Privacy-Preserving Recommendation Systems for Consumer Healthcare Services

Stefan Katzenbeisser
Information and System Security Group
Philips Research Europe
Eindhoven, The Netherlands
stefan.katzenbeisser@philips.com

Milan Petković
Information and System Security Group
Philips Research Europe
Eindhoven, The Netherlands
milan.petkovic@philips.com

Abstract

Advances in e-health bring new challenges with regard to the protection of sensitive patient data; an increasing number of applications require to share data with consumer healthcare services. Typically the providers of those services reside outside the traditional health care domain, where medical data protection laws (such as HIPAA) do not apply. Instead, technical means of protection should safeguard critical health data that is shared with third parties. In this paper, we show that cryptographic privacy-enhancing protocols are a key tool to protect the privacy of patients in upcoming consumer e-health services. In particular we focus on services offering health advice, allowing to locate specialists and supporting the formation of patient communities.

1 Introduction

Data security and privacy are traditionally important issues in healthcare. Recent developments such as digitization of health records and their use in emerging applications in the personal health care domain pose new challenges towards the protection of patient data. In contrast to other domains, such as the financial sector, which can absorb some costs of system abuse (e.g. credit card fraud), healthcare cannot. Once sensitive information about an individual's health problems is uncovered and social damage is done, it is impossible to revoke the information.

It is expected that the importance of privacy and security in this domain will escalate with the forthcoming applications of personal, user-focused healthcare services. In the traditional healthcare sector, data protection relies to a great extent on regulations (such as HIPAA in the US or the EC Directive 95/46) and procedures. This form of protection is quickly becoming ineffective once information is either shared between healthcare providers or transferred beyond the traditional healthcare domain. Indeed, new applications will demand more consumer/patient involvement at all levels of healthcare. People will take a more active role in their own health management and consequently be more involved in keeping, managing and sharing important and very sensitive health records. It is anticipated that these tasks will require assistance of external service providers, which may not be bound to specific healthcare data protection laws.

This trend of patient empowerment has already been widely supported. First, a number of solutions [2,5,7] have been introduced in the market that allow patients to collect their own health-related information and to store them on portable devices, PCs, and in online services. These solutions are often referred to as Personal Health Record (PHR) services. PHRs store patient health data that is supplied by the patient himself, external third parties such as wellness centers, and health care providers. Already a number of products in the market allow patients to automatically enter measurements and other medical data into their PHRs [4,6]; for example a weight-scale sends its information via Bluetooth to a PC from which the data is uploaded to PHRs. Furthermore, as third parties such as fitness clubs and weight control organizations are professionalizing, they may want to use data from or provide data to a patient's health record—a need that has recently been recognized by the Continua standardization alliance [3]. Finally, healthcare providers are a natural source of patient health records. A PHR system usually comes with applications for both the patient and the care provider (mainly general practitioners), the latter allowing care providers to add information to the patient's PHR.