# ESPAC: Enabling Security and Patient-centric Access Control for eHealth in cloud computing

Mrinmoy Barua*, Xiaohui Liang,
Rongxing Lu and Xuemin Shen

Department of Electrical and Computer Engineering,
University of Waterloo,
Waterloo, Ontario, Canada N2L 3G1
E-mail: mbarua@ecemail.uwaterloo.ca
E-mail: x27liang@bbcr.uwaterloo.ca
E-mail: rxlu@bbcr.uwaterloo.ca
E-mail: xshen@bbcr.uwaterloo.ca
*Corresponding author

**Abstract:** We consider the problem of patient self-controlled access privilege to highly sensitive Personal Health Information (PHI), where PHI is expected to be securely stored in cloud storage for uninterrupted anytime, anywhere remote access. In order to assure the privacy of PHI, we propose Efficient and Secure Patient-centric Access Control (ESPAC) scheme which allows data requesters to have different access privileges based on their roles, and then assigns different attribute sets to them. Extensive security and performance analyses demonstrate that the ESPAC scheme is able to achieve desired security requirements with acceptable communication delay.

**Keywords:** eHealth; security; privacy; ABE; attribute-based encryption; access control; cloud computing.

**Reference** to this paper should be made as follows: Barua, M., Liang, X., Lu, R. and Shen, X. (2011) 'ESPAC: Enabling Security and Patient-centric Access Control for eHealth in cloud computing', *Int. J. Security and Networks*, Vol.

**Biographical notes:** Mrinmoy Barua is now pursuing the PhD Degree in Electrical and Computer Engineering at the University of Waterloo, Canada. His research interests include applied cryptography, wireless network security, and security and privacy in eHealth. He was the recipient of a Chancellor's Gold Medal in 2000, Ontario Graduate Scholarship (OGS) in 2007, R.S. McLaughlin Fellowship in 2008, Natural Sciences and Engineering Research Council of Canada Graduate Scholarships in 2009, and President's Graduate Scholarship at university of Waterloo in 2009.

Xiaohui Liang is currently working toward a PhD Degree with the Department of Electrical and Computer Engineering, University of Waterloo, Canada. He is currently a Research Assistant with the Broadband Communications Research (BBCR) Group, University of Waterloo. His research interests include network security and privacy, applied cryptography, and e-healthcare system.

Rongxing Lu is currently working toward a PhD Degree with the Department of Electrical and Computer Engineering, University of Waterloo, Canada. He is currently a research assistant with the Broadband Communications Research (BBCR) Group, University of Waterloo. His research interests include wireless network security, applied cryptography, and trusted computing.

Xuemin (Sherman) Shen received a BSc (1982) Degree from Dalian Maritime University, China, and MSc (1987) and PhD Degrees (1990) from Rutgers University, New Jersey, all in Electrical Engineering. He is a Professor and University Research Chair, Department of Electrical and Computer Engineering, University of Waterloo. His research focuses on mobility and resource management, UWB wireless networks, wireless network security, and vehicular ad hoc and sensor networks. He served as an Area Editor for IEEE Transactions on Wireless Communications and Editor-in-Chief for Peer-to-Peer Networks and Applications. He is a Fellow of Engineering Institute of Canada, a registered Professional Engineer of Ontario, Canada, and a Distinguished Lecturer of the IEEE Communications Society.

This paper is a revised and expanded version of a paper entitled 'PEACE: An Efficient and Secure Patient-centric Access Control Scheme for eHealth Care System' presented at *Proc. IEEE INFOCOM'11-SCNC*, 10–15 April, 2011, Shanghai, China.

## 1 Introduction

Electronic health (eHealth) care is a promising paradigm that has drawn extensive attention from both academia and industry recently. It describes the application of information and communication technologies across the whole range of function that affect the patient's PHI. The eHealth care system shows a high potential to improve the quality of diagnosis, reduce medical costs and help address the reliable and on-demand health care challenges posed by the aging society. Recent advances in Wireless Body Area Networks (WBANs) have made it possible to deploy bio-sensors on, in, or around the patient body and allow to continuous monitoring of physiological parameters (e.g., electrocardiogram (ECG), blood oxygen levels) with physical activities (Barua et al., 2011). It has lent great forces to the migration of health care system from hospital or care unit to the patient's residence. Integrating this technology with the existing wireless technologies permits real-time mobile and permanent monitoring of patients, even during their daily normal activities. In such a heterogeneous wireless environment, secure communication of the patient PHI with integrity and confidentiality guarantees is an essential part of a reliable eHealth care system.

In addition, the eHealth care system needs to ensure the availability of PHI in electronic form adheres to the same levels of privacy and disclosure policy as applicable to present-day paper-based patient-records accessible only from the physician's office. Instead of storing the PHI locally, the recent advancement of cloud computing allows us to store all PHI at the cloud storage and ensures availability with reduces the capital and operational expenditures (Kamara and Lauter, 2010). In cloud storage (or data storage as a service), data is stored on multiple third-party servers where the storage can be administrated on demand. Migrating patients PHI into this cloud storage offers enormous conveniences to the eHealth care providers, since they do not have to care about the complexities of direct hardware management (Yu et al., 2010). However, computerised PHI are open to potential abuse and security threats. Storing large amounts of patient's sensitive medical data in third-party cloud storage is vulnerable to loss, leakage, or theft (Johnson, 2009). Stored data confidentiality is considered as one of the biggest challenges raised by cloud storage environment. Especially in a public clouds environment, which are operated by commercial service providers and shared by various other customers, data confidentiality is a desired property.

Traditional data access schemes which are used to provide data confidentiality are mostly depend on the system itself to enforce authorisation policies and rely on the system trusted infrastructure. Providing data confidentiality by server side data encryption is not also appropriate for the health application when the server is not fully trusted. In addition, patient's privacy with proper access control to cloud storage is a growing concern in the eHealth care industry due to its direct involvement to human. Patient generally wants to be sure that his sensitive health information can only be accessed by particular authorised users and his original identity will not be exposed.

