PSCPA: Patient Self-controllable Privacy-preserving Cooperative Authentication in Distributed m-Healthcare Systems

Jun Zhou, Zhenfu Cao
Department of Computer Science and Engineering
Shanghai Jiao Tong University
Shanghai 200240, China
Email: zhoujun_tdt@sjtu.edu.cn; zfcao@cs.sjtu.edu.cn

***Abstract*—Distributed m-healthcare systems significantly facilitate efficient patient treatment of high quality, while bringing about the challenge of keeping both the confidentiality of the personal health information and the patients' identity privacy simultaneously. It makes many existing data access control and anonymous authentication schemes inefficient in distributed m-healthcare systems. To solve the problem, in this paper, a novel authorized accessible privacy model (AAPM) is established. Patients can authorize physicians by setting an access tree supporting flexible threshold predicates. Then, based on it, a patient self-controllable privacy-preserving cooperative authentication scheme (PSCPA) realizing three levels of security and privacy requirement in distributed m-healthcare system is proposed. The directly authorized physicians can both decipher the personal health information and authenticate patients' identities by satisfying the access tree with their attribute sets. Due to the indistinguishability of the transcript simulation from the patients and physicians for the indirectly authorized physicians, they can only decipher the personal health information rather than authenticate patients' identities. The unauthorized persons can obtain neither. Moreover, PSCPA is extended in emergent cases and to resist Denial of Service (Dos) attacks. Finally, the formal security proof and simulation results show our scheme far outperforms the previous ones in terms of computational, communication and storage overhead.**

***Keywords*-Authentication; access control; privacy-preserving; security; distributed m-healthcare system**

I. Introduction

Distributed m-healthcare systems have been increasingly adopted by the European Commission activities, the US Health Insurance Portability and Accountability Act (HIPAA) and many other governments for efficient and high-quality treatment [1-3]. The personal health information is always shared among the patients suffering from the same disease, between the patients and physicians as equivalent counterparts or even across distributed healthcare providers for medical consultant [28], [29]. This kind of personal health information sharing allows each collaborating healthcare provider to process it locally with higher efficiency and scalability, greatly enhances the treatment quality, significantly alleviates the complexity at the patient side and therefore becomes the preliminary component of a distributed m-healthcare system.

However, it also brings about a series of challenges, especially how to ensure the security and privacy of the patients' personal health information from various attacks in the wireless communication channel such as eavesdropping and tampering [5], [26].

As to the security facet, we mean the access control of personal health information, namely it is only the authorized physicians or institutions that can recover the patients' personal health information during the data sharing in the distributed m-healthcare system. In practice, most patients are concerned about the confidentiality of their personal health information since it is likely to make them in trouble for each kind of unauthorized collection and disclosure. For example, the patients' insurance application may be rejected once the insurance company has the knowledge of the serious health condition of its consumers. Therefore, in distributed m-healthcare systems, which part of the patients' personal health information should be shared and which part of physicians should their personal health information be shared with have increasingly become two intractable problems demanding urgent solutions. There has emerged various research [8-11, 15, 16, 18, 19] focusing on it such as a fine-grained distributed data access control scheme [9] using the technique of attribute based encryption and a rendezvous-based access control method [10] providing access if and only if the patient and the physician meet in the physical world. Unfortunately, the problem of simultaneously protecting patients' privacy was left unsolved.

In this paper, we consider achieving these two goals with high efficiency. In distributed m-healthcare systems, all the members can be classified into three categories: the directly authorized physicians who are authorized by the patients, the indirectly authorized physicians who are authorized by the directly authorized physicians for medical consultant or research purpose (i.e. since they are not authorized by the patients, we use the term 'indirectly authorized' instead), and the unauthorized persons. The patient's identity can only be authenticated by the patient directly authorized physicians. When patients' personal health information tends to be transferred by directly authorized physicians and shared among distributed healthcare providers or research institutions for medical consultation or scientific research, the identity privacy of the patients should be well protected since only the personal health information is required for these tasks. In this paper, by extending the techniques of attribute based access control [22] and designated verifier signatures [21] on de-identified health information [27], we realize three different levels of privacy-preserving requirement: only the physicians directly authorized by the patients can access the patients' personal health information and authenticate their identities simultaneously; the physicians and research staff indirectly authorized by patients cannot authenticate the patients' identities but recover the personal health information; while the unauthorized persons can obtain neither. The main contributions of this paper are summarized as follows.

(1) A novel authorized accessible privacy model (AAPM) for the privacy-preserving cooperative authentication is established to allow the patients to authorize the corresponding physicians by setting an access tree supporting flexible threshold predicates.

(2) Based on AAPM, a patient self-controllable privacy-preserving cooperative authentication scheme (PSCPA) in the distributed m-healthcare system is proposed, realizing three different levels of security and privacy requirement for the patients.

(3) The efficiency analysis and simulation results show that our scheme far outperforms the previous constructions in terms of computational, communication and storage overhead.

The rest of this paper is organized as follows. We discuss related work in the next section. In Section III, the network model of a distributed m-healthcare system is illustrated. We provide some backgrounds and preliminaries required throughout the paper in Section IV. In Section V, we establish a novel authorized accessible privacy model (AAPM). Based on it, we propose a patient self-controllable privacy-preserving cooperative authentication scheme (PSCPA) in the distributed m-healthcare system in Section VI. In Section VII, we give the security proof and performance evaluations of the proposed scheme. Finally, Section VIII concludes the paper.

II. Related Work

Besides the constructions for authorized access control of patients' personal health information [8-11, 15, 16, 18, 19] we mentioned above, there exist anonymous identification schemes by pseudonyms and other privacy-preserving techniques [4, 10-14, 17, 20, 23, 25]. Lin et. al. proposed SAGE achieving not only the content oriented privacy but also the contextual privacy against a strong global adversary [12]. Sun et. al. proposed a solution to privacy and emergency responses based on anonymous credential, pseudorandom number generator and proof of knowledge [11, 13]. Lu et. al. proposed a privacy-preserving authentication scheme in anonymous P2P systems based on Zero-Knowledge Proof [14]. However, the heavy computational overhead of Zero-Knowledge Proof makes it cannot be directly applied to the distributed m-healthcare systems where the computational resource for both patients and physicians is limited. Riedl et. al. presented a new architecture pseudonymiaztion of information for privacy in E-health (PIPE) [25]. Slamanig et. al. integrated pseudonymization of medical data, identity management, obfuscation of metadata with anonymous authentication to prevent disclosure attacks and statistical analysis in [26] and suggested a secure mechanism guaranteeing anonymity and privacy in both the personal health information transferring and storage at a healthcare provider [7]. Schechter et. al. proposed an anonymous authentication of membership in dynamic groups [6]. However, since the anonymous authentication mentioned above [6], [7] are established based on public key infrastructure (PKI), the need of an online certificate authority (CA) and one unique public key encryption for each symmetric key k for data encryption at the portal of authorized physicians made the overhead of the construction grow linearly with size of the group. Furthermore, the anonymity level is dependent on the size of the anonymity set making the anonymous authentication impractical in specific surroundings where the patients are sparsely distributed.

In this paper, our proposed authorized accessible privacy model (AAPM) and the patient self-controllable privacy-preserving authentication scheme (PSCPA) are proposed by extending the traditional designated verifier signature to an attribute based counterpart. The security and anonymity level is significantly enhanced by associating it to GBDH problem and the number of patients' attributes to deal with the privacy leakage in patient sparsely distributed scenarios in [6, 7]. Meanwhile, our construction cost is linear to the number of attributes rather than the physicians in healthcare providers. Therefore, it better adapts to the distributed m-healthcare systems where the number of physicians is great and the patients need the timely responses from the healthcare providers.

Last but not least, it is noticed that our construction essentially differs from the trivial combination of attribute based encryption [22] and designated verifier signature [21]. As the simulation results shows, we achieve the functionalities of both access control for personal health information and anonymous authentication for patients simultaneously with the efficiency significantly less than the trivial combination of the two building blocks above. Therefore, our PSCPA far outperforms the previous schemes [21, 22] in access control for patients' personal health information and [6, 7] in realizing privacy-preserving cooperative authentication in distributed m-healthcare systems.

