access control scheme in Cloud servers, Fig. 4.

Having a fair Certification authority (*CA*) authorizes a superkey to each user, the superkey could be utilized to prove the user having a legal key and to verify the identity so as to ensure the security, authenticity, reliability, and completeness of information transmission. The central authority (*CA*) is considered as to build a structure for access control according to the relationships between the users. The proposed scheme consists of three phases, namely Initialization, Key generation and Derivation. The details of these phases are described in the following sub-sections.

Initialization

This study applies partially ordered access. A central authority (*CA*) builds the set-up for the partially ordered. A partially ordered set is a pair $(S, \preccurlyeq)$, where $\preccurlyeq$ appears a reflexive, antisymmetric, transitive binary relation with the set S. In this paper, users are divided into disjoint sets S_i for $i = 1, 2, \ldots, n$, which is a subset, called security classes. Each class presents personal authorization to access to the authorized files that he/she is authorized to obtain a decryption key for encrypted files. It is presented as $S_i = \{$u: u is the file ID of S_i with access authority$\}$, $n \in N$ and '$\preccurlyeq$' is a binary partial order relation over the set $S = \{S_1, S_2, \ldots, S_n\}$. For the set $(S, \preccurlyeq)$, $S_j \preccurlyeq S_i$ $(i, j \in N)$ means that the user in security class S_i can read or store the data held by the user in the security class Sj, but the opposite is not allowed. For example, each class has its own cryptographic key, $Sj = \{1, 2\}$, $Si = \{1, 2, 3\}$, $\{1, 2\} \preccurlyeq \{1, 2, 3\}$, then $Sj \preccurlyeq Si$.

Fig. 4 Access environment



For $Sj \preccurlyeq Si$, showing *Si* could receive the decryption key for the authorized *file1* and *file2* in *Sj*. There are a lot of users with different identities in PHR scheme, such as patients, doctors, pharmacist, nurses, or researchers and relatives of patients. Each user is represented the security class *Si* with personal superkey *Hi*, for $i = 1, 2, \ldots, n$. *CA* establishes a structure for these users, where there are *n* users which form two sets $S = \{S1, S2, \ldots, Sn\}$ and $H = \{H1, H2, \ldots, Hn\}$, as below:

S_1	S_2	…	S_i	…	S_n	
H_1	H_2	…	H_i	…	H_n	←secret & distinct

This PHR scheme is patient-centered and integrated with various healthcare records from different healthcare centers and health information established by distinct users. PHR of users is encrypted with a key to form an encrypted file being stored in Cloud servers. *CA* will build a structure that there are m files which form a set $file = \{file1, file2, \ldots, filem\}$, and *CA* generates a corresponding decryption key to each *fileu*, for $u = 1, 2, \ldots, m$. The encrypted files are protected by the key from being randomly accessed. The decryption key is shown as *DKu*, for $u = 1, 2, \ldots, m$.

$file_1$	$file_2$	…	$file_u$	…	$file_m$	
1	2	…	u	…	m	file ID, public
DK_1	DK_2	…	DK_u	…	DK_m	decryption keys, secret and distinct

A security class *Si* presents authorization to access to *fileu*, written as $Si = \{u$: u is the file ID of Si with access authority$\}$. For example $S1 = \{1, 2, 3, 4\}$, $S2 = \{1, 2, 3\}$, $\{1, 2, 3\} \preccurlyeq \{1, 2, 3, 4\}$, and then $S2 \preccurlyeq S1$. The following adjacency matrix can explain the access relationship. Assuming that there are six security classes and four files, put the {security classes}×{files} data in the two-dimensional array.

$$\begin{array}{c|cccc} & file_1 & file_2 & file_3 & file_4 \\ \hline S_1 & 1 & 1 & 1 & 1 \\ S_2 & 1 & 1 & 1 & 0 \\ S_3 & 0 & 1 & 1 & 1 \\ S_4 & 1 & 1 & 0 & 0 \\ S_5 & 0 & 1 & 1 & 0 \\ S_6 & 0 & 0 & 1 & 1 \end{array}$$

The indicate function $I(x, y)$ is defined to present user *i* with authorization to obtain DK_u for accessing to $file_u$.

$$I(x,y) = \begin{cases} 1, \text{if user } x \text{ has access to file} y \\ 0, \text{otherwise} \end{cases}$$

Variable *x* represents user's superkey *H* ID *i* and variable *y* represents *file* ID *u*. In each row, user *i* uses his secret superkey H_i to access to row *i*. Row *i*, by construction, contains the set of *file* ID's which user *i* is authorized to visit. For example, $I(3, 2) = 1$ because user 3 has access to $file_2$. $I(6, 1) = 0$ because user 6 has no access to $file_1$.

Key generation phase

Step 1: *CA* refers to the user *i* in $S = \{S1, S2, \ldots, Sn\}$ establishing individual and non-repeated superkey *Hi*, for $i = 1, 2, \ldots, n$ to keep *Hi* in secret.

Step 2: *CA* manages superkeys *Hi* of all users and sets indices for legal superkey *Hi*,

$$I_{\{H_1,\ldots,H_n\}}(x) = \begin{cases} 1 \quad , if \quad x \in \{H_1, \ldots, H_n\} \\ 0 \ , o.w. \end{cases} . I_{\{H_1,\ldots,H_n\}}(x)$$

means the indicate function of set $H = \{H1, H2, \ldots, H\text{n}\}$. The legality of *Hi* is verified by $I_{\{H_1,\ldots,H_n\}}(x)$.

Step 3: *CA* establishes function $Ai(x)$ for each user *i*. Let

$$A_i(x) = \left\{ \prod_{\substack{k=1 \\ k \neq i}}^{n} \frac{(x-H_k)}{(H_i-H_k)} \right\} \times I_{\{H_1,\ldots,H_n\}}(x),$$

for $i = 1, 2, \ldots, n$, $x \in R$.