To store PHI in a cloud storage with patient-centric access control privilege, we use ciphertext-policy attribute-based encryption (Bethencourt et al., 2007) in ESCAP. Identity based encryption is adopted to ensure secure end-to-end communication among patient, eHealth care service provider, and cloud storage. Our contributions are in three-fold:

- provide an architectural model of eHealth care system
- show how ESPAC provides a secure communication between remote patient and eHealth care provider
- present an patient-centric access control policy that helps ESPAC to has more reliability.

To construct this access control policy, we assign different attribute sets to data requesters based on their relation to the patient. For example, general users may know some common attributes of a patient, e.g., location, gender; patient's relatives or health care givers may know more private information of a patient, likely medication details, patient date of birth, patient phone number, etc.; health insurance providers may have more privileges and can know patient health card number, Social Identification number, etc.

The remainder of this paper is organised as follows. Section 2 contains a brief description of related work. System model and security requirements are presented in Section 3. Preliminaries such as bilinear pairing, security definition are introduced in Section 4. The proposed ESPAC scheme is presented in Section 5. Section 6 and Section 7 provide security analysis and performance analysis of the proposed ESPAC scheme respectively. The paper is concluded in Section 8.

## 2 Related works

Existing research works related to proposed ESPAC includes

- secure and privacy preserving eHealth care system
- Attribute-Based Encryption
- access control over untrusted cloud storage.

Hybrid security policy for WBANs with Quality of Services (QoS) have recently been proposed for secure eHealth care system in Barua et al. (2011). Public key cryptography is used for session key management and private key cryptography is used for regular data encryption in WBANs environment. Due to the nature of the real-time traffic, emergency health application traffic is given high priority compare to other applications traffic. Lu et al. (2010) propose a mobile health care social network, where two patients can communicate each other if they have the same symptoms. Performance analyses demonstrate that emergency response time can be minimised by using proposed mobile health care social network. Lin et al. (2009) present a privacy preserving scheme for health care that can effectively works against global adversary. Both content and contextual privacy can be achieved by the proposed work.

Attribute-Based Encryption (ABE), a novel extension from identity based encryption by enabling expressive access policy to control the decryption process is first presented in Sahai and Waters (2005). Key Policy Attribute-based Encryption (KP-ABE) and Ciphertext Policy Attribute-based Encryption (CP-ABE) are the two main variants of ABE proposed so far. In both cases, user has a set of attributes that associate with user's private key. The attribute set is used to describe a user's credentials. In KP-ABE, user's private key is embedded with an access policy, whereas ciphertext is encrypted by an pre-defined access policy in CP-ABE (Goyal et al., 2006; Bethencourt et al., 2007). Liang et al. (2010) present a patient self-controllable access policy so that patients would have the primary control of the access to their own PHI.

Health records sharing and integrating in health care cloud is discussed in Zhang and Liu (2010). The paper describes the security reference model for managing different security issues in health care clouds. Yu et al. (2010) propose a fine gained data access control in cloud computing based on KP-ABE. Confidentiality of user access privilege and user secret key accountability can be achieved by the work. A mandatory access control model to protect patient's metadata with privacy is presented in Luna et al. (2010). It is shown that the use of fragmentation after encryption greatly improves overall security because potential attackers need to compromise more data file to gain access. Without disclosing the data contents, data owner delegates most of the computation tasks involved in fine-grained data access control to untrusted cloud server by combining techniques of ABE, proxy re-encryption, and lazy re-encryption. An efficient cloud storage sharing scheme is presented in Liu et al. (2010). The scheme works on hierarchical identity based encryption, where intended recipients can share the file by using their private keys. Wang et al. (2010) combine hierarchical identity-based encryption and CP-ABE to achieve fine-grained access control in cloud storage services.

## 3 System model and security requirements

In this section, we define the system model and then describe the security requirements of the proposed ESPAC scheme.

### *3.1 System model*

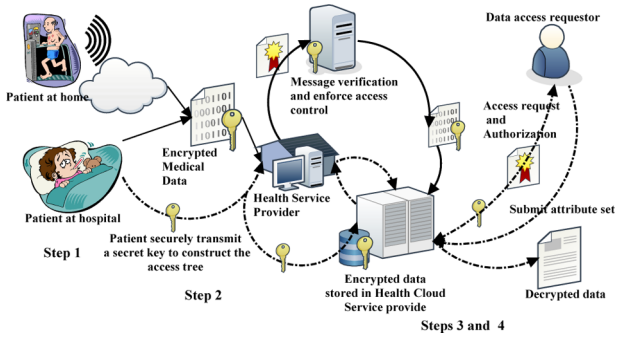
In our system model, we define the following entities:

- *Trusted Authority* (*TA*): It generates the public and secret key parameters for the ESPAC. The trusted authority is responsible for attributes' keys issuing, revoking, and updating. It grants differential access rights to individual users based on their attributes and roles. Trusted authority also maintains an index-table, where it stores the location of distributed data storage server. Authorised health service providers (e.g., Hospital, urgent care) are denoted as trusted parties.
- *Cloud service provider*: It provides data outsourcing services and consists of data servers and data service manager. The main responsibility of the data storage server is to serve and retrieve data according to authorised users' request. Data service manager negotiates with health care service provider to control the access from outside users to the stored encrypted data.
- *Registered user*: Patient who is registered to the trusted authority is considered as registered user. A registered user is responsible for defining attribute-based access policy and encrypting the sensitive PHI under the predefined policy before storing at the cloud-storage.
- *Data-access requester*: Cloud users who request to access some specific PHI are called the data-access requester. The ESPAC scheme ensures that any data-access requester can only decrypts the encrypted data if and only if he can successfully completes the access-policy.

The encrypted data is stored in a centralised storage, health-cloud, for future access. Based on the major operations, the proposed scheme can be classified into four major steps, as shown in Figure 1.

*Step 1* (*PHI collection*): In this initial step, using different body sensors, PHI is sensed and ready to be transmitted to the trusted eHealth care service provider.