III. Network Model

The basic e-healthcare system consists of three components: BANs, wireless transmission networks and the healthcare providers [1], [2]. Body sensor networks consist of various kinds of sensors monitoring and collecting all personal health information to the patient hand-held mobile device. The wireless transmission networks transfer personal health information to the physicians in healthcare providers. The healthcare provider consists of physicians and the patient information database (PIDs) [26]. Authorized physicians can access their corresponding patients' personal health information and authenticates their identities. The basic architecture of the E-healthcare system is illustrated in Fig. 1.

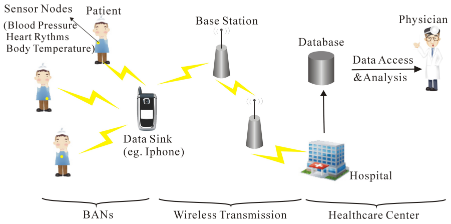

Fig. 1. An Basic Architecture of the E-health System

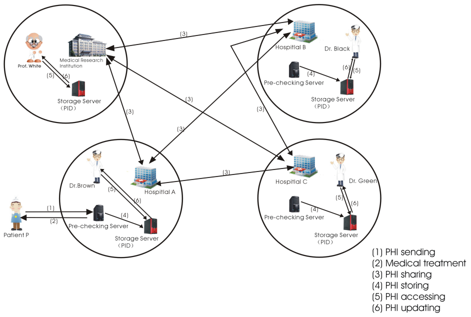

Fig. 2. An Overview of Our Distributed m-Healthcare System

Then, we further illustrate the unique characteristics of distributed m-healthcare systems where all the personal health information can be shared among patients, authorized physicians, distributed healthcare providers and medical research institutions. A distributed m-healthcare system model is shown in Fig. 2. There are three distributed healthcare providers A, B, C and the medical research institution D, where Dr. Brown, Dr. Black, Dr. Green and Prof. White are working respectively. It is assumed that patient P is registered in hospital A and Dr. Brown is one of his directly authorized physicians. For medical consultant or other research purpose, it is likely for Dr. Brown to share patient P's personal health information among hospital A, B, C and the research institution D.

IV. Preliminaries

A. Basic Concepts on Bilinear Pairings

Let $\mathbb{G}_0$ be a cyclic additive group generated by g, whose order is a prime p. Let $\mathbb{G}_1$ be a cyclic multiplicative group with the same order p. Let $e : \mathbb{G}_0 \times \mathbb{G}_0 \rightarrow \mathbb{G}_1$ be a bilinear mapping with the following properties.

(1) Bilinearity: for all $u, v \in \mathbb{G}_0$ and $a, b \in \mathbb{Z}_p$, we have $e(u^a, v^b) = e(u, v)^{ab}$.

(2) Non-degeneracy: $e(g, g) \neq 1$.

We say $\mathbb{G}_0$ is a bilinear group if the group operations in $\mathbb{G}_0$ and the bilinear map $e : \mathbb{G}_0 \times \mathbb{G}_0 \rightarrow \mathbb{G}_1$ are both efficiently computable. Notice that the map e is symmetric since $e(g^a, g^b) = e(g, g)^{ab} = e(g^b, g^a)$.

The related complexity assumptions are as follows.

(1) **Bilinear Diffie-Hellman Problem (BDHP)**. Given g as a generator of $\mathbb{G}_0$ as well as g^a, g^b, g^c for unknown randomly chosen $a, b, c \in \mathbb{Z}_p^*$, compute $e(g, g)^{abc}$. $\mathbb{G}_0$ and $\mathbb{G}_1$ are groups in which there are no known algorithms for efficiently solving the Diffie-Hellman problem in either $\mathbb{G}_0$ or $\mathbb{G}_1$.

(2) **Decisional Bilinear Diffie-Hellman Problem (DBDHP)**. Given g as a generator of $\mathbb{G}_0$ as well as g^a, g^b, g^c for unknown randomly chosen $a, b, c \in \mathbb{Z}_p^*$ and $h \in \mathbb{G}_1$, decide whether $h = e(g, g)^{abc}$.

(3) **Gap Bilinear Diffie-Hellman Problem (GBDHP)**. Given g as a generator of $\mathbb{G}_0$ as well as g^a, g^b, g^c for unknown randomly chosen $a, b, c \in \mathbb{Z}_p^*$, compute $e(g, g)^{abc}$ with the help of the DBDH oracle.

V. Authorized Accessible Privacy Model

In this section, we propose a novel authorized accessible privacy model (AAPM) for distributed m-healthcare systems. The basic idea of AAPM embraces two folds. On one hand, the patient can authorize the associated physicians by setting an access tree supporting flexible threshold predicates. On the other hand, only the directly authorized physicians are allowed to correctly authenticate the identity of the patient by satisfying the access tree with their own attribute sets respectively. AAPM consists of the following two components: an attribute based designated verifier signature scheme (ADVS) and the corresponding adversary models.

A. Attribute based Designated Verifier Signature Scheme

In an attribute based designated verifier signature scheme, there are three entities: a central attribute authority, the patient and the physicians in the healthcare provider. Each physician owns a set of attributes associated to his intrinsic characteristics or expertise such as 'GENDER=MALE', 'AFFILIATION=REDCROSS HOSPITAL', 'PROFESSIONAL=ANGIOCARDIOPATHY' and 'RANKS=PROFESSOR'. The authority takes charge of issuing private keys corresponding to each attribute the physicians possesses. Then, a patient suffering from heart diseases can sign his personal health information with such a specific access structure that only the physicians satisfying 'PROFESSIONAL=ANGIOCARDIOPATHY' and 'RANKS=(at least two of) PROFESSOR, CHIEF PHYSICIAN, OFFICER' can decipher the personal health information and verify his authentic identity. Fig. 3 illustrates the aforementioned access structure in the distributed m-healthcare system. By extending the concepts of attribute-based signature [23] and designated verifier signature [21], our scheme consists of four algorithms: **Setup**, **Key Extraction**, **Sign** and **Verify**. Denote the universe of attributes as U. We say an attribute set ω satisfies a specific access structure $\mathbb{A}$ if and only if $\mathbb{A}(\omega) = 1$ where ω is chosen from U. The algorithms are defined as follows.

Setup. On input 1^l, where l is the security parameter, this algorithm outputs public parameters and y as the master key for the central attribute authority.

Key Extract. Suppose that a physician requests an attribute set $\omega_D \in U$. The attribute authority computes sk_D for him if he is eligible to be issued with sk_D for these attributes.

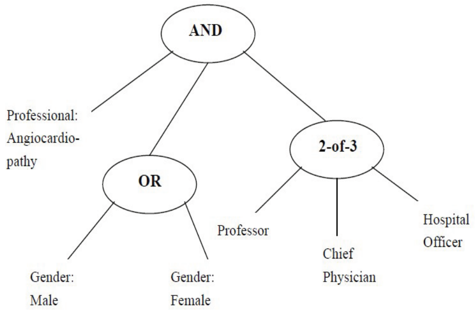

Fig. 3. An Example Access Structure in Our Distributed m-Healthcare System

Sign. A deterministic algorithm that uses the patient's private key, the uniform public key of the healthcare provider where the physicians work and a message m to generate a signature σ. That is, $\sigma \leftarrow Sign(sk_P, pk_D, m)$.

Verify. Assume a physician wants to verify a signature σ with an access structure $\mathbb{A}$ and possesses a subset of attributes $\omega_J \subseteq \omega_D$ satisfying $\mathbb{A}(\omega_J) = 1$, a deterministic verification algorithm can be operated. Upon receiving a message m and a signature σ, he takes as input his attribute private key sk_D and the patient's public key pk_P, then returns $True$ if the signature is correct, or $\perp$ otherwise. That is, $\{True, \perp\} \leftarrow Verify(sk_D, pk_P, m, \sigma)$.

Transcript Simulation Generation. We require that the directly authorized physicians who hold the private key $sk_D^{authorized}$ can always produce identically distributed transcripts indistinguishable from the original protocol via the **Transcript Simulation** algorithm.

In addition to the main algorithms described above, we also require the following properties.

Correctness. All signatures generated correctly by **Sign** would pass **verify** operated by the directly authorized physicians,

$$Pr[True \leftarrow Verify(sk_D^{authorized}, pk_P, m, Sign(sk_P, pk_D, m))] = 1. \tag{1}$$

B. Adversary Models

(1) Unforgeability. In an attribute based designated verifier signature scheme, as to unforgeability, we mean that the adversary wants to forge a signature w.r.t an unsatisfied verifier's specific access structure. The definition of unforgeability allows an adversary not to generate an effective signature with an access structure $\mathbb{A}^*$ for the verifiers if he has not queried the private key for ω^* or any superset of it such that $\mathbb{A}^*(\omega^*) = 1$, or he has not queried the signature on the forged message m^* with an access structure $\mathbb{A}^*$ such that $\mathbb{A}^*(\omega^*) = 1$. We provide a formal definition of existential unforgeability of PSCPA under a chosen message attack. It is defined using the following game between an adversary $\mathscr{A}$ and a simulator $\mathscr{B}$.