Step 4: *CA* selects non-repeated random integers $\{DK_1, DK_2, \ldots, DK_m\}$(supposing there are *m* confidential files) as the decryption key for encrypting/decrypting confidential files. *CA* keeps DK_u in secret and publishes the public parameter *u*.

Step 5: *CA* sets $J_i = \{u$: $1 \le u \le m$, u is the file ID of S_i with access authority$\}$. There are n users for $i = 1, 2, \ldots, n$ and *m* files for $u = 1, 2, \ldots, m$. J_i is the set of *file* ID which user *i* is authorized to visit.

Step 6: *CA* sets the index $I_{J_i}(y) = \begin{cases} 1 \quad , if \quad y \in J_i \\ 0 \quad , o.w. \end{cases}$ to present user *i* with authorized access to DK_u and each user *i* establishes function $B_i(y)$, Let

$$B_i(y) = \left\{ \sum_{u \in J_i} DK_u \left[\prod_{\substack{t=1 \\ t \neq u}}^{m} \frac{(y-t)}{(u-t)} \right] \right\} \times I_{J_i}(y)^{,y,u,t \in R}$$

Step 7: *CA* establishes function $G(x,y) = \sum_{i=1}^{n} A_i(x)B_i(y)^{,x,y \in R.}$

That is $G(x,y) = A_1(x)B_1(y) + A_2(x)B_2(y) + \ldots + A_n(x)B_n(y) for(x,y) \in R \times R$ and declares it publicly.

Key derivation phase

Having established the key, user *i* could obtain DK_u by substituting personal superkey H_i and the ID *u* of $file_u$ with authorized access and further access to PHR by decrypting $file_u$ with DK_u. Such a method follows the following steps.

Step 1: User i substitutes personal superkey H_i into $I_{\{H_1,\ldots,H_n\}}(x) = \begin{cases} 1 & ,if \quad x \in \{H_1,\ldots,H_n\} \\ 0 & ,o.w. \end{cases}$. When the superkey H_i appears in the legal verification list of CA, $H_i \in \{H_1,\ldots,H_n\}$, then $I_{\{H_1,\ldots,H_n\}}(H_i) = 1$. When H_i of user i is not an authorized superkey in the list, $I_{\{H_1,\ldots,H_n\}}(H_i) = 0$.

Step 2: User i substitutes personal superkey H_i into $A_i(x) = \left\{ \prod_{\substack{k=1 \\ k \neq i}}^{n} \frac{(x-H_k)}{(H_i-H_k)} \right\} \times I_{\{H_1,\ldots,H_n\}}(x)$. When the personal superkey H_i of user i is legally verified in CA, the user substitutes $I_{\{H_1,\ldots,H_n\}}(x) = 1$ for calculation, and then $A_i(H_i) = 1$ and $A_i(H_k) = 0$ for $k \neq i$

Step3: User i substitutes $file_u$ ID u for $I_{J_i}(y) = \begin{cases} 1 & ,if \quad y \in J_i \\ 0 & ,o.w. \end{cases}$, $J_i = \{u$: $1 \leq u \leq m$, u is the file ID of S_i with access authority$\}$. When user i presents authorization to access to DK_u, $y \in J_i$ then $I_{J_i}(y) = 1$.

Step4: User i substitutes $file_u$ ID u for $B_i(y) = \left\{ \sum_{u \in J_i} DK_u \left[\prod_{\substack{t=1 \\ t \neq u}}^{m} \frac{(y-t)}{(u-t)} \right] \right\} \times I_{J_i}(y)$. When user i is authorized to access to DK_u, then $B_i(y) = DK_y$ if $y \in J_i$ and $B_i(y) = 0$ if $y \notin J_i$.

Step5: User i calculates $G(x,y) = \sum_{i=1}^{n} A_i(x)B_i(y)$. If $x \in \{H_1$, H_2, …, $H_n\}$ and $y \in J_{x.}$, $G(x, y) = DK_y$. The user could successfully obtain the decryption key, and $G(x, y) = 0$, otherwise.

Solution to key management of dynamic access problems

PHR scheme, a patient-centered structure, integrates the medical information of patients from various ends. Such information is store in Cloud servers to achieve the purpose of medical information integration and resources share and exchange. Cloud computing environments show the characteristics of easy expansion and resource share that it presents several advantages to satisfy the integration, share and exchange of PHR. In PHR scheme, the requirements of users to rapidly propose access request and receive permission from Cloud service providers should be satisfied.

The common situation is that different users would need to update the access authority with the change of events or time. For example, a car-accident patient is sent to an emergency ward. In addition to doctors proceeding primarily treatment, a conscious patient could propose his identity or an unconscious patient has documents to define the identity. When the doctor confirms the identity of the patient and sends requests to access to the patient's PHR through CA in Cloud center, Doctor S_i could successfully obtain the patient's PHR with private key H_i and read the personal information in PHR, such as hypertension or heart diseases. Such important information could provide doctors reference for clinical decision-making in emergency. Once the patient gets better and leaves the emergency ward, the doctor's authorization to access to PHR is automatically revoked. Not until the next accident, a different security class S_i could be added to the PHR scheme.

In terms of healthcare, patients would maintain and update PHR, such as blood pressure and diet habits, in addition to the medical information from hospitals. In other situations, personal medical records will be appended, revised, and deleted for different requirements, such as the authorization change of nurses, relatives, medical research units, and family doctors.

In this case, dynamic access schemes need to be established completely to ensure the instant and entire service of PHR. The key is the services provided by the PHR system being able to support distinct dynamic access demands so as to correspond to the data change of users and PHR in Cloud computing environments.