**Figure 1** Major steps of the proposed ESPAC scheme (see online version for colours)




*Step 2* (*Secure data communication*): In this step, public key cryptography is used to securely transfers collected PHI to the eHealth care service provider. Patient securely transfer a secret key to the trusted eHealth care provider, if he authorised the service provider to build-up the access tree.

*Step 3* (*PHI processing at eHealth care provider*): After receiving the PHI securely, eHealth care service provider classifies the PHI based on the attributes set chosen by the patient. It then makes different privacy levels of data requesters based on their roles (e.g., level-1: general users, level-2: pharmacist, level-3: doctors, etc.) and assigns different set of attributes to these different levels.

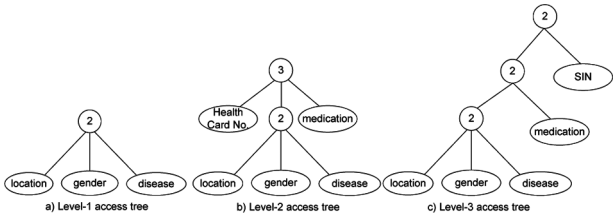
*Step 4* (*Transfer PHI to the cloud storage and control access*): After the data classification, encrypted data securely transfers to the cloud storage, shows as 'Health Cloud' in Figure 1. eHealth care service providers may operate either real-time or periodically based on the existing infrastructures. Data-access requester sends request to the cloud storage with a data block identity. They may also request for the corresponding attribute sets. In this case, the cloud storage provider communicates with the eHealth care service provider and verifies the authentication of the requesters. The data requester, as a node in the access tree ($\mathbb{T}$), can decrypt a ciphertext if and only if other corresponding nodes (users) also cooperate with him, or he has all the attribute sets to complete the $\mathbb{T}$.

In our system model, we classify the data requester as health worker, physicians, researchers, insurance companies, and agencies, etc. Some of them only need the accumulated number of patients in a specific area, some need disease related syndromes, age and gender specific characteristics, while others may need medication details. Figure 2 shows possible access structures based on different privacy levels, where intermediate nodes work as a logic gates. For example, "2 of (location, gender, disease)" in Figure 2(a) can be converted to "(location AND gender) OR (gender AND disease) OR (disease AND location)".

### 3.2 Security requirements

We aim at achieving the following security objectives.

**Figure 2** Access trees based on different data privacy level




1 *Patient-centric access control*: The system should provide patient-centric access control, where a patient can decide who can get the access to his or her stored PHI.

2 *Message integrity, source authentication and non-repudiation*: All accepted messages should be delivered unaltered, and the origin of the messages should be authenticated by the eHealth care service provider. To ensure the non-repudiation, the patient can not refute the validity of a PHI afterward.

3 *Prevention of Ciphertext-only attack*: The system should be secured enough to prevent recover of the plaintext from a set of stored ciphertexts.

4 *Provide patient privacy*: Privacy is one of the important concerns from a patient perspective. Illegal disclosure and improper use of patient PHI can cause legal disputes and undesirable damaging in patient's personal life.

5 *Resistant to collusion attack*: If multiple users collude, generally they may be able to decrypt a ciphertext by combining their attributes. Users can not get any access to the encrypted data even by sharing information in a group.

6 *Resistant to Denial-of-Service (DoS) attack:* The DoS attack may be caused due to the large groups of legitimate users access the eHealth care service provider at the same time, or the attacker continuously launch false traffic with a High Data Rate (HDR). The system should ensure acceptable QoS level to resist the DoS attack.

## 4 Preliminaries

Since the bilinear pairing and the attribute-based ciphertext policy work as the basis of our proposed scheme, we briefly review some related definitions and problem hardness, which closely follow those in Boneh and Franklin (2001).

*Basic of bilinear pairing*: Consider two groups $\mathbb{G}_1$ an additive, and $\mathbb{G}_2$ a multiplicative group of the same prime order $q$. Let $P$ and $Q$ be the two generators of $\mathbb{G}_1$, and $aP$ is the $a$ times addition of $P$. We can write the mapping $e$ as $e : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ which has the following properties:

1 *Bilinear*: $\forall P, Q \in \mathbb{G}_1, \forall a, b \in Z_q^*$
$e(aP, bQ) = e(P, Q)^{ab}$

2 *Non-degeneracy*: $P \neq 0 \Rightarrow e(P,P) \neq 1$

3 *Symmetric*: $\forall P,Q \in G_1, e(P,Q) = e(Q,P)$

4 *Computability*: $e$ is efficiently computable.

*Definition (BDH parameter generator)*: An algorithm $Gen$ is called a Bilinear Diffe-Hellman (BDH) parameter generator if $Gen$ takes a sufficient large security parameter $K > 0$ as input, runs in polynomial time in $K$, outputs a prime number $q$, the description of two groups $\mathbb{G}_1$ and $\mathbb{G}_2$ of order $q$, and the description of a bilinear map $e : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$.

*Definition (BDH Problem hardness)*: Given a random element $P \in \mathbb{G}_1$, as well as $aP$,$bP$,$cP$, for some random $a,b,c \in \mathbb{Z}_q^*$; there is no efficient algorithm to compute $e(P,P)^{abc} \in \mathbb{G}_2$ from $P, aP, bP, cP \in \mathbb{G}_1$. This implies the hardness of the BDH in the group $\mathbb{G}_1$ (Boneh and Franklin, 2001).

*Definition (Access structure (Bethencourt et al., 2007))*: Let $\{a_1, a_2, \ldots, a_n\}$ be a set of health attributes. The sets $\mathbb{A}$ ($\mathbb{A} \subset 2^{\{a_1,a_2,\ldots,a_n\}}$) are called the authorised attributes set, and the sets not in $\mathbb{A}$ are called the unauthorised sets. $\mathbb{A}$ is monotone if $\forall B,C : if B \subseteq \mathbb{A}$ and $B \subseteq C$ then $C \subseteq \mathbb{A}$.