Initial Phase. $\mathscr{A}$ chooses and outputs a challenge access structure $\mathbb{A}^*$ that will be included in the forged signature.

Setup Phase. After receiving the challenge access structure $\mathbb{A}^*$, $\mathscr{B}$ selects a proper security parameter 1^l, runs the **Setup** algorithm to generate key pairs (sk, pk), sends pk and other public parameters to the adversary $\mathscr{A}$ and remains the private key sk secretly.

Query Phase. After receiving the public parameters, $\mathscr{A}$ can operate a polynomially bounded number of queries on ω_D and $(m, \mathbb{A}^*)$ to the key extraction oracle and the signing oracle between the patient and the corresponding physician at most q_k, q_s times respectively. $\mathscr{B}$ answers with sk_D and σ as the responses. As to the verifying queries, $\mathscr{A}$ can request a signature verification on a pair (m, σ) between the patient and the directly authorized physicians at most q_v times. In respond, $\mathscr{B}$ outputs $True$ if it is correct, or $\perp$ otherwise.

Forgery Phase. Finally, the adversary $\mathscr{A}$ outputs a signature σ^* on messages m^* with respect to $\mathbb{A}^*$ which is the challenge access structure sent to $\mathscr{B}$ during the initial phase. The forged signature must satisfy the following three properties.

(1) $\mathscr{A}$ did not send queries of the attribute set $\omega_D \subseteq \omega^*$ satisfying $\mathbb{A}^*(\omega_D) = 1$ to the key extraction oracle.

(2) $(m^*, \mathbb{A}^*)$ has not been queried to the signing oracle between the patient P and the corresponding physician D.

(3) σ^* is a valid signature of the message m^* between the patient P and the corresponding physician D.

Definition 1. Assume the probability of an adversary $\mathscr{A}$ to win the game is $Succ^{EFCMA}_{PSCPA,\mathscr{A}(t,q_{H_0},q_{H_1},q_k,q_s,q_v)}(l)$. We say that PSCPA is existentially unforgeable under a chosen message attack if the probability of success of any polynomially bounded adversary $\mathscr{A}$ running in time at most t and making at most $q_{H_0}, q_{H_1}, q_k, q_s, q_v$ queries to the random oracle $\mathscr{H}_0$, $\mathscr{H}_1$, key extraction oracle, signing oracle and the verifying oracle in the game described above is negligible. Namely

$$Succ^{EFCMA}_{PSCPA,\mathscr{A}(t,q_{H_0},q_{H_1},q_k,q_s,q_v)}(l) \leq \epsilon. \tag{2}$$

(2) Anonymity for the Patient. To guarantee a strong privacy for the patient, the signature reveals nothing about the identity of the patient except the information explicitly revealed. Its formal definition is described as follows.

Definition 2. A PSCPA scheme satisfies the property of patient privacy if for any two attribute sets ω_0, ω_1 w.r.t identities ID_0, ID_1, a message m and a signature σ on predicate $\mathbb{A}$ satisfying $\mathbb{A}(\omega_0) = \mathbb{A}(\omega_1) = 1$, any adversary $\mathscr{A}$, even with unbounded computational ability cannot identify which attribute set is utilized to generate the signature with the probability better than random guessing. Namely, $\mathscr{A}$ can only correctly output the identity generating the signature with probability no better than $\frac{1}{2}$ even the adversary $\mathscr{A}$ has access to the directly authorized physicians' private keys.

VI. PSCPA DESIGN

A. PSCPA Construction

Access Tree $\mathscr{T}$. Let $\mathscr{T}$ be a tree representing an access structure. Each non-leaf node of the tree represents a threshold gate, described by its children and a threshold value. If

$number_x$ is the number of children of a node x and k_x is its threshold value, then $0 < k_x \leq number_x$. When $k_x = 1$, the threshold gate is an OR gate and when $k_x = number_x$, it is an AND gate. Each leaf node x of the tree is described by an attribute and a threshold value $k_x = 1$. To facilitate working with the access tree, we define a few functions. First, we denote the parent of the node x in the tree by $parent(x)$. The function ω_x is defined only if x is a leaf node and denotes the attributes associated with the leaf node x in the tree. The access tree also defines an ordering between the children of every node, that is, the children of a node x are numbered from 1 to $number_x$. The function $index(x)$ returns such a number associated with the node x, where the index values are uniquely assigned to nodes in the access tree for a given key in an arbitrary manner.

Satisfying an access tree. Let $\mathscr{T}$ be an access tree with root r. Denote by $\mathscr{T}_x$ the subtree of $\mathscr{T}$ rooted at the node x. Therefore, $\mathscr{T}$ is the same as $\mathscr{T}_r$. If a set of attribute β satisfies the access tree $\mathscr{T}_x$, we denote it as $\mathscr{T}_x(\beta) = 1$. We compute $\mathscr{T}_x(\beta)$ recursively as follows. If x is a non-leaf node, evaluate $\mathscr{T}_{x'}(\beta)$ for all children $x^{'}$ of node x. $\mathscr{T}_x(\beta)$ returns 1 if and only if at least k_x children return 1. If x is a leaf node, then $\mathscr{T}_x(\beta)$ returns 1 if and only if $\omega_x \in \beta$.

Before describing our scheme, we review Lagrange interpolation in advance. Recall that, given d points $q(1), q(2), \cdots, q(d)$ on a $d-1$ degree polynomial, we can use Lagrange interpolation to compute $q(i)$ for any $i \in \mathbb{Z}_p$. Let S be a set of d elements. We can define the Lagrange coefficient $\Delta_{j,S}(i)$ of $q(j)$ in the computation of $q(i)$ as follows

$$\Delta_{j,S}(i) = \prod_{\eta \in S, \eta \neq j} \frac{i-\eta}{j-\eta}. \tag{3}$$

The notations used in our scheme are illustrated in Table I.

TABLE I
NOTATIONS IN OUR SCHEME

Notation	Description
d_x	threshold for node x in access tree $\mathscr{T}$
k_x	number of attributes required to be owned by the patient w.r.t. node x
$q_x(\cdot)$	$D_x = d_x - 1$-degree polynomial assigned to node x
ψ_x	a default attribute set of size $d_x - 1$ for node x
$sk_D^{ephemeral}$	uniform private key of the healthcare center
$pk_D^{ephemeral}$	uniform public key of the healthcare center
ω_D	the set of attributes owned by the physician
sk_D	private key of the physician
ω_x^*	attributes in predicate of node x for physicians
$sk_P^{ephemeral}$	private key of the patient
$pk_P^{ephemeral}$	public key of the patient
$\psi_x^{'}$	a subset of default attribute set of size $d_x - k_x$ chosen by the patient
K_{Enc}/K_{Dec}	symmetric key for message encryption/decrption
K_{Sig}	signing key for ADVS
ω_J	the subset of physician's attribute set of size k_x chosen to satisfy the predicate
H_0, H_1, H_2	hash functions mapping $\{0,1\}^* \rightarrow \mathbb{G}_0$, $\{0,1\}^* \rightarrow \mathbb{Z}_p^*$ and $\mathbb{G}_1 \rightarrow \{0,1\}^{k_{Enc}}$

Setup: Let $\mathbb{G}_0$ be a bilinear group of prime order p and g be a generator of $\mathbb{G}_0$. Construct a bilinear map $e : \mathbb{G}_0 \times \mathbb{G}_0 \rightarrow \mathbb{G}_1$, where $\mathbb{G}_1$ is a group of the same order p. Pick $g_1 \in \mathbb{G}_0$, $y \in \mathbb{Z}_p^*$ at random and compute $g_2 = g^y$. Three cryptographically collision-resistant hash functions are selected: $H_0 : \{0,1\}^* \rightarrow \mathbb{G}_0$, $H_1 : \{0,1\}^* \rightarrow \mathbb{Z}_p^*$ and $H_2 : \mathbb{G}_1 \rightarrow \{0,1\}^{k_{Enc}}$ where k_{Enc} is the length of symmetric key in the secure private key encryption construction chosen by the patient. Then, define the attributes in universe U as elements in $\mathbb{Z}_p$. If $q_x(\cdot)$ is a polynomial w.r.t. leaf nodes, a default attribute set from $\mathbb{Z}_p$ with the size of $d_x - 1$ is given as $\psi_x = \{\psi_1, \psi_2, \cdots, \psi_{d_x-1}\}$ in the access tree. ω_D represents the set of attributes possessed by the physician.