The proposed method is flexible that it could deal with all security management problems of dynamic keys, such as adding a new security class, removing an existing security class, and updating a user authorized. The involved solutions are simple, mainly addition and deduction, that it does not require enormous computation and storage space for parameter update. Regarding the grand formula $G(x, y)$ in section three,

$$G(x,y) = \sum_{i=1}^{n} A_i(x)B_i(y)$$

Function $A_i(x)$ is related to information verification for verifying the existence of H_i in the legal verification list of CA and the use of personal superkey for verification. Function $B_i(y)$ relates to PHR data verification for verifying the authorization of a user to obtain the decryption key DK_u to further decrypt the encrypted PHR data. The dynamic access requirements of PHR in Cloud are considered the users and PHR data.

(1) *Users are changeable*. Unlike static access model which could establish all user parameters in the beginning of access scheme, the constant increase or removal of PHR users and doctors, nurses, pharmacists, and various medical researchers could propose new requests for the patient-centered PHR system. User parameters need to be continuous updated to the initial access scheme to correspond to the dynamic users.

(2) *PHR files require appending and revision*. PHR integrates a patient's personal medical information from

different sources, such as the medical history, insurance message, allergy records, vaccination, past operations, recently measured blood pressure and blood glucose, and recently used drugs. In addition to the patient, authorized users with requests should be able to update the medical records and revise the documents in the PHR system. For this reason, the parameters in PHR message could be appended and removed with dynamic requests, after the establishment of access scheme.

In regard to the above considerations, the established grand formula $G(x, y)$ is nimble and flexible, which could be easily updated and revised the parameters instantaneously. The following section would explain grand formula $G(x, y)$ implementing the dynamic access scheme in the three situations: (1)Adding a new security class, (2)Removing an existing security class, (3)Updating a user authorized.

Adding a new security class

In case that S_v is a new security to be inserted into the user hierarchy; *CA* executes the procedure below for inserting the new security class S_v.

Step1: *CA* distributes the secret parameter Superkey H_v to a new security class S_v.

Step2: *CA* establishes $A_v(x)$. $A_v(x)$ is identical to that of $A_i(x)$ except that n is replaced by $n+1$, $A_v(x) = \prod_{\substack{v=1 \\ v \neq k}}^{n+1} \frac{x-H_k}{H_v-H_k}$. The index $I_{\{H_1,\dots,H_{n+1}\}}(x) = \begin{cases} 1 & ,if \quad x \in \{H_1,\dots,H_{n+1}\} \\ 0 & ,o.w. \end{cases}$ is updated.

Step3: *CA* establishes the parameter $J_i = \{u\text{: } 1 \leq u \leq m, u$ is the file ID of authorized $S_i\}$ for S_v

Step4: *CA* establishes $B_v(y)$, $B_v(y) = \left\{ \sum_{u \in J_v} DK_u \left[\prod_{\substack{t=1 \\ t \neq u}}^{m} \frac{(y-t)}{(u-t)} \right] \right\} \times I_{J_v}(y)$.

The index $I_{J_v}(y) = \begin{cases} 1 & ,if \quad y \in J_v \\ 0 & ,o.w. \end{cases}$ is updated.

Step5: *CA* updates formula $G(x, y)$ in the original scheme that the new formula appears

$$G'(x,y) = G(x,y) + A_v(x)B_v(y)$$

In the above process to append a user, *CA* simply updates the indices $I_{\{H_1,\dots,H_{n+1}\}}(x)$ and $I_{J_v}(y)$ and establishes $A_v(x)$, $B_v(y)$, J_v for the new security class S_v. The information is updated to formula $G(x, y)$. Few costs are required for computing the new security class S_v, and merely addition is required for updating the entire scheme.

【Example 2.1】

In this example, security class $S_1 \sim S_6$ and file$_1$ ~ file$_5$ have existed in the PHR scheme. Assume the new security class S_7 Family doctor being added in the PHR scheme and authorized to access to blood pressure, major operation, and drug allergy, as below.

First, *CA* would distribute Superkey H_7 to the family doctor and updates the indices as $I_{\{H_1,\dots,H_{n+1}\}}(x)$ and $I_{J_v}(y)$, according to authorization of the doctor for PHR. *CA* defines $J_7 = \{1, 3, 4\}$ for S_7 and establishes

$$A_7(x) = \left\{ \frac{(x-H_1)(x-H_2)(x-H_3)(x-H_4)(x-H_5)(x-H_6)}{(H_7-H_1)(H_7-H_2)(H_7-H_3)(H_7-H_4)(H_7-H_5)(H_7-H_6)} \right\} \times I_{\{H_1,\dots,H_7\}}(x)$$

$$B_7(y) = \left\{ DK_1 \times \frac{(y-2)(y-3)(y-4)(y-5)}{(1-2)(1-3)(1-4)(1-5)} + DK_3 \times \frac{(y-1)(y-2)(y-4)(y-5)}{(3-1)(3-2)(3-4)(3-5)} + DK_4 \times \frac{(y-1)(y-2)(y-3)(y-5)}{(4-1)(4-2)(4-3)(4-5)} \right\} \times I_{J_7}(y)$$

Finally, all parameters are updated to the new formula $G'(x,y) = G(x,y) + A_7(x)B_7(y)$

Removing an existing security class

Assuming that an existing security class S_v is to be removed from the PHR scheme, *CA* could precede the following algorithms.

Method 1: *CA* removes the relevant parameter $A_v(x)B_v(y)$ in the security class S_v from $G(x, y)$.

$$G'(x,y) = G(x,y) - A_v(x)B_v(y)$$

Method 2: J_v is defined as the set of *file* ID's which the user v is authorized to visit. Instinctively, CA updates J_v and deletes the authorization of the user.