In the access-tree construction, ciphertexts are labelled with a set of descriptive authorised attributes. Secret keys are identified by an access tree in which each interior node of the tree is a threshold gate and the leaves are associated with attributes.

*Setup* $(1^t)$: The Probabilistic Polynomial Time (PPT) setup algorithm takes as input a security parameter $1^t$. It outputs the public parameters $PK$ and a master key $MK$ which is known only to the private key generator.

$\mathbf{Encrypt_1(PKs, m, PKr)}$: The encryption algorithm takes the public parameters of the sender and receiver and encrypt the message 'm' by doing mapping and XOR operations. We use $Encrypt_2(PK, M, A)$ function to encrypt the message M and store in the health cloud. This encryption algorithm takes the system public parameters PK, a message M, and an access structure $\mathbb{A}$ over the universe of health attributes. The encrypted ciphertext CT can only be decrypted if and only if the user possesses the set of health attributes that satisfy the access tree structure.

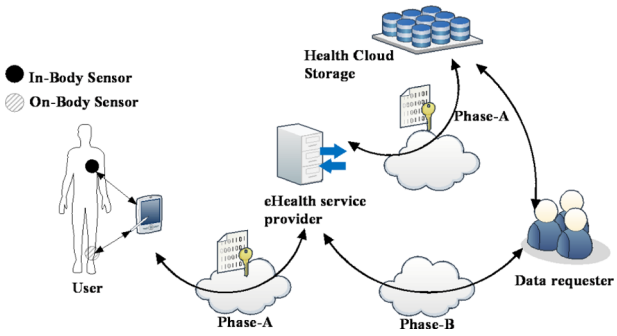
$\mathbf{Decrypt_1(PK, C, d)}$: The decryption algorithm takses as input the public parameter PK, ciphertext C, and the product of the receiver's secret key and sender PK's hash value. The health care provider uses this function to decrypt the encrypt message sent by the user for further processing. Another decryption function $Decrypt_2(PK, CT, SK)$ takes as input the public parameters PK, a ciphertext CT, which contains the access policy $\mathbb{A}$, and a secret key SK, which is a private key for a set S of health attributes. If the set S of attributes satisfies the access structure $\mathbb{A}$, the algorithm will decrypt the ciphertext and return the message M.

The set of algorithms must satisfy the standard consistency requirements: For $(PK, MK) \leftarrow Setup(1^t)$ (MK is the system Master Key), $(k, E) \leftarrow Encryption(PK, \gamma)$, $D_{\mathbb{A}} \leftarrow KeyGen(PM, MK, \mathbb{A})$ and $\mathbb{A}(\gamma) = 1$ (i.e., the attribute set $\gamma$ satisfies the access structure $\mathbb{A}$), then we have $Pr[Decryption(PK, E(M), D_{\mathbb{A}}) = k] = 1$.

## 5 Proposed ESPAC scheme

The four major categories that have described in the system model can be further integrated into two major phases, as shown in Figure 3.

**Figure 3** Two major phases of the proposed scheme (see online version for colours)




### 5.1 Phase-A: secure data communication

In the phase-A, the scheme defines the secure and privacy preserving communication between different eHealth users. Here, we describe the secure communication steps between a remote user and an eHealth service provider; communication among others e.g., eHelath service provider and the cloud storage or data requesters will follow the same steps.

*Step 1* (*System initialisation*): Given the security parameter $S'$, the bilinear parameters $(q, G_1, G_2, e, P)$ are generated by the function $setup(S')$. It is assumed that a unique $ID$ is given to the health care provider (hcp) by a trusted authority and the health service providers will do the following initialisations:

1 select a random number $\alpha \in_R Z_q^*$ and compute the public key $PK_{hcp} = \alpha.P$

2 generate the hash function $H_1 : \{0,1\} \rightarrow \mathbb{G}_1^*$ and compute the key $K_{hcp} = H_1(ID)$ for message encryption and decryption

3 generate the secure hash function $H_2 : \{0,1\}^* \rightarrow \{0,1\}^n$, $H_3 : \{0,1\}^* \rightarrow G_1^*$ and $H_4 : G_2 \rightarrow \{0,1\}^*$

4 compute the remote user's pseudo-identity $(U_{PID}) = H_2(U_{ID})$, and store a copy of it for future verification

5 securely distribute $U_{PID}$, $H_2$, $H_3$, and $H_4$ to its subscribers

6 for attribute-based Encryption and Decryption, the TA chooses two random number $\bar{\alpha}, \bar{\beta} \in \mathbb{Z}_p$, $p$ is the prime order with generator $g$

7 compute public key, $PK = \mathbb{G}, g, h = g^{\bar{\beta}}, f = g^{1/\bar{\beta}}, e(g,g)^{\bar{\alpha}}$ (details in Bethencourt et al. (2007))

8 compute master key, $MK = (\bar{\beta}, g^{\bar{\alpha}})$.

An individual user ($U$) will do the following steps:

1 user Chooses a random number $r \in_R Z_q^*$ and computes the public key $PK_U = r.P$

2 user selects a random number $\beta \in_R Z_q^*$, to calculate the session key $P_\beta = \beta.P$

3 user computes the message token $T = H_2(m|U_{PID}|session\_id)$ and sends it to the receiver along with encrypted data and session key.

*Step 2* (*Secure message communication*): After the system initialisation, both parties use the data encryption and decryption algorithms to securely transmit their data. Here, we show how an user will encrypt the message 'm' (equation (1)) and decrypt the encrypted message by the corresponding eHealth care service provider. The user encrypts the message, m, based on the public key of the corresponding receiver using the identity based encryption (Boneh and Franklin, 2001).

$$v = Encrypt_1(PK_{hcp}, m, PK_U) = m \oplus H_4(g_U^r). \quad (1)$$

Here, $Q_U = H_3(U_{PID})$; $H_3 : \{0,1\}^* \to G_1^*$, a random oracle; $g_u = e(Q_U, PK_{hcp})$, and $H_4 : G_2 \to \{0,1\}^*$, a random oracle.