Key Extract: Assume that the healthcare provider holds a uniform ephemeral private key $sk_D^{ephemeral} = hc$ shared by each physician working in it and the corresponding public key is $pk_D^{ephemaral} = e(g_1, g_2)^{hc}$. Let the attribute private key of the physician be

$$sk_D = (\gamma_i, \delta_i) = ((g_1 H_0(i))^{q_x(i)}, g^{q_x(i)})_{i \in \omega_D \cup \psi_x}, \tag{4}$$

and the public parameters be

$$(p, g, e, \mathbb{G}_0, \mathbb{G}_1, H_0, H_1, H_2, g_1, g_2) \tag{5}$$

ω_x^* is a set of required attributes the patient chooses for the predicate which his expected physicians must satisfy.

Sign: The signing algorithm outputs a signature of message m which can only be verified by the directly authorized physicians whose set of attributes satisfies the access tree $\mathscr{T}$. The patient firstly chooses a polynomial $q_x(\cdot)$ for each node x including the leaf nodes in the access tree $\mathscr{T}$. These polynomials are chosen in the following way of a top-down manner, starting from the root node R. For each node x in the tree, let d_x be the threshold value of node x and set the degree of the polynomial $q_x(\cdot)$ to be $D_x = d_x - 1$.

Starting with the root node R, the algorithm chooses a random $y \in \mathbb{Z}_p$ and sets $q_R(0) = y$. Then, it chooses d_R other points in the polynomial q_R randomly to define it completely. For any other node x, it sets $q_x(0) = q_{parent(x)}(index(x))$ and chooses d_x other points randomly to completely define $q_x(\cdot)$. Additionally, the threshold in each node polynomial $q_x(\cdot)$ can be flexibly adjusted to satisfy the different requirements of the patient, supporting all predicates $\mathscr{T}$. Specifically in detail, the node predicate $\mathscr{J}_{k_x, \omega_x^*}(\cdot) \rightarrow 0/1$ towards each node polynomial $q_x(\cdot)$ with threshold k_x from 1 to $d_x - 1$ is supposed as follows

$$\mathscr{J}_{k_x, \omega_x^*}(\omega_x) = \begin{cases} 1, |\omega_x \cap \omega_x^*| \geq k_x, \\ 0, otherwise. \end{cases} \tag{6}$$

In our scheme, the access structure can be adjusted flexibly without from scratch to save a lot of resources w.r.t the date sink (i.e. the hand-held mobile device) deployed on the patient. For example, if the predicate threshold required by a polynomial $q_x(\cdot)$ w.r.t node x in the access tree $\mathscr{T}$ is reduced from k_x to $k_x^{'}$, it means that there are $k_x - k_x^{'}$ child nodes of

x whose thresholds and associated attributes will be reduced to zero and included in $\psi_x^{'}$ to be satisfied by the physician's attributes by default and vice versa.

To sign a message m with the verification predicate $\mathscr{T}$, the patient chooses an ephemeral private key $sk_P^{ephemeral} = b \in \mathbb{Z}_p^*$ at random and computes the corresponding ephemeral public values $pk_P^{ephemeral} = g^b$. For the leaf node x in the access tree $\mathscr{T}$, let the current threshold required for the physician be k_x. For the leaf node polynomial $q_x(\cdot)$, the patient randomly selects a default subset $\psi_x^{'} \subseteq \psi_x$ with $|\psi_x^{'}| = d_x - k_x$ and calculates $pk_{P_i}^{ephemeral} = H_0(i)^b$ for $i \in \omega_x^* \cup \psi_x^{'}$. Then, he can derive the corresponding keys for authentication

$$K_{Encp} = e(g_1, g_2)^{sk_P^{ephemeral}} = e(g_1, g_2)^b, \tag{7}$$

$$K_{Enc} = H_2(K_{Encp}), \tag{8}$$

$$K_{Sig} = K_{Encp} pk_D. \tag{9}$$

Finally, the patient randomly selects $r_i \in \mathbb{Z}_p^*$ for each $i \in \omega_x^* \cup \psi_x^{'}$, makes g^{r_i} $(i \in \omega_x^* \cup \psi_x^{'})$ public and computes the signature as follows

$$\sigma^{'} = H_1(m \parallel K_{Sig}), \tag{10}$$

$$C = E_{K_{Enc}}(m), \tag{11}$$

$$\sigma_i^{''} = \{H_0(i)^{r_i}\}_{i \in \omega_x^* \cup \psi_x^{'}}, \tag{12}$$

where $E_{K_{Enc}}(\cdot)$ is a secure private key encryption construction chosen by the patient. After that, he can output the signature $\sigma = (\omega_x^*, C, \sigma^{'}, \sigma_i^{''})$.

Verify: If the set of attributes possessed by the physician satisfies the access tree $\mathscr{T}$, he can operate the verification as a recursive algorithm.

For the leaf node x, to verify the signature with the node predicate $\mathscr{J}_{k_x, \omega_x^*}(\cdot)$, namely to prove owning at least k_x attributes among an attribute set ω_x^* with the size of n_x, the physician firstly selects a k_x element subset $\omega_J \subseteq \omega_D \cap \omega_x^*$, chooses $r_i^{'} \in_R \mathbb{Z}_p^*$ for each $i \in \omega_x^* \cup \psi_x^{'}$ and computes

$$V^{'} = \prod_{i \in \omega_J \cup \psi_x^{'}} \gamma_i^{\Delta_{i, \omega_J \cup \psi_x^{'}}(0)}, \tag{13}$$

$$V^{''} = \prod_{i \in \omega_x^* \cup \psi_x^{'}} (\sigma_i^{''})^{r_i^{'}}, \tag{14}$$

$$V^{'''} = \prod_{i \in \omega_J \cup \psi_x^{'}} e(pk_{P_i}^{ephemeral}, \delta_i^{\Delta_{i, \omega_J \cup \psi_x^{'}}(0)} g^{r_i r_i^{'}}), \tag{15}$$

$$V^{''''} = \prod_{i \in \omega_x^* \setminus \omega_J} e(pk_{P_i}^{ephemeral}, g^{r_i r_i^{'}}), \tag{16}$$

$$K_{Decp}^x = \frac{e(V^{'} V^{''}, pk_P^{ephemeral})}{V^{'''} V^{''''}}$$
$$= \frac{e(g_1^{q_x(0)} \prod_{i \in \omega_J \cup \psi_x^{'}} H_0(i)^{q_x(i)\Delta_{i, \omega_J \cup \psi_x^{'}}(0) + r_i r_i^{'}}, g^b)}{\prod_{i \in \omega_J \cup \psi_x^{'}} e(H_0(i)^b, g^{q_x(i)\Delta_{i, \omega_J \cup \psi_x^{'}}(0) + r_i r_i^{'}})} \cdot$$

$$\frac{e(\prod_{\omega_x^* \setminus \omega_J} H_0(i)^{r_i r_i^{'}}, g^b)}{\prod_{i \in \omega_x^* \setminus \omega_J} e(H_0(i)^b, g^{r_i r_i^{'}})}$$
$$= e(g_1^{q_x(0)}, g^b)$$
$$= e(g_1, g_2)^b.$$

We now consider the recursive case when x is a non-leaf node. The verification algorithm will proceeds as follows. For all nodes z that are children of x, it calls the same verification algorithm with respect to itself and stores the corresponding partial output as F_z. Let $\mathbb{S}_x$ be an arbitrary k_x-sized set of child nodes z such that $F_z \neq \perp$. If no such set exists, the node will not be satisfied and the function will return $\perp$. Then, the physicians can compute

$$\begin{aligned} K_{Decp}^x &= e(F_x, g^b) = e(\textstyle\prod_{z \in \mathbb{S}_x} F_z^{\Delta_{i, \mathbb{S}_x^{'}}(0)}, g^b) \\ &\quad (i = index(x) \ and \ \mathbb{S}_x^{'} = \{index(z) : z \in \mathbb{S}_x\}) \\ &= e(\textstyle\prod_{z \in \mathbb{S}_x} g_1^{q_z(0)\Delta_{i, \mathbb{S}_x^{'}}(0)}, g^b) \\ &= e(\textstyle\prod_{z \in \mathbb{S}_x} g_1^{q_{parent(z)}(index(z))\Delta_{i, \mathbb{S}_x^{'}}(0)}, g^b) \\ &= e(\textstyle\prod_{z \in \mathbb{S}_x} g_1^{q_x(i)\Delta_{i, \mathbb{S}_x^{'}}(0)}, g^b) \\ &= e(g_1^{q_x(0)}, g^b). \end{aligned}$$