$J_v' = \phi$ = empty set

【Example 2.2】

Assuming that S_7 Family doctor in the original scheme is no longer authorized, *CA* tends to remove S_7 from the scheme, as below:

CA could choose one of the following methods to remove S_7; one is to update formula $G'(x, y) = G(x, y) - A_7(x)B_7(y)$ to remove the relevant parameters in S_7 and the other is to update $J_7' = \phi$ so that S_7 could not pass the authorization verification.

Updating a user authorized

In the initial phase of PHR scheme, *CA* would establish the access authority for the security class S_i. When a user is updated the PHR authorization, *CA* would proceed the following steps.

Step1: *CA* resets $J_i' = \{u\text{: } 1 \leq u \leq m,\ u$ is the file ID of authorized $S_i\}$. J_i' presents the new authorization of S_i after update. When the authorization to PHR is changed, *CA* would re-calculate the adjacency matrix to generate a new set J_i.

Step2: *CA* updates $B_i(y)$ to $B_i'(y)$, as J_i is replaced by J_i' and the information of J_i is relevant with $B_i(y)$. Assuming that a new authorization of set J_i' is given to user *i*, then

$$G'(x,y) = G(x,y) - A_i(x)B_i(y) + A_i(x)B_i'(y)$$

According to the above steps, the establishment of J_i could easily updates the authorization of user *i* to access to PHR. When the user *i* does not present any authorization, $B_i(y)$ does not need to be updated, but just take $J_i' = \phi$ = empty set.

【Example 2.3】

Assuming that S_4 Medical researcher could access to *file*$_4$ Drug allergy in the original scheme, but no longer could after the research project being changed, a new authorization allows to access to *file*$_2$ Electrocardiogram, as below:

CA updates $J_4 = \{4\}$ to $J_4' = \{2\}$ and updates $B_4'(y)$.

$$B_4{}'(y) = \left\{ DK_2 \times \frac{(y-1)(y-3)(y-4)(y-5)}{(2-1)(2-3)(2-4)(2-5)} \right\} \times I_{J_4}(y)$$

$$\text{Then } G'(x,y) = G(x,y) - A_4(x)B_4(y) + A_4(x)B_4'(y)$$

In this dynamic access section, the construction and updating of $G(x, y)$ involve only simple arithmetic calculations. These can be done on a fly for a system consisting of millions of servers and millions of files. This scheme is easy to operate as the user *i* just enters a pair of valid (H_i, u) to get the correct DK_u. The system administrator calculates and updates $G(x, y)$ in the background in real time. Every server follows exactly the same operational steps to retrieve the correct decryption key.

Security analyses and discussion

In this section, a security analysis is performed to examine whether the proposed scheme is secure or not for practical applications. The analysis focuses upon four types of attack that may impact the system security.

Equation attack

Equation Attack: Attackers attempt to obtain the decryption key $\mathrm{DK_u}$ by utilizing public formula $\mathrm{G}(\cdot)$ for mathematical algorithms.

Equation Attack occurs in authorization updates when a user is removed but others remain unchanged that any attackers could obtain the decryption key DK_u by deducting the old public $G(\cdot)$ from the new public $G'(\cdot)$, $G'(\cdot) - G(\cdot) = 0$. The designed scheme could effectively resist *Equation Attack*. Three dynamic updates are proposed in section four.

1. Addition of a new security class $G'(x, y) = G(x, y) + A_v(x)B_v(y)$
2. Deletion of a current security class $G'(x, y) = G(x, y) - A_v(x)B_v(y)$
3. Updating of a user authorized $G'(x, y) = G(x, y) - A_i(x)B_i(y) + A_i(x)B_i'(y)$

When deducting the old public parameter $G(x, y)$ from the updated $G'(x, y)$ in any dynamic updates, attackers could merely obtain $A_v(x)B_v(y)$ or $A_i(x)B_i(y) + A_i(x)B_i'(y)$. $A_v(x)$ and $B_v(y)$ are the polynomial established by Lagrange interpolation, and they are finally multiplied to form $(n\text{-}1)(m\text{-}1)^{\text{th}}$ order polynomial with 2 unknowns.

$$A_v(x) = \left\{ \prod_{\substack{u=1 \\ u \neq v}}^{n} \frac{(x - H_u)}{(H_i - H_u)} \right\} \times I_{\{H_1,\ldots,H_n\}}(x)$$

$$= a_0 + a_1 x + \ldots + a_{n-1}x^{n-1}, n \in R$$

$$B_v(y) = \left\{ \sum_{j \in J_v} DK_u \left[\prod_{\substack{t=1 \\ t \neq u}}^{m} \frac{(y-t)}{u-t} \right] \right\} \times I_{J_v}(y)$$

$$= b_0 + b_1 x + \ldots + b_{m-1}x^{m-1}, m \in R$$

$$A_v(x)B_v(y) = a_0b_0 + a_1b_0x + a_0b_1y + a_1b_1xy\ldots$$
$$+ a_{n-1}b_{m-1}x^{n-1}y^{m-1}$$

Let $x = 0$ or $y = 0$, the attacker obtains the polynomial $A_v(x)B_v(y)$, which is just a series of disordered information. Compromising Attack therefore is ineffective in this method.

External attack

External Attack: Illegally authorized external personnel attempt to obtain the decryption key $\mathrm{DK_u}$ or decrypt for private medical information through public parameters.