The encrypted message is decrypted using the $Dec(PK_U, v, d)$ function, where $d = \alpha H_3(U_{PID})$ and $\alpha$ is the secret key of the corresponding agent.

$$\begin{aligned} Decrypt_1(PK_U, v, d) &= m \quad (2)\\ Decrypt_1(PK_U, v, d) &= v \oplus H_4(e(d, PK_U)\\ &= v \oplus H_4(e(\alpha H_3(U_{PID})), rP)\\ &= v \oplus H_4(e(H_3(U_{PID}), P)^{r\alpha})\\ &= v \oplus H_4(e(H_3(U_{PID}), \alpha P)^{r})\\ &= (m \oplus H_4(g_u^r) \oplus H_4(g_u^r)) = m. \end{aligned}$$

*Step 3* (*Message signature and verification*): To ensure data integrity, the receiver will verify the message signature after receiving it. By doing it, the eHealth service provider can verifies the data originated from the specific patient and can not be altered after signing it. We use the cryptographic digital signature (equation (3)), based on the bilinear pairing to provide data integrity. The patient first creates a session key $P_\beta = \beta P$, here $\beta \in_R Z_q^*$, and computes the message token $T$. He then computes the signature using the equation (3).

$$S = \frac{1}{v + \beta + r + T} P. \quad (3)$$

The eHealth service provider verifies the signature by using the equation (4).

$$e(vP + P_\beta + PK_{U_{PDA}} + TP, S) = e(P, P) \quad (4)$$

$$\begin{aligned} &e(vP + P_\beta + PK_{U_{PDA}} + TP, S)\\ &\quad = e((v + \beta + r + T)P, (v + \beta + r + T)^{-1}P)\\ &\quad = e(P, P)^{(v+\beta+r+T)(v+\beta+r+T)^{-1}} = e(P, P) \end{aligned}$$

### 5.2 Phase B: control of data requesters access

In a traditional public key cryptography system, the receiver and sender need each other public parameters to encrypt a message. But in the eHealth care system, the patient does not have any knowledge about the data requester or does not know who is going to access his PHI. Therefore, the security scheme by itself has to be capable to grant access control remotely. We use attribute-based ciphertext policy with privacy leveling to solve this challenge. Based on the different roles of the data requesters, an access tree is created and the requester needs to provide corresponding attributes (nodes of the tree) to have the secret key and thereafter he can use the secret key to decrypt the encrypted data (PHI). Providing falls attributes will stop the decryption processes immediately and the data requester learns nothing more than the attributes he or she is entitled. Details construction of the access tree with related key-generation, encryption, and decryption algorithms are described below.

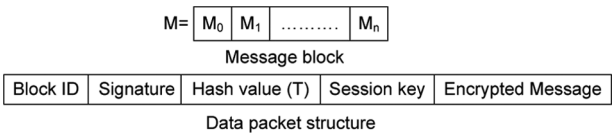
*Access tree* ($\mathbb{T}$): Let $\mathbb{T}$ represent an access structure. Each non-leaf node of the tree represents a threshold gate. If $num_x$ is the number of children of a node x and $k_x$ is the threshold value, then $0 \le k_x \le num_x$. When $k_x = 1$, the threshold gate is an OR gate, when $k_x = num_x$, it is an AND gate, finally when $1 \le k_x \le num_x$, it is a combination of AND and OR gates (Figure 2). The function $parent(x)$ returns the parent of node x. The function $att(x)$ is defined only if x is a leaf node and denotes the attribute associated with the leaf node x. The function $index(x)$ returns an ordering number associated with node x.

Let $\mathbb{T}$ be an access tree with root 'r'. Denote by $\mathbb{T}_x$ the subtree of $\mathbb{T}$ rooted at the node 'x'. Hence $\mathbb{T}$ is the same as $\mathbb{T}_r$. If a set of health attributes $\omega$ satisfies the access tree $\mathbb{T}_x$, we denote it as $\mathbb{T}_x(\omega) = 1$. We compute $\mathbb{T}_x(\omega)$ recursively as follows:
If 'x' is a non-leaf node, evaluate $\mathbb{T}_z(\omega)$ for all children $z$ of node 'x'. $\mathbb{T}_x(\omega)$ returns 1 if and only if at least $k_x$ children return 1. If 'x' is a leaf node, then $\mathbb{T}_x(\omega)$ returns 1 if and only if $att(x) \in \omega$.

*Dasta formation and authentication*: Before encrypting the data packets, the trusted eHealth care provider classifies the dataset based on some privacy levels and assign some attributes on that message block (M). It then concatenates the message block (M), user pseudo identity $U_{PID}$, and the $session\_id$. After that the trusted eHealth care provider computes the token

value $T = H_2(M|U_{PID}|session\_id)$. It then computes the signature using equation (3). Local health care provider will store the block sequence and patient pseudo identity for future verification. Figure 4 shows the data packet structure. The health cloud service provider will check the message authenticity by verify the signature using equation (4).

**Figure 4** Data packet architecture




The health cloud service provider will generate the signature in the same way and store along with the encrypted messages for the data requester verification purposes.