Now, we have defined the verification function for each node in the access tree $\mathscr{T}$. By the recursive algorithm defined above, the physicians can complete verification by simply calling the function on the root node R of the access tree $\mathscr{T}$.

$$K_{Decp} = e(F_R, g^b) = e(g_1^{q_R(0)}, g^b) = e(g_1, g_2)^b, \tag{17}$$

$$K_{Dec} = H_2(K_{Decp}), \tag{18}$$

$$H_1(D_{K_{Dec}}(C) \parallel K_{Decp} e(g_1, g_2)^{hc}) = H_1(m \parallel K_{Sig}) = \sigma^{'}, \tag{19}$$

where $D_{K_{Dec}}(\cdot)$ is the decryption algorithm for the private key encryption. If equation (19) holds, the physician outputs $Ture$; otherwise, outputs $\perp$.

Transcript Simulation: The directly authorized physicians can produce the signature σ_T intended for themselves by performing the following. Firstly, they can use K_{Dec}, K_{Decp} obtained in verification to encrypt a specific message m to C and compute $\sigma_T^{'} = H_1(m \parallel K_{Decp} e(g_1, g_2)^{hc}) = H_1(m \parallel K_{Sig})$. Then, he can generate $\sigma_{T_i}^{''}$ just as what the patient did in the sign algorithm and complete $\sigma_T = (\omega_x^*, C, \sigma_T^{'}, \sigma_{T_i}^{''})$. Note that the signature is indistinguishable from the original signature created by the patient.

The correctness of our PSCPA can be deduced straightforwardly from the construction described above.

B. Extension to Emergent Cases

In our scheme, the uniform ephemeral public key of the healthcare provider $pk_D^{ephemeral} = e(g_1, g_2)^{hc}$ can be obtained directly by each patient for generating the signing key. As a meaningful extension and an equivalence to PSCPA, it is assumed that a set of attributes is also assigned to each patient. Only the registered patients whose attributes satisfy the required access tree $\mathscr{T}_P$ assigned by the healthcare provider can recover $pk_D^{ephemaral}$ and K_{Sig} for effective signature. This extension can be utilized to prevent the adversaries from launching Denial of Service (DoS) attacks. Upon receiving the personal health information, the pre-checking department of the healthcare provider verifies the signature first and only the personal health information successfully verified will be transferred to the physicians and continued with following treatment. Therefore, by this extension it is likely to save a lot of network and medical resources such as communication bandwidth and physicians, making healthcare services much more efficiently. In emergent cases, the patient can directly submit her/his ephemeral private key b which can be updated afterwards to the rescue members or delegate it to close friends and relatives for signing the emergent health data.

VII. Analysis

A. Security Proof

Theorem 1 (Unforgeability) Let $\mathscr{A}$ be an adversary with existential forgeability under chosen message attack against our PSCPA scheme with a success probability defined as $Succ^{EFCMA}_{PSCPA,\mathscr{A}(t,q_{H_0},q_{H_1},q_k,q_s,q_v)}$. In time t, he can make at most q_{H_0}, q_{H_1} queries to the random oracle H_0, $H_1 : \{0,1\}^* \to \mathbb{Z}_p^*$ ($p \geq 2^l$, l is the system's security parameter), q_k queries to the key extraction oracle, q_s queries to the signing oracle and q_v queries to the verification oracle. Then, provided that $E_{K_{Enc}}(\cdot)$ is a secure private key encryption construction, there exists a simulator $\mathscr{B}$ who can use $\mathscr{A}$ to solve an instance of the GBDH Problem with the probability:

$$Succ^{GBDH}_{\mathscr{B}} \geq \prod\nolimits_{x\in|\mathbb{X}_{unsat}|} \frac{1}{C(d_x-1,d_x-k_x)} Succ^{EFCMA}_{Sig,\mathscr{A}} - \frac{q_v}{2^l-q_{H_1}-q_s}.$$

Proof. Provided that $E_{K_{Enc}}(\cdot)$ is a secure private key encryption construction, it is necessary for us to prove the remaining part of the construction secure under the authorized accessible privacy model (AAPM) described in section IV. The construction can be proven secure in the selective predicate model. Given a random instance $\{g, g^a, g^b, g^c\}$ of the Gap Bilinear Diffie-Hellman problem, we will show how the simulator $\mathscr{B}$ can use $\mathscr{A}$ to obtain the value $e(g,g)^{abc}$ with the help of DBDH oracle. Let the default attribute set be $\psi_x = \{\psi_1, \psi_2, \cdots, \psi_{d_x-1}\}$ for the predefined integer d_x. Firstly, the adversary $\mathscr{A}$ outputs the challenge predicate $\mathscr{T}^*$ including a node challenge predicate, namely, a threshold function k_x ($k_x \leq d_x$) out of n_x element attribute set ω_x^* for each polynomial $q_x(\cdot)$. Then $\mathscr{B}$ selects randomly a subset $\psi_x^* \subseteq \psi_x$ with $|\psi_x^*| = d_x - k_x$. In the proof we regard the hash functions as the random oracles $\mathscr{H}_0$ and $\mathscr{H}_1$. $\mathscr{B}$ simulates all the oracles to answer $\mathscr{A}$'s queries and maintains $\mathscr{H}_0$-list and $\mathscr{H}_1$-list to record all the hash queries and the corresponding responses. $\mathscr{H}_0$-list consists of the items (s, l, h), where s is the input of the hash and h is the output of the hash. $\mathscr{H}_1$-list consists of the items $(m, r, \sigma^{'}, tag)$, where (m, r) is the input of the hash and $\sigma^{'}$ is the output of the hash function. $tag = 1$ if $t = \frac{r}{pk_D^{ephemeral}} = e(g,g)^{abc}$, otherwise $tag = 0$, which is determined by DBDH oracle. We assume that $\mathscr{A}$ is well-behaved in the sense that $\mathscr{A}$ will never repeat the same queries in our simulation. $\mathscr{B}$ simulates the setup algorithm and sets $g_1 = g^a, g_2 = g^c, pk_P^{ephemeral} = g^b$.

$\mathscr{H}_0$ **Queries**. Upon receiving a query s_i ($i \in [1, q_{H_0}]$) to $\mathscr{H}_0$, $\mathscr{B}$ simulates $\mathscr{H}_0$ as follows.

(1) If there exists (s_i, l_i, h_i) in $\mathscr{H}_0$-list, return h_i to the adversary $\mathscr{A}$.

(2) Otherwise, if $s_i \in \omega_x^* \cup \psi_x^*$, $\mathscr{B}$ chooses $l_i \in \mathbb{Z}_p^*$ at random and computes $h_i = g^{l_i}$. Else, $\mathscr{B}$ chooses $l_i \in \mathbb{Z}_p^*$ at random and computes $h_i = g^{l_i}/g_1$. Then, $\mathscr{B}$ adds (s_i, l_i, h_i) into $\mathscr{H}_0$-list and returns h_i as the answer.

$\mathscr{H}_1$ **Queries**. $\mathscr{B}$ simulates $\mathscr{H}_1$ as follows. For any query $(m_i, r_i)(i \in [1, q_{H_1}])$ to $\mathscr{H}_1$, $\mathscr{B}$ submits (g^a, g^b, g^c, t_i) to the DBDH oracle and it will tell $\mathscr{B}$ whether $t_i = e(g,g)^{abc}$. Then, there are following two cases for $\mathscr{B}$ to simulate $\mathscr{H}_1$ oracle.

(1) If $t_i = e(g,g)^{abc}$, $\mathscr{B}$ checks $\mathscr{H}_1$-list

(1.1) If there exists an item $(m_i, \perp, \sigma_i^{'}, 1)$ in $\mathscr{H}_1$-list, $\mathscr{B}$ returns $\sigma_i^{'}$ as the answer. The items of this form in $\mathscr{H}_1$-list can be added during the signing queries.