Since personal medical records, health records, or physiological information are recorded in PHR, attackers often

Table 1 The defined symbol and parameter

Notation	Definition	Function
S_i	Security class, S_i = {u: u is the file ID of authorized S_i}, for $i = 1, 2, \dots, n$	To classify the security class of users
H_i	Superkey H_i, for $i = 1, 2, \dots, n$	To obtain the key authoring $file_u$
DK_u	Decryption key, for $u = 1, 2, \dots, m$	To decrypt the key of $file_u$
$file_u$	$File_u$, for $u = 1, 2, \dots, m$	The DK_u-encrypted file
$I_{\{H_1,\dots,H_n\}}(x)$	The indicate function of set $\{H_1, H_2 \dots, H_n\}$	To calculate whether H_i is in the legal verification list of *CA*
J_i	$J_i = \{u: 1 \le u \le m$, u is the file ID of authorized $S_i\}$	The set of files authorized by the users
$I_{J_i}(x)$	The indicate function of set J_i	To calculate whether the user presents authorized file set

tend to steal or sell such information that results in the loss of hospitals or users. The proposed PHR in Cloud computing environments covers numerous external users, in addition to the legal multi-users. Illegally authorized external personnel need to obtain the decryption key with the public parameters for useful patients' records or medical information that the encrypted medical files would become meaningful PHR after the decryption.

When an external attacker has the public parameter, most importantly the public formula $G(x, y)$, sufficient security should be emphasized, as there is a decryption key DK_u in the formula. In this method, each security class S_i could utilize private superkey H_i for obtaining the decryption key DK_u through the public function $G(x, y)$. An external attacker has to obtain the private key with Lagrange interpolation polynomial to acquire the decryption key DK_u. Since merely the public $G(x, y)$ and file ID u can be acquired, an external attacker cannot effectively apply mathematical algorithms to obtaining the private key DK_u because of too many unknowns. In this case, attackers cannot acquire medical information or patient's records through external attacks.

Moreover, any encryption/decryption methods could be selected by *CA* to establish DK_u, such as the symmetric key systems DES, 3DES, and AES. Based on diffusion and confusion, statistical methods would not decrypt the codes that they still present difficulty in decryption. As a result, attackers could not obtain the contents with the secret code.

Collaborative attack

Collaborative attack: Two or more legally authorized users collaboratively collect the private superkeys H_i and attempt to acquire the decryption key DK_j or the superkeys $H_i^{'}$ of other users.

In this study, partially ordered relationship appears in security class S_i. When S_i is authorized to access to S_j, it could be achieved simply by the same formula $G(x, y)$.

$$G(x,y) = A_1(x)B_1(y) + A_2(x)B_2(y) + \dots + A_n(x)B_n(y)$$

Consequently, two or more internal users tending to attack the other legal user is taken into account. Two cases are presented. Case1, the collaborative attackers appear partially ordered relationship with the attacked internal user. case2, the collaborative attackers do not present partially ordered relationship with the attacked internal user.

Case 1 The collaborative attackers, who are not authorized, attempt to collect the private *superkeys* H_i for obtaining the private key of the other authorized user. Based on Example 4.1, the collaborative attackers is authorized $S_3 = \{1, 4\}$, $S_4 = \{4\}$, while the attacked user is authorized $S_7 = \{1, 3, 4\}$. S_7 presents an additional authorization to access to $file_3$, comparing to S_3 and S_4 that S_3 and S_4 tend to

Table 2 The resulting after adding a new security class

	$file_1(Dk_1)$ Blood pressure	$file_2(DK_2)$ Electrocardiogram	$file_3(DK_3)$ Major operation	$file_4(DK_4)$ Drug allergy	$file_5(DK_5)$ Health insurance
$S_1(H_1)$: Patient	1	1	1	1	1
$S_2(H_2)$: Doctor	1	1	1	1	0
$S_3(H_3)$:nurses	1	0	0	1	0
$S_4(H_4)$: Medical researcher	0	0	0	1	0
$S_5(H_5)$: Health insurance unit	0	0	0	0	1
$S_6(H_6)$: Family	1	0	0	0	0
$S_7(H_7)$: Family doctor	1	0	1	1	0

Table 3 The resulting after revoking the existing current security class

	$file_1(Dk_1)$ Blood pressure	$file_2(DK_2)$ Electrocardiogram	$file_3(DK_3)$ Major operation	$file_4(DK_4)$ Drug allergy	$file_5(DK_5)$ Health insurance
$S_1(H_1)$: Patient	1	1	1	1	1
$S_2(H_2)$: Doctor	1	1	1	1	0
$S_3(H_3)$:nurses	1	0	0	1	0
$S_4(H_4)$: Medical researcher	0	0	0	1	0
$S_5(H_5)$: Health insurance unit	0	0	0	0	1
$S_6(H_6)$:Family	1	0	0	0	0

collaboratively attack S_7 to obtain the decryption key DK_3, whose data are stored in $A_7(x)B_7(y)$.

$$A_7(x) = \left\{ \frac{(x - H_1)(x - H_2)(x - H_3)(x - H_4)(x - H_5)(x - H_6)}{(H_7 - H_1)(H_7 - H_2)(H_7 - H_3)(H_7 - H_4)(H_7 - H_5)(H_7 - H_6)} \right\}$$

$$\times I_{\{H_1,\ldots,H7\}}(x)$$

$$B_7(y) = \left\{ DK_1 \times \frac{(y-2)(y-3)(y-4)(y-5)}{(1-2)(1-3)(1-4)(1-5)} + DK_3 \times \frac{(y-1)(y-2)(y-4)(y-5)}{(3-1)(3-2)(3-4)(3-5)} \right.$$

$$\left. + DK_4 \times \frac{(y-1)(y-2)(y-3)(y-5)}{(4-1)(4-2)(4-3)(4-5)} \right\} \times I_{J_7}(y)$$

Nevertheless, S_3 and S_4 merely have superkeys H_3, H_4, which cannot pass the verification of $A_7(x)$. With Lagrange interpolation, a null value will be received, and then $A_7(x)B_7(y) = 0 \times B_7(y) = 0$. Collaborative attacks therefore cannot acquire additional information, same as single attackers.