$Encrypt_2(PK,M,T)$: The algorithm first chooses a polynomial $q_x$ for each node x in the tree $T$. These polynomials are chosen in a top-down manner, starting from the root node. For each node x in the tree, set the degree $d_x$ of the polynomial $q_x$ to be one less than the threshold value of $k_x$. Starting with the root node 'R', the algorithm chooses a random $s \in \mathbb{Z}_p$ and sets $q_R(0) = s$. Then it chooses $d_R$ other points of 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 $q_x$.Finally, the ciphertext is then constructed by giving the access tree structure $\mathbb{T}$ and compute

$$CT = (\mathbb{T}, C^{'} = Me(g,g)^{\alpha s},$$
$$C = h^s, \forall y \in Y : C_y = g^{q_y(0)}, C_y^{'} = H(att(y))^{q_y(0)}). \quad (5)$$

*KeyGen*$(MK, S)$: The key generation algorithm takes as input a set of attributes $S$ and outputs a key that identifies with the set. The algorithm first chooses a random $r \in \mathbb{Z}_p$, and then random $r_j \in \mathbb{Z}_p$ for each attribute $j \in S$, and outputs the key as

$$SK =$$
$$(D = g^{(\alpha+r)}/\beta, \forall j \in S : D_j = g^r.H(j)^{r_j}, D_j^{'} = g^{r_j}). \quad (6)$$

$Decrypt_2(CT, SK)$: The decryption procedure works as recursively and is defined by the function $DecryptNode(CT, Sk, x)$ that takes as input a ciphertext $CT$ and a private key $SK$. If the node x is a leaf node, then the function works as follows:

$$\begin{aligned} DecryptNode(CT, SK, x) &= \frac{e(D_i, C_x)}{e(D_i^{'}, C_x^{'})} \\ &= \frac{e(g^r.H(i)^{r_i}, h^{q_x(0)})}{e(g^{r_i}, H(i)^{q_x(0)})} \\ &= e(g,g)^{rq_x(0)} \end{aligned}$$

Here $i \in S$. If $i \notin S$, we define $DecryptNode(CT, Sk, x) = \perp$. When x is a non-leaf node, the algorithm is called by its all child nodes z. It then stores the output of $DecryptNode(CT, Sk, z)$ as $F_z$. Detail is shown in Bethencourt et al. (2007). If the data requester can submit all the attributes correctly, the algorithm then executes on the root node 'R'. If the tree is satisfied by $S$, we set $A = DecryptNode(CT, SK, r) = e(g,g)^{rq_R(0)} = e(g,g)^{rs}$. The message 'M' can be decrypted by computing

$$C^{'}/(e(C,D)/A) = C^{'}/(e(h^s, g^{(\alpha+r)/\beta})/e(g,g)^{rs}) = M$$

## 6 Security analysis

In this section, we evaluate the security and privacy issues of the proposed scheme.

*The ESPAC scheme ensures user and eHealth agent's identity privacy*: User and health agent use pseudo identity instead of their unique identity, and these pseudo identities are generated by a strong one-way hash function. The construction of the hash function is easy to sample and compute but hard to invert. Therefore, the privacy is ensured by the proposed scheme.

*The scheme is secure to chosen ciphertext-only attack*: Data transmissions from user to health agent, as well as from health agent to health cloud service provider are done with proper encryption schemes ($Encryption_1$ and $Encryption_2$). The processes are indistinguishable under chosen ciphertext attack based on the BDH problem hardness and this hardness ensures there is no PPT algorithm that can decrypt the message from a set of chosen ciphertext.

*The scheme is resistant to the eavesdropping and collusion attacks*: An eavesdropping attacker aims at accessing the private and sensitive patient's medical data. This attack may be happened during the patient to eHealth care provider or eHealth care provider to the health cloud data communication. The BDH hardness ensures that the proposed scheme is resistant to this eavesdropping attack. To access the data at the health cloud server, an attacker needs to has sufficient attributes to complete the access tree. Here the random number '$s$' is divide into multiple shares based on the attributes set. For the non-privacy dataset, he may get access and its allowed in our scheme. But he cannot modified the data due to the verification bindings. However, for the patient sensitive data, a unique random number is embedded into both 'C' and 'D' of the equation shown in the $Decrypt_2(CT, SK)$ function. Without knowing that secret number, it is impossible to access the data in a PPT. This hardness also demonstrates our scheme as a resistant to the collusion attack. Therefore, any attacker cannot successfully launch the eavesdropping or collusion attack to our proposed scheme.

*The scheme ensures message integrity, non-repudiation, and source authentication*: We use the patient's secret key and the session identity to generate the signature 'S'

(equation (3)). The data receiver can verifies the signature by using the public parameters of the sender, shown in equation (4). This verification ensures the corresponding source authentication. The scheme generates the message token value 'T' by computing the hash value of the concatenated message, patient's identity ($P_{ID}$), and a session sequence number. Only the patient and the eHealth care provider know the patient's original identity and the session sequence number. This token value is also used to generate the signature 'S'. Therefore the message integrity with non-repudiation can be provided by our proposed scheme.

*ESPAC ensures backward and forward secrecy*: The scheme prevents user to access the plaintext before providing the required attributes that satisfy the access policy. On the other hand, any user who drops an attribute should be prevented from accessing the plaintext of the subsequent data exchanged after he drops the attribute, unless the other attributes that he is holding satisfy the access policy. Thus, ESPAC ensures backwards and forward secrecy.

## 7 Performance analysis

To evaluate the performance of the presented ESPAC, we first show the timing cost of operations used in ESPAC. We evaluate the computation timing cost by varying number of attributes. Finally, we analyse the performance of ESPAC to resist DoS attack, and conduct a simulation using NS 2.33.

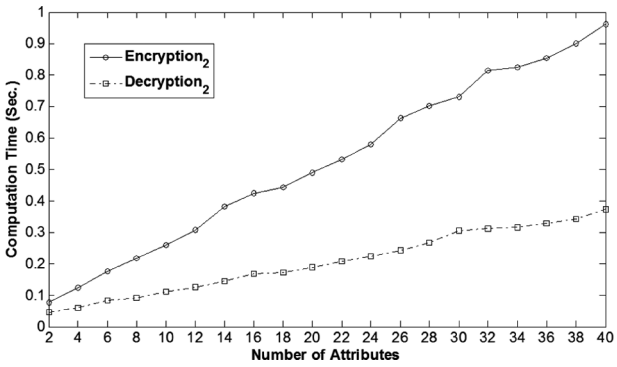
*Time cost*: Based on the open source project CP-ABE (Bethencourt et al., 2011), we evaluate computation time of attribute-based encryption and decryption in a Pentium-IV 3-GHz PC that has 1-GB of RAM. We varying the number of attributes from '2' to '40' and make different access-policies based on these attributes. Encryption and Decryption functions are executed on a 512-byte data-block. Simulation results show that computation time of $Encryption_2$ is increasing linearly with number of attributes, shown in Figure 5. $Decryption_2$ needs less computation time compare to $Encryption_2$. When the number of attributes is '2', $Encryption_2$ and $Decryption_2$ take 76 ms and 45 ms, respectively. These computation time reaches to 962 ms and 372 ms, when the number of attributes is altered to 40. Time cost of ESPAC operations is given in Table 1.