(1.2) Otherwise, $\mathscr{B}$ chooses $\sigma_i^{'} \in_R \mathbb{Z}_p^*$ such that there is no item $(\cdot, \cdot, \sigma_i^{'}, \cdot)$ in $\mathscr{H}_1$-list. $\mathscr{B}$ then adds $(m_i, t_i, \sigma_i^{'}, 1)$ into $\mathscr{H}_1$-list and returns $\sigma_i^{'}$ as the answer.

(2) Else if $t_i \neq e(g,g)^{abc}$, $\mathscr{B}$ chooses $\sigma_i^{'} \in_R \mathbb{Z}_p^*$ such that there is no item $(\cdot, \cdot, \sigma_i^{'}, \cdot)$ in $\mathscr{H}_1$-list. $\mathscr{B}$ then adds $(m_i, t_i, \sigma_i^{'}, 0)$ into $\mathscr{H}_1$-list and returns $\sigma_i^{'}$ as the answer.

Key Extraction Queries. Suppose $\mathscr{A}$ adaptively makes a request for the private key towards the challenge predicate $\mathscr{T}^*$ where $\mathscr{T}^*(\omega) = 0$. To simulate the private key, $\mathscr{B}$ needs to assign a polynomial $q_x(\cdot)$ of degree d_x for each node x in the access tree $\mathscr{T}^*$. Assume that the adversary $\mathscr{A}$ makes at most q_k private key extraction queries and the requesting set of attributes ω satisfies $|\omega \cap \omega_x^*| < k_x$ for some polynomial $q_x(\cdot)$. Simulator $\mathscr{B}$ firstly defines three sets Γ, $\Gamma^{'}$, $\mathbb{S}$ in the following manner: $\Gamma = (\omega \cap \omega_x^*) \cup \psi_x^*$ and $\Gamma^{'}$ such that $\Gamma \subseteq \Gamma^{'} \subseteq \mathbb{S}$ and $\left|\Gamma^{'}\right| = d_x - 1$. Let $\mathbb{S} = \Gamma^{'} \cup \{0\}$.

For every $s_i \in \Gamma^{'}$, simulator $\mathscr{B}$ runs $\mathscr{H}_0$ oracle to get (s_i, l_i, h_i) in the $\mathscr{H}_0$-list, picks $\lambda_i \in \mathbb{Z}_p^*$ at random, computes $sk_{D_i} = ((g_1 h_i)^{\lambda_i}, g^{\lambda_i})$ and let $\lambda_i = q_x(s_i)$.

For the $s_i \in \mathbb{S}\backslash\Gamma^{'}$, the simulator $\mathscr{B}$ runs $\mathscr{H}_0$ oracle to get (s_i, l_i, h_i) in the $\mathscr{H}_0$-list, computes

$$\begin{aligned} sk_{D_i} = ((\textstyle\prod_{s_j\in\Gamma'} (g_1 h_i)^{\Delta_{s_j,\mathbb{S}(s_i)}\lambda_j}) g_2^{\Delta_{0,\mathbb{S}(s_i)} l_i}, \\ (\prod_{s_j\in\Gamma'} g^{\Delta_{s_j,\mathbb{S}(s_i)}\lambda_j}) g_2^{\Delta_{0,\mathbb{S}(s_i)}}) \end{aligned} \tag{20}$$

and returns $\{sk_{D_i}\}_{i\in\omega}$ to the adversary $\mathscr{A}$.

Now the simulator $\mathscr{B}$ defines $\lambda_i = q_x(s_i)$ for a random polynomial $q_x(\cdot)$ of degree $d_x - 1$ over $\mathbb{Z}_p^*$ such that $q_x(0) = c$. In this way, from the view of adversary $\mathscr{A}$, when $s_i \in \Gamma^{'}$ the simulated sk_{D_i} and those sk_{D_i} in the real attack are identically distributed. Even when $s_i \notin \Gamma^{'}$, the above simulation is also correctly distributed. Since $s_i \notin \Gamma^{'}$ means

$s_i \notin \Gamma$, we have $g_1 h_i = g^{l_i}$. Noting $g_2 = g^c$, we have

$$\begin{aligned} sk_{D_i} &= ((g^{l_i(\sum_{s_j \in \Gamma'} \Delta_{s_j,\mathbb{S}}(s_i) q_x(s_j))}) g^{\Delta_{0,\mathbb{S}}(s_i) l_i c}, \\ &\qquad (g^{\sum_{s_j \in \Gamma'} \Delta_{s_j,\mathbb{S}}(s_i) q_x(s_j)}) g^{\Delta_{0,\mathbb{S}}(s_i) c}) \\ &= (g^{l_i(\sum_{s_j \in \Gamma'} \Delta_{s_j,\mathbb{S}}(s_i) q_x(s_j) + \Delta_{0,\mathbb{S}}(s_i) q_x(0))}, \\ &\qquad g^{\sum_{s_j \in \Gamma'} \Delta_{s_j,\mathbb{S}}(s_i) q_x(s_j) + \Delta_{0,\mathbb{S}}(s_i) q_x(0)}) \\ &= (g^{l_i q_x(s_i)}, g^{q_x(s_i)}) \\ &= ((g_1 h_i)^{q_x(s_i)}, g^{q_x(s_i)}) \\ &= ((g_1 H_0(s_i))^{q_x(s_i)}, g^{q_x(s_i)}). \end{aligned}$$

Therefore, all the private keys $\{sk_{D_i}\}_{i \in \omega \cup \psi_x^*}$ simulated by $\mathscr{B}$ are distributed identically to the ones in the real attack. Finally, the simulator $\mathscr{B}$ can construct the private keys for the access tree $\mathscr{T}^*$ and the distribution of the private keys for $\mathscr{T}^*$ is identical to that in the original scheme.

Signing Queries. $\mathscr{B}$ simulates the signing oracle as follows. After receiving $\mathscr{A}$'s choice of the message m_i, $\mathscr{B}$ checks the $\mathscr{H}_1$-list.

(1) If there is an item $(m_i, t_i, \sigma_i^{'}, 1)$ in $\mathscr{H}_1$-list where $t_i = e(g,g)^{abc}$, $\mathscr{B}$ outputs $(\omega_x^*, \sigma_i^{'}, \sigma_{i_j}^{''})$ as the signature, where $\sigma_{i_j}^{''}$ can be generated according to the answers from running $\mathscr{H}_0$ oracle for queries $s_j \in \omega_x^* \cup \psi_x^*$.

(2) Else, $\mathscr{B}$ chooses $\sigma_i^{'} \in_R \mathbb{Z}_p^*$ such that there is no item $(\cdot, \cdot, \sigma_i^{'}, \cdot)$ in $\mathscr{H}_1$-list. $\mathscr{B}$ then adds $(m_i, \perp, \sigma_i^{'}, 1)$ into $\mathscr{H}_1$-list and returns $(\omega_x^*, \sigma_i^{'}, \sigma_{i_j}^{''})$ as the signature, where $\sigma_{i_j}^{''}$ can be generated the same as what is operated in the first case.

Verifying Queries. After receiving $\mathscr{A}$'s request (m_i, σ_i), $\mathscr{B}$ simulates the verifying oracle as follows.

(1) If there is no item $(\cdot, \cdot, \sigma_i^{'}, \cdot)$ in $\mathscr{H}_1$-list, $\mathscr{B}$ simulates the verification and rejects $(m_i, \sigma_i^{'})$ as an invalid signature.

(2) Else if there is an item $(\cdot, \cdot, \sigma_i^{'}, \cdot)$ in $\mathscr{H}_1$-list, and

(2.1) If this item has the form of $(m_i, \perp, \sigma_i^{'}, 1)$ or $(m_i, t_i, \sigma_i^{'}, 1)$, $\mathscr{B}$ will accept it as a valid signature.

(2.2) Otherwise, $\mathscr{B}$ will reject it as an invalid signature.

This makes a difference only if $(m_i, \sigma_i^{'})$ is a valid signature and $\sigma_i^{'}$ is not queried from $\mathscr{H}_1$. Since $\mathscr{H}_1$ is uniformly distributed, for all verifying queries this case happens with the probability less than $\frac{q_v}{2^l - q_{H_1} - q_s}$.