Case2: Although collaborative attackers do not appear partially ordered relationship with the attacked internal user, they collect the parameters to enhance the probability of getting the decryption key DK_u. Based on Example 4.1, the collaborative attackers are authorized $S_3 = \{1, 4\}$, $S_4 = \{4\}$, while the attacked user is authorized $S_5 = \{5\}$. There is no partially ordered relationship between S_5 and S_3, S_4. In order to obtain the authorization of S_5 to access to *file*$_5$, S_3 and S_4 attempt to collaboratively acquire the decryption key DK_5. Nonetheless, S_3 and S_4 simply have the superkeys H_3, H_4, which cannot pass $A_5(x)$ verification that a null value will be received.

Despite the partially ordered relationship between the collaborative attackers and the attacked user or the number of collaborative attackers, they cannot obtain the non-authorized DK_u by collecting the private *superkey* H_i.

Furthermore, attackers would tend to obtain the superkey H_i, in addition to the decryption key DK_u. However, they cannot succeed. From $A_7(x)$ in Case 1, S_3 and S_4 simply have the superkeys H_3, H_4, but not other useful information to acquired H_7 from $A_7(x)$ established in Lagrange interpolation. Collaborative attacks therefore cannot be operated in this method.

Reverse attack

Reverse attack: A legal internal attacker attempts to obtain other users' superkeys Hi' with the public formula G(x, y) and personal parameters.

Based on Example 4.1, legal users $S6$ and $S7$ could acquire the decryption key $DK1$ through $G(x, y)$. $S6$ and $S7$ appear partially ordered relationship, $S6 \preccurlyeq S7$ where $S6 = \{1\}$, $S7 = \{1, 3, 4\}$. For an attacker $S6$ tending to obtain the private parameter $H7$ of $S7$ with personal parameter $H6$ and the public parameter $G(x, y)$, he has to obtain $S7$ for accessing to *file3*, *file4*.

Table 4 The resulting after updating of a user authorized

	file$_1$(DK_1) Blood pressure	*file*$_2$(DK_2) Electrocardiogram	*file*$_3$(DK_3) Major operation	*file*$_4$(DK_4) Drug allergy	*file*$_5$(DK_5) Health insurance
$S_1(H_1)$: Patient	1	1	1	1	1
$S_2(H_2)$: Doctor	1	1	1	1	0
$S_3(H_3)$:nurses	1	0	0	1	0
$S_4(H_4)$: Medical researcher	0	1	0	0	0
$S_5(H_5)$: Health insurance unit	0	0	0	0	1
$S_6(H_6)$:Family	1	0	0	0	0

Table 5 Notation table

Definition	Notation
n	Number of the security classes
m	Number of the files
v_i	Degree of the polynomial f(x) (there are N security classes and each of them has v_i predecessors)
$\|p\|$	The bit-length of an integer p
$T_{I()}$	Time for performing an interpolating polynomial
T_{mul}	Time for performing a multiplication computation

In this method, a sole public formula is designed.

$$G(x,y) = A_1(x)B_1(y) + \ldots + A_6(x)B_6(y) + A_7(x)B_7(y)$$

S_6 replaces (H_6, 1) for the above polynomial point, while S_7 could compute the points (H_7, 1), (H_7, 3), (H_7, 4) to have *CA* authorize them the key. Nevertheless, substituting S_6 for point (H_6, 3) or point (H_6, 4) will not be able to normally acquire the decryption keys DK_3, DK_4 of *file*$_3$ and *file*$_4$.

S_6 tends to obtain the decryption keys DK_3, DK_4 of authorized S_7, it therefore attacks H_7 in $A_7(x)B_7(y)$ or DK_3, DK_4. S_6 therefore could substitute point (H_6, 1) for formula $G(H_6, 1) = DK_1$ that $G(H_6, 1) - DK_1 = 0$

$$\Rightarrow A_1(H_6)B_1(1) + \ldots + A_6(H_6)B_6(1) + A_7(H_6)B_7(1) + \ldots$$

$$+ A_n(H_6)B_n(1) - DK_1 = 0$$

$$\Rightarrow c_0d_{0+}c_1d_0x + c_0d_1y + c_1d_1xy\ldots + c_{n-1}d_{m-1}x^{n-1}y^{m-1} - DK_1 = 0$$

Accordingly, the formula $G(x, y)$ indeed is a (n-1)(m-1)th order polynomial with 2 unknowns. Attackers cannot recognize the items, which are contributed by $A_7(x)B_7(y)$, from the polynomial. Besides, the formula $G(x, y)$ is simply that it does not present abundant parameters for attackers. Even a single $A_7(x)B_7(y)$ is obtained, there is individual scheme to protect $A_7(x)$ and $B_7(y)$.

The information of superkey H_7 is stored in the polynomial $A_7(x)$ established by Lagrange interpolation.

$$A_7(x) = \left\{\frac{(x-H_1)(x-H_2)(x-H_3)(x-H_4)(x-H_5)(x-H_6)}{(H_7-H_1)(H_7-H_2)(H_7-H_3)(H_7-H_4)(H_7-H_5)(H_7-H_6)}\right\} \times I_{\{H_1,\ldots,H7\}}(x)$$

$A_7(x)$ would verify the input superkey H_i being in the legal verification list of *CA*. If it is not a CA-authorized internal user, it could not pass the calculation of indicate function $I_{\{H_1,\ldots,H_n\}}(x)$. On the other hand, if it is not a personal superkey H_7, the value of Lagrange interpolation would be 0.