**Table 1** Time cost for ESPAC operations

| *Operation* | *Time* | *Operation* | *Time* |
|---|---|---|---|
| $\text{Encryption}_1$ | $C_e$ | Signature | $C_m$ |
| Verification | $C_e$ | $\text{Decryption}_1$ | $C_e$ |
| $\text{Encryption}_2$ | $C_e + 2C_m$ | $\text{Decryption}_2$ | $2C_e + C_m$ |

We denote by $C_e$ a computation of the pairing, and $C_m$ a scalar multiplication in $G_1$. Usually, pairing operations cost is much more than other computations. A single pairing $C_e$ needs about 10 times more time to compute than a scalar multiplication $C_m$ (Zhu et al., 2005), and our simulation shows a single pairing needs around 65 ms to compute. We consider 550ms as the computation time for the pairing using a PDA (Ramachandran et al., 2007) for further network analysis.

**Figure 5** Computation time of encryption and decryption with different no. of attributes




*Analysis*: The system blocking probability can be increased by HDR traffic, or accessing the system by a large number of misbehaving users at a time. This increased rate of blocking probability is considered as a cause of DoS attack. In our analysis, we aim to minimise the blocking probability by restricting data rate and using multiple servers. We assume that the service provider serves multiple users. Users demand services according to a Poisson process and request independent and identical distributed exponential service time. We use M/M/1/K and M/M/m/K queuing model for analysis and assume that the blocking probability should be less than 30% to provide adequate Quality of Service (QoS) to the users. Blocking probability $P_1(K)$ and $P_m(K)$ of the M/M/1/K and M/M/m/K queue respectively can be written as follow:

$$P_1(K) = \frac{1-\rho}{1-\rho^{K+1}}\rho^i \quad \text{and} \quad P_m(K) = \frac{\rho^m/m!}{\sum_{i=0}^{K}\frac{\rho^i}{i!}};$$

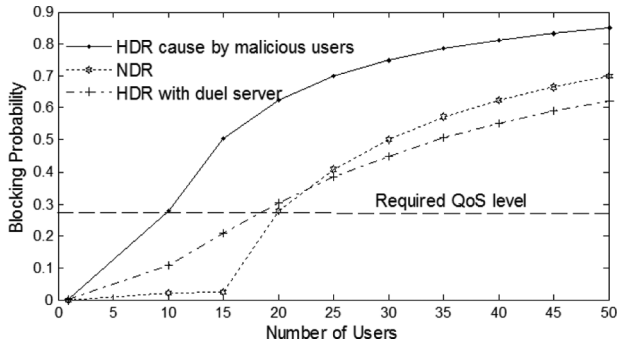
$$\text{for} \quad \rho = \frac{\lambda}{\mu} \neq 1 \quad \text{and} \quad 0 \leq i \leq K.$$

Derivation of above equations can be found in Cassandras and Lafortune (2010). We consider the arrival rates $\lambda$ for the normal and HDR traffic are 3 and 6 per unit of time, respectively, while the service rate $\mu = 10$ is fixed. The number of users, K, varies from 0 to 50. For the M/M/m/K queue, the number of servers $m = 2$.

Figure 6 shows that HDR created by malicious users causes high blocking probability compared to Normal Data Rate (NDR), and ineffective to maintain QoS level for more than 10 users. We can use multiple server with fixed upper bound of the data rate to resist the DoS attack, and to ensure the required QoS with an acceptable number of users.

*eHealth care scenario*: We consider two types of users, wired and wireless, are connected to the eHealth care

**Figure 6** Queuing comparisons for the QoS requirements




service provider. The eHealth care provider is linked to the cloud server through a wired connection. We define two types of scenarios, Normal Scenario (NrS) and High-dense Scenario (HdS), in our model. NrS consists of 5 mobile users and 3 users with wired connection. For the HdS, we just double the respective numbers.

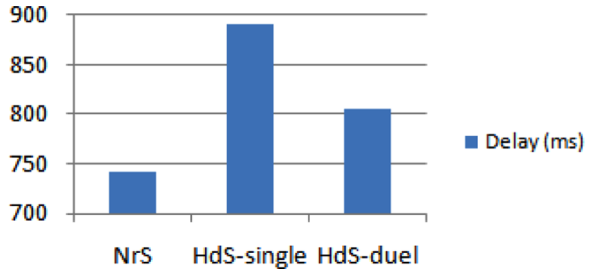
*Network simulation*: Based on the theoretical analysis, we consider NrS, and HdS with single and duel server in our simulation. The performance metric used in our simulation is end-to-end delay, and all the wireless users are assumed to be in the access-point communication range. Table 2 gives the different parameters used in our simulation.

**Table 2** Simulation parameters

| | |
|---|---|
| Simulation time | 150 s |
| Number of nodes | NrS [wireless 5, wired 3] |
| | HdS [wireless 10, wired 6] |
| Packet type | wireless-CBR, Wired-TCP |
| Packet size | 512 bytes |
| Mobility | 2–5 Km/h [for wireless users] |

Figure 7 shows the average end-to-end delay of the different scenarios using the ESPAC scheme.

**Figure 7** Comparison of average end-to-end delay (see online version for colours)




Simulation results show that the average end-to-end delay of the proposed scheme is around 750 ms in a NrS and increases to 900 ms in a HdS, which is minimised to 800 ms by using the duel server. Based on the performance analyses, we can apply ESPAC scheme in a duel server mode to resist DOS attack and provide a high QoS level for users.