Now, if the adversary $\mathscr{A}$ outputs a valid signature (m^*, σ^*) such that $Verify(m^*, \sigma^*, sk_D, pk_P^{ephemeral}) = 1$, it means that there is an item $(\cdot, \cdot, \sigma^{'*}, \cdot)$ in $\mathscr{H}_1$-list. By the definition of the EFCMA adversary model, m^* cannot be queried in the signing oracle, therefore $\sigma^{'*}$ must be returned as the hash value of $\mathscr{A}$'s query (m^*, r^*). That is to say there is an item $(m^*, t^*, \sigma^{'*}, 1)$ in $\mathscr{H}_1$-list and $t^* = e(g,g)^{abc}$. Besides, for the success of $\mathscr{B}$, it is required for the correct guess of $d_x - k_x$ element subset ψ_x^* from a $d_x - 1$ element set ψ_x, the probability is $\frac{1}{C(d_x - 1, d_x - k_x)}$. Since the simulator $\mathscr{B}$ has the knowledge of the challenge predicate $\mathscr{T}^*$, he can compute the number of unsatisfied leaf node polynomials $q_x(\cdot)$ corresponding to the challenged set of attributes selected by the adversary $\mathscr{A}$, namely there are the corresponding number of polynomials whose default sets of attributes are required for the adversary's guessing. We denote this number as $|\mathbb{X}_{unsat}|$. Therefore, $\mathscr{B}$ successfully solves the GBDH problem with the probability:

$$Succ_{\mathscr{B}}^{GBDH} \geq \prod_{x \in |\mathbb{X}_{unsat}|} \frac{1}{C(d_x - 1, d_x - k_x)} Succ_{Sig,\mathscr{A}}^{EFCMA} - \frac{q_v}{2^l - q_{H_1} - q_s}.$$

Theorem 2 (Privacy of Patient's Identity) Our patient self-controllable cooperative authentication scheme (PSCPA) achieves signer-attribute privacy.

Proof. Due to the space restriction, we outline our proof sketch as follows. In our scheme, without loss of generality, for a (k_x, n_x) threshold attribute based verification with respect to one specific node x in the access tree $\mathscr{T}$, it is straightforward to see that the verifier cannot reveal which k_x attributes are really used in leaf node x for verification since any attribute subset of the size k_x can satisfy the predicate. In this way can the unconditional signer-attribute privacy of PSCPA be achieved.

We discuss some other security and privacy issues of our PSCPA constructions in distributed m-healthcare systems. It is noted that in PSCPA, the unique signing key K_{Sig} can be generated by both the patient and the directly authorized physicians. Therefore, when the medical consultation is needed, the signature transcription simulation of the patients' personal health information σ_T is transferred by the directly authorized physicians and shared among various other healthcare providers. Consequently, the indirectly authorized physicians in these institutions cannot distinguish it is signed by the patients or their directly authorized physicians but decipher the personal health information due to the unconditional signer-attribute privacy. The unauthorized persons can obtain neither. Only in this way can three levels of our privacy-preserving cooperative authentication be realized.

B. Performance Analysis

(1) Numerical analysis. We now consider the efficiency of PSCPA in terms of storage overhead, computational complexity and communication cost.

As to the storage overhead, the size of public parameters in our scheme is linear to the number of attributes in ω_x^* and $\psi_x^{'}$. The private key consists of two group elements in $\mathbb{G}_0$ for every leaf node in the key's corresponding access tree $\mathscr{T}$. That is the number of group elements in private keys equals to the number of attributes in the union of ω_D and a default set of attributes ψ_x. Assuming ω_x^* is one of the public parameters, the signature almost consists of one group element in $\mathbb{G}_0$ corresponding to each attribute in ω_x^* and $\psi_x^{'}$. Therefore, the communication cost is independent of the number of attributes in ω_D possessed by each physician. As to the computational overhead, compared to the hash functions (eg. SHA-1) and private key encryption (eg. AES), the most resource-consuming operations in PSCPA are parings and exponentiations which we will focus on for evaluating the computational complexity. In the signing procedure, the number of modular exponentiations is almost linear to the number of attributes in the union of the requiring attribute set ω_x^* and a default subset of attributes $\psi_x^{'}$. The verification

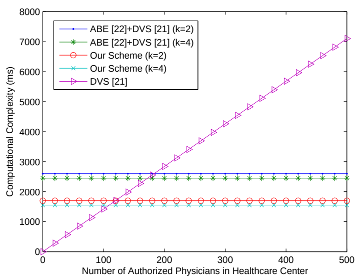

Fig. 4. Comparison of Computational Overhead among DVS, ABE+DVS and PSCPA towards N

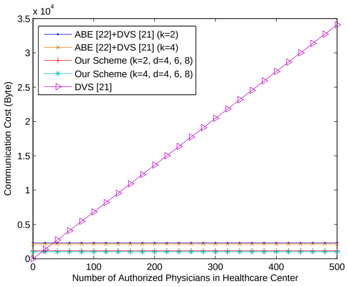

Fig. 5. Comparison of Communication Overhead among DVS, ABE+DVS and PSCPA towards N

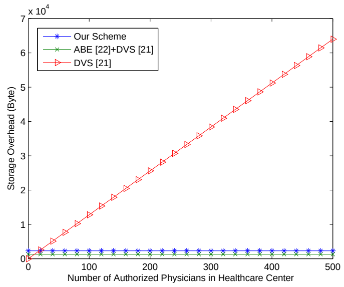

Fig. 6. Comparison of Storage Overhead among DVS, ABE+DVS and PSCPA towards N

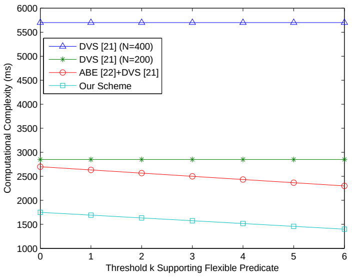

Fig. 7. Comparison of Computational Overhead among DVS, ABE+DVS and PSCPA towards k

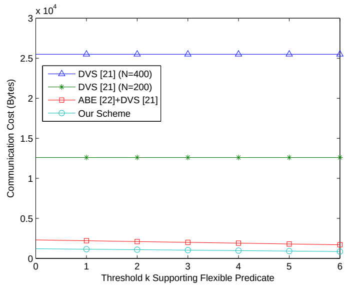

Fig. 8. Comparison of Communication Overhead among DVS, ABE+DVS and PSCPA towards k

procedure is by far the hardest to define performance for. In our verification algorithm described in Section V, the number of parings and exponentiations might always be linear to the number of nodes in the access tree. However, it can be reduced to $O(|\mathbb{S}_R|\,(n+d-k))$ [22] where $\mathbb{S}_R$ denotes the first k_R sets of the smallest size corresponding to the leaf nodes. To achieve the same security, our construction performs more efficiently than the traditional designated verifier signature (DVS) [21] for all the directly authorized physicians, where the overheads are linear to the number of directly authorized physicians. On the other hand, our construction also essentially distinguishes from the combination of a fine-grained attribute based encryption [22] and a traditional DVS [21] supporting flexible predicates, since in our construction the partial verifying key $e(g_1, g_2)^b$ is utilized for the secret key for encrypting m. It prevents the patient and the physicians from negotiating another symmetric encryption key in advance and saves almost half of the computational complexity, the signature size as well as the communication cost. Assume that n, n_D, d, k represent the size of the required set of attributes ω_x^*, the physician's attribute set ω_D, the default attribute set ψ_x and the flexible threshold respectively. P and E represent pairing and modular exponentiation operations. The storage and computational overhead of our construction PSCPA are illustrated in Table II and III respectively.