The data of DK_3, DK_4 are stored in the polynomial B_7(y) established by Lagrange interpolation.

$$B_7(y) = \left\{DK_1 \times \frac{(y-2)(y-3)(y-4)(y-5)}{(1-2)(1-3)(1-4)(1-5)} + DK_3 \times \frac{(y-1)(y-2)(y-4)(y-5)}{(3-1)(3-2)(3-4)(3-5)} + DK_4 \times \frac{(y-1)(y-2)(y-3)(y-5)}{(4-1)(4-2)(4-3)(4-5)}\right\} \times I_{J_7}(y)$$

A user should also be authorized by CA to pass the verification of the indicate function $I_{J_i}(x)$, the set of $J_i = \{u$: 1 ≤ u ≤ m, u is the file ID of authorized $S_i\}$, or a null value would be acquired.

In such an attack, the polynomial cannot be reversed for illegal information that Equation attack can be effectively stopped.

Discussion

In this subsection, we want to discuss the computational overheads needed and the storage required in our scheme. Each parameter of the proposed scheme is defined in Table 1. Because of the dynamic access control scheme, the changes in circumstances can be seen in Table 2 when adding a new security class. Table 3 shows the changes in circumstances when revoking the existing current security class. The changes in circumstances are displayed in Table 4 when updating of a user authorized. Table 5 defines the parameters for analyzing the performance. The analysis of computation complexity is shown in Table 6.

The computation of interpolating polynomial had been quantified in Knuth [22]. Knuth pointed out that the process of interpolating at (n+1) points required (n^2+n)/2 divisions and (n^2+n) subtractions by Newton's formula, where n was the degree of the interpolating polynomial.

As to the evaluation of the polynomial for the derivation of the successor's secret parameters, Knuth [22] also figured out that this scheme needed (2n-1) multiplications and (2n) additions plus one modular operation by applying Horner's rule.

Regarding efficient computations, this scheme therefore required $2nT_{I()} + nT_{mul}$ to create $G(x, y)$ in the process of key generation, where T$_{I()}$ was the computation for interpolating

Table 6 Analysis of computation complexity

	Key generation/derivation	Storage of public parameters	Storage of private keys for each security class
The proposed scheme	$(\sum_{1\le i\le n} v_i + 3n)T_{I()} + 2nT_{mul}$	(m+1)|p|	|p|

polynomial, $T_{l()}$ = (2*n*-1) multiplications + (2*n*) additions + 1 modular operation, $\left(\sum_{1\le i\le n} v_i + n\right) T_{l()} + nT_{mul}$ was required computing in the process, and it totally spent $\left(\sum_{1\le i\le n} v_i + 3n\right) T_{l()} + 2nT_{mul}$. In regard to storage, the public parameters $G(x, y)$, u in this study required $(m+1)|p|$, and the storage for each security class of a private key H_i was $|p|$.

Conclusion

Under the patient-centered Personal Health Records (PHR) in Cloud computing environments, partial order relationship is applied to managing the users so that they could dynamically access to PHR with the individual authorization as well as remain the privacy for legal authorities to precede access control. Based on the key management scheme with Lagrange interpolation polynomial, it could accurately access to PHR and is suitable for enormous dynamic multi-users. In this method, the public formula $f_i(x)$ is integrated into a sole $G(x, y)$. Such a key management provides a better management in Cloud computing environments. The established formula $G(x, y)$ is flexible that it could instantaneously appending and deleting user authorization for appending and revising PHR during dynamic updates. Besides, the effect can be achieved merely by few additions that it provides faster and easier solutions. The following achievements are presented in this study.

(1) Patients could remain the right to completely access to PHR. The Patient-centered PHR allows patients to determine the access users and remove the outdated authorization.
(2) The access authority for various users could be precisely established. Doctors could merely access to their own patients. Once the patient is transferred, new access authority should be correctly transferred to the new doctor.
(3) The scheme could resist internal and external attacks, providing safer, more private and persistent heal management.
(4) The public parameters are merely $G(x, y)$ and u, and the generation of keys and the algorithms are simply. Users merely substitute personal parameter H_i and the public parameter u for $G(x, y)$ to obtain the decryption key.
(5) The solely public formula G(x, y) is convenient for the management of *CA*.
(6) Dynamic access control problems could be easily overcome.

In face of the threats of Cloud, a safer and more efficient access scheme is established for enhancing the reliability of PHR encryption, ensuring the security of users' medical information, reinforcing the dynamic access policy of each user, and protecting patients' privacy. Besides, the flexibly dynamic access control scheme for multi-users allows PHR being developed in Cloud computing.

Acknowledgment This work was supported partially by National Science Council of Republic of China under Grants NSC 101-2410-H-129 -001.