*Key storage efficiency*: Compared to the traditional access-control schemes, ESPAC's users do not need to store data requesters' IDs. In the initialisation phase, TA computes the constant size $PK$ and $MK$. Service providers or registered users only need to store the assigned attributes keys. Let, $N$ is the average size of attributes and '$c$' is the number of assigned attributes. Thus, the storage overhead is $\mathcal{O}(c\,logN)$. The proposed scheme ensures users' storage efficiency as they do not need to store data-requesters IDs for future access control. In a trusted environment, they may even release more storage space by storing the access-policy in the trusted service provider end.

## 8 Conclusion

In this paper, we have proposed a scheme, ESPAC, to achieve patient-centric access control with security and privacy by exploiting attribute-based encryption. Moreover ESPAC enables the eHealth care service provider to reduce the overall maintaining cost by moving data to a centralised storage or cloud storage for further processing and long-term storage. Moreover, storing PHIs in the cloud storage provides anytime, anywhere access to stored patient's health information. The proposed scheme also preserves user privacy with data integrity. Through detailed security and performance analyses, it has been demonstrated that the proposed scheme is highly efficient to resist various possible attacks and malicious behaviour. In our future work, we will extend the proposed scheme to support encrypted keyword search in cloud computing.

## Acknowledgements

This research is partly supported by the NSERC (Natural Sciences and Engineers Research Council of Canada).

## References

Barua, M., Alam, M.S., Liang, X. and Shen, X. (2011) 'Secure and quality of service assurance scheduling scheme for wban with application to ehealth', *Wireless Communications and Networking Conference (WCNC), 2011 IEEE*, Cancun, Quintana-Roo, Mexico, pp.1–5.

Bethencourt, J., Sahai, A. and Waters, B. (2007) 'Ciphertext-policy attribute-based encryption', *Security and Privacy, 2007. SP '07. IEEE Symposium on*, Washington DC, USA, pp.321–334.

Bethencourt, J., Sahai, A. and Waters, B. (2011) *Advanced Crypto Software Collection, Ciphertext-Policy Attribute-Based Encryption*, http://acsc.cs.utexas.edu/cpabe/

Boneh, D. and Franklin, M.K. (2001) 'Identity-based encryption from the weil pairing', *CRYPTO '01: Proceedings of the 21st Annual International Cryptology Conference on Advances in Cryptology*, Springer-Verlag, London, UK, pp.213–229.

Cassandras, C.G. and Lafortune, S. (2010) *Introduction to Discrete Event Systems*, Springer, NY,USA.

Goyal, V., Pandey, O., Sahai, A. and Waters, B. (2006) 'Attribute-based encryption for fine-grained access control of encrypted data', *Proceedings of the 13th ACM Conference on Computer and Communications Security', CCS '06*, ACM, New York, NY, USA, pp.89–98.

Johnson, M. (2009) 'Data hemorrhages in the health-care sector', in Dingledine, R. and Golle, P. (Eds.): *Financial Cryptography and Data Security*, Vol. 5628 of Lecture Notes in Computer Science, Springer, Berlin/Heidelberg, pp.71–89.

Kamara, S. and Lauter, K. (2010) 'Cryptographic cloud storage', *Proceedings of the 14th International Conference on Financial Cryptograpy and Data Security, FC'10*, Springer-Verlag, Berlin, Heidelberg, pp.136–149.

Liang, X., Lu, R., Lin, X. and Shen, X. (2010) 'Patient self-controllable access policy on phi in ehealthcare systems', *AHIC 2010*, Kitchener, Ontario, Canada, pp.1–5.

Lin, X., Lu, R., Shen, X., Nemoto, Y. and Kato, N. (2009) 'Sage: a strong privacy-preserving scheme against global eavesdropping for ehealth systems', *Selected Areas in Communications, IEEE Journal on*, Vol. 27, No. 4, pp.365–378.

Liu, Q., Wang, G. and Wu, J. (2010) 'Efficient sharing of secure cloud storage services', *Computer and Information Technology (CIT), 2010 IEEE 10th International Conference on*, Los Alamitos, CA, USA, pp.922–929.

Luna, J., Dikaiakos, M., Marazakis, M. and Kyprianou, T. (2010) 'Data-centric privacy protocol for intensive care grids', *Information Technology in Biomedicine, IEEE Transactions on*, Vol. 14, No. 6, pp.1327–1337.

Lu, R., Lin, X., Liang, X. and Shen, X. (2010) 'A secure handshake scheme with symptoms-matching for mhealthcare social network', *Mobile Networks and Applications*, Springer, Netherlands, pp.1–12.

Ramachandran, A., Zhou, Z. and Huang, D. (2007) 'Computing cryptographic algorithms in portable and embedded devices', *Portable Information Devices, 2007. PORTABLE07, IEEE International Conference on*, Orlando, Florida, pp.1–7.

Sahai, A. and Waters, B. (2005) 'Fuzzy identity-based encryption', in Cramer, R. (Ed.): *Advances in Cryptology — EUROCRYPT 2005*, Vol. 3494 of Lecture Notes in Computer Science, Springer, Berlin/Heidelberg, p.557.

Wang, G., Liu, Q. and Wu, J. (2010) 'Hierarchical attribute-based encryption for fine-grained access control in cloud storage services', *Proceedings of the 17th ACM Conference on Computer and Communications Security, CCS '10*, ACM, New York, NY, USA, pp.735–737.

Yu, S., Wang, C., Ren, K. and Lou, W. (2010) 'Achieving secure, scalable, and fine-grained data access control in cloud computing', *INFOCOM, 2010 Proceedings IEEE*, San Diego, CA, USA, pp.1–9.

Zhang, R. and Liu, L. (2010) 'Security models and requirements for healthcare application clouds', *Cloud Computing (CLOUD), 2010 IEEE 3rd International Conference on*, Miami, FL, USA, pp.268–275.

Zhu, R., Yang, G. and Wong, D. (2005) 'An efficient identity-based key exchange protocol with kgs forward secrecy for low-power devices', *Internet and Network Economics*, Vol. 3828, pp.500–509.