TABLE II
STORAGE OVERHEAD OF PSCPA

Items	Storage Overhead
Public Key	$O(n+d-k)$
Private Key	$O(n_D+d)$
Signature	$O(n+d-k)$

TABLE III
COMPUTATIONAL OVERHEAD OF PSCPA

Items	Computational Overhead
Sign	$O(n+d-k)E$
Verify	$O(|\mathbb{S}_r|\,(n+d-k))(P+E)$

(2) Implementation. In our implementation, we choose MIRACLE Library for simulating cryptograpersonal health informationc operations using Microsoft C/C++ compilers. For choosing an appropriate elliptic curve for simulation, let h be the group size of the elliptic curve and k be its embedding degree. To achieve a comparable security of 1024-bit RSA, it is necessary for us to make hk more or less equal to 1024. According to the standards of Paring-based Crypto Library [24], elliptic curves with $h = 512$ and $k = 2$ results in the fastest bilinear pairing in contrast to those with $k > 2$ for SS curves. Our test on Linux platform with an Intel Core2 Duo 2.53GHz CPU, it takes about 7ms and 27ms to perform a pairing and a scalar multiplication respectively. Consider a large quantity of pairing operations in our construction, it is reasonable to choose 512-bit SS curve $y^2 = x^3 + x$ for simulation. Assume that N represents the number of directly authorized physicians and we set $n = n_D = 10$, $d = 6$, $k \in \mathbb{Z} \wedge k \in [0, 6]$, $N \in \mathbb{Z} \wedge N \in [0, 500]$, the efficiency comparisons between the traditional DVS [21], the combination of ABE [22] and DVS [21] with flexible thresholds and our construction are evaluated. Fig. 4 shows that the computational complexity of PSCPA remains constant

regardless of the number of directly authorized physicians and nearly half of the combination construction of ABE [22] and DVS [21] supporting flexible predicate. Fig. 5 illustrates the communication cost of PSCPA also remains constant, almost half of the combination construction and independent of the number of attributes d in ω_D. Fig. 6 shows that though the storage overhead of PSCPA is slightly more than the combination construction, it is independent of the number of directly authorized physicians and performs significantly better than traditional DVS [21], all of whose computational, communication and storage overhead increase linearly to the number of directly authorized physicians. Fig. 7 and 8 show that the computational and communication overhead of the combination construction decrease slightly faster than PSCPA as the threshold k increases, however, even when k reaches the maximum value equaling to d, the overheads are still much more than PSCPA. The comparison between our scheme and the anonymous authentication based on PKI [6], [7] w.r.t the storage, communication and computational overhead towards N and k is identical to DVS [21], since to realize the same identity privacy, in all the constructions [6, 7, 21], a pair of public key and private key would be assigned to each directly authorized physician and the number of signature operations is also linear to the number of physicians, independent of the threshold k. The simulation results show our PSCPA better adapts to the distributed m-healthcare system than the previous schemes, especially for enhancing the energy-constrained mobile device, namely the data sink's efficiency.

VIII. Conclusions

In this paper, a novel authorized accessible privacy model (AAPM) and a patient self-controllable privacy-preserving cooperative authentication scheme (PSCPA) realizing three levels of security and privacy requirement in the distributed m-healthcare system are proposed, followed by the formal security proof and efficiency evaluations. Patients can authorize the physicians by setting an access tree supporting flexible threshold predicates. The directly authorized physicians, the indirectly authorized physicians and the unauthorized physicians would know both the patient's identity and the personal health information, only the personal health information and nothing respectively. Finally, simulation results show our PSCPA far outperforms previous schemes in terms of storage, computational and communication overhead.

Acknowledgment

This work was supported by the National Natural Science Foundation of China under grant 60773086, 60972034 and 61033014.

References

[1] L.Gatzoulis and I. Iakovidis, *Wearable and Portable E-health Systems*, IEEE Eng. Med. Biol. Mag., 26(5):51-56, 2007.

[2] I. Iakovidis, *Towards Personal Health Record: Current Situation, Obstacles and Trends in Inplementation of Electronic Healthcare Records in Europe*, International Journal of Medical Informatics, 52(1):105-115, 1998.

[3] E. Villalba, M.T. Arredondo, S. Guillen and E. Hoyo-Barbolla, *A New Solution for A Heart Failure Monitoring System based on Wearable and Information Technologies*, In International Workshop on Wearable and Implantable Body Sensor Networks 2006-BSN 2006, April, 2006.

[4] R. Lu and Z. Cao, *Efficient Remote User Authentication Scheme Using Smart Card*, Computer Networks, 49(4):535-540, 2005.

[5] M.D.N. Huda, N. Sonehara and S. Yamada, *A Privacy Management Architecture for Patient-controlled Personal Health Record System*, Journal of Engineering Science and Technology, 4(2):154-170, 2009.

[6] S. Schechter, T. Parnell and A. Hartemink, *Anonymous Authentication of Membership in Dynamic Groups*, in Proceedings of the Third International Conference on Financial Cryptography, 1999.

[7] D. Slamanig, C. Stingl, C. Menard, M. Heiligenbrunner and J. Thierry, *Anonymity and Application Privacy in Context of Mobile Computing in eHealth*, Mobile Response, LNCS 5424, pp. 148-157, 2009.

[8] M. Li, S. Yu, W. Lou and K. Ren, *Group Device Paring based Secure Sensor Association and Key Management for Body Area Networks*, In IEEE Infocom 2010.

[9] S. Yu, K. Ren and W. Lou, *FDAC: Toward Fine-grained Distributed Data Access Control in Wireless Sensor Networks*, In IEEE Infocom 2009.

[10] F.W. Dillema and S. Lupetti, *Rendezvous-based Access Control for Medical Records in the Pre-hospital Environment*, In HealthNet 2007.

[11] J. Sun, Y. Fang and X. Zhu, *Privacy and Emergency Response in E-healthcare Leveraging Wireless Body Sensor Networks*, IEEE Wireless Communications, pp. 66-73, February, 2010.

[12] X. Lin, R. Lu, X. Shen, Y. Nemoto and N. Kato, *SAGE: A Strong Privacy-preserving Scheme against Global Eavesdropping for E-health Systems*, IEEE Journal on Selected Areas in Communications, 27(4):365-378, May, 2009.

[13] J. Sun, X. Zhu, C. Zhang and Y. Fang, *HCPP: Cryptography Based Secure EHR System for Patient Privacy and Emergency Healthcare*, ICDCS'11.

[14] L. Lu, J. Han, Y. Liu, L. Hu, J. Huai, L.M. Ni and J. Ma, *Pseudo Trust: Zero-Knowledge Authentication in Anonymous P2Ps*, IEEE Transactions on Parallel and Distributed Systems, vol. 19, No. 10, October, 2008.

[15] J. Zhou and M. He, *An Improved Distributed key Management Scheme in Wireless Sensor Networks*, In 9th. International Workshop of Information Security Applications 2008-WISA 2008, September, 2008.

[16] M. Li, S. Yu, N. Cao and W. Lou, *Authorized Private Keyword Search over Encrypted Data in Cloud Computing*, ICDCS'11.

[17] M. Chase and S.S. Chow, *Improving Privacy and Security in Multi-authority Attribute-based Encryption*, In ACM CCS 2009, pp. 121-130, 2009.

[18] J. Bethencourt, A. Sahai, and B. Waters, *Ciphertext-Plicy Attribte-Based Encryption*, In IEEE Symposium on Security and Privacy, 2007.

[19] N. Cao, Z. Yang, C. Wang, K. Ren and W. Lou, *Privacy-preserving Query over Encrypted Graph-structured Data in Cloud Computing*, ICDCS'11.

[20] F. Cao and Z. Cao, *A Secure Identity-based Multi-proxy Signature Scheme*, Computers and Electrical Engineering, vol. 35, pp. 86-95, 2009.

[21] X. Huang, W. Susilo, Y. Mu and F. Zhang, *Short Designated Verifier Signature Scheme and Its Identity-based Variant*, International Journal of Network Security, 6(1):82-93, January, 2008.

[22] V. Goyal, O. Pandey, A. Sahai and B. Waters, *Attribute-based Encryption for Fine-grained Access Control of Encrypted Data*, In ACM CCS'06, 2006.

[23] J. Li, M.H. Au, W. Susilo, D. Xie and K. Ren, *Attribute-based Signature and its Applications*, In ASIACCS'10, 2010.

[24] PBC Library, *http://crypto.stanford.edu/pbc/times.html*.

[25] B. Riedl, V. Grascher and T. Neubauer, *A Secure E-health Architecture based on the Appliance of Pseudonymization*, Journal of Software, 3(2):23-32, February, 2008.

[26] D. Slamanig and C. Stingl, *Privacy Aspects of E-health*, In 3rd. International Conference on Availability, Reliability and Security, 2008.

[27] De-identified Health Information, *http://aspe.hhs.gov/admnsimp/bannerps.htm*.

[28] R. Lu, X. Lin, X. Liang and X. Shen, *A Secure Handshake Scheme with Symptoms-Matching for mHealthcare Social Network*, IEEE Journal on Selected Areas in Communications, Vol.27, No.4, pp.387-399, 2009.

[29] J. Sun and Y. Fang, *Cross-domain Data Sharing in Distributed Electronic Health Record System*, IEEE Transactions on Parallel and Distributed Systems, vol. 21, No. 6, 2010.