References

1. Committee on Quality of Health Care in America IoM, *Crossing the quality chasm*. National Academy Press, Washington, DC, 2001.
2. Kaelber, D. C., Jha, A. K., Johnston, D., Middleton, B., and Bates, D. W., A research agenda for personal health records. *J. Am. Med. Inform. Assoc.* 15(6):729–736, 2008.
3. Pagliari, C., Detmer, D., and Singleton, P., Potential of electronic personal health records. *Br. Med. J.* 335(7615):330–333, 2007.
4. National Research Council, *Networking health: prescriptions for the internet*. National Academy Press, Washington, DC, 2000.
5. AHIMA, AMIA, The value of personal health records: a joint position statement for consumers of healthcare. *J. Am. Med. Inform. Assoc.* 78(4):22–24, 2007.
6. Tang, P. C., Ash, J. S., Bates, D. W., Overhage, J. M., and Sands, D. Z., Personal health records: definitions, benefits, and strategies for overcoming barriers to adoption. *J. Am. Med. Inform. Assoc.* 13 (2):121–126, 2006.
7. Li, M., Yu, S., Ren, K., and Lou, W., "Securing Personal Health Records in Cloud Computing: Patient-centric and Fine-grained Data Access Control in Multi-owner Settings," *Security and Privacy in Communication Networks,* pp. 89-106, 2010.
8. Shortliffe, E. H., The evolution of electronic medical records. *Acad. Med.* 74:414–419, 1999.
9. Cimino, J. J., Socratous, S. A., and Clayton, P. D., Internet as clinical information system: application development using the world wide Web. *J. Am. Med. Informat. Assoc.* 2:273–284, 1995.
10. Schneider, J. H., Online personal medical records: Are they reliable for acute/critical care? *Soc. Crit. Care Med.* 29:196–201, 2001.
11. Department of Health and Human Services, Security and electronic signature standards. *Fed. Regist.* 63(155):43241–43243, 1998.
12. Google health, Available: http://www.google.com/intl/en-US/health/about/index.html
13. Microsoft health Vault, Available:http://www. healthvault.com/Personal/index.html
14. US Public Law, ""Health Insurance Portability and Accountability Act of 1996," 104th Congress, Public Law 104–191, 1996.
15. Yanga, C. M., Lina, H. C., Changb, P., and Jianc, W. S., Taiwan's Perspective on electronic medical Records' security and privacy protection: lessons learned from HIPAA. *Comput. Meth. Programs. Biomed.* 82:277–282, 2006.
16. Qualys On Demand Vulnerability Management, "CASE STUDY: Geisinger Health System—Bringing HIPAA Compliance to an Electronic Medical Record System," http://www.qualys.com/docs/geisinger.pdf
17. "Meeting HITECH's Challenge to the Health Care Industry," An Oracle White Paper, May 2010.
18. Atluri, V., and Huang, W., "An Authorization Model for Workflows," *Proceedings of the Fourth European Symposium on Research in Computer Security*, pp. 25-27, 1996.

19. Barkley, J. F., Ferraiolo, D. F., and Kuhn, D. R., “A role based access control model and reference implementation within a corporate intranet”. *ACM Trans. Inform. Syst.Secur. (TISSEC)* 2:34–64, 1999.
20. Botha, R., “CoSAWoE – A Model for Context-sensitive Access Control in Workflow Environments,” *South Africa computer journal*, 2001.
21. Coyne, E., Fenstein, H., Sandhu, R., and Youman, C., Role-based access control models. *IEEE Computer* 29(2):38–47, 1996.
22. Denning, D. E., “Cryptographic Checksums for Multilevel Database Security,” *Proceedings of the 1984 IEEE Symposium on Security and Privacy*, pp. 52–61, 1984.
23. Bardram, J. E., Pervasive healthcare as a scientific discipline. *Methods. Inform. Med.* 47:129–142, 2008.
24. http://www.tafm.org.tw/data/012/meeting/209.pdf
25. US Department of Health and Human Services, “Personal Health Records and Personal Health Record Systems,” *National Committee on Vital and Health Statistics,* pp. 15, 2006.
26. Vaquero, L. M., Rodero-Merino, L., Caceres, J., and Lindner, M., A break in the clouds: towards a cloud definition. *ACM SIGCOMM Comput. Comm.* 39(1):50–55, 2008.
27. Mell, P. and Grance, T., “The NIST Definition of Cloud Computing,” *National Institute of Standards and Technology.* 2009.
28. Brunette, G. and Mogull, R., “Security Guidance for Critical Areas of Focus in Cloud Computing V2.1,” *Cloud Security Alliance*, 2009.
29. Gens, F., “"New IDC IT Cloud Services Survey: Top Benefits and Challenges,” *IDC eXchange*, 2009
30. Minister of Justice, “Personal Information Protection and Electronic Documents Act (PIPEDA),” 2011.
31. Benaloh,J., Chase,M., Horvitz,E., and Lauter, K., “Patient Controlled Encryption: Ensuring Privacy of Electronic Medical Records,” *In Proceedings of the ACM workshop on Cloud computing security*, pp. 103-114, 2009.
32. Stalling,W., “Network and Network Security – Principles and Practice,” *Prentice Hall International Edition*, pp. 1-14, 1995.
33. Stallings, W., “Cryptography and Network Security, Principles and Practice,” *Prentice Hall*, 2003
34. AIM (Advance Informatics in Medicine), “Secure Environment for Information Systems in Medicine,” SEISMED (A2033)/SP14/HILD/05.07. 95.
35. Shamir, A., “Identity-based Cryptosystems and Signature Schemes,” *Advances in Cryptology-Proceedings of CRYPTO’84*, *Springer-Verlag LNCS 196*, pp.47-53, 1985.
36. National Bureau of Standards, FIPS pub. 46, “Data Encryption Standard,” *US Department of Commerce*, January 1977.
37. Lai, X., and Massey, J., “"A proposal for a New block encryption standard”, Proceedings of Eurocrypt’91, Springer-Verlag. *LNCS* 473:389–404, 1991.
38. Miller, V., “Use of Elliptic Curves in Cryptography”, Advances in Cryptology-Crypto’85. *LNCS* 218:417–426, 1985.
39. Rivest, R., Shamir, A., and Adleman, L., A method for obtaining digital signatures and public-Key cryptosystems. *Commun. ACM* 21(2):120–126, 1978.
40. ElGamal, T., “A Public-Key Cryptosystem and a Signature Scheme based on Discrete Logarithms”, Advances in Cryptology-Crypto’85, Springer-Verlag. *LNCS* 196:10–18, 1985.
41. Koblitz, N., Elliptic curve cryptosystems. *Math. Comput.* 48:203–209, 1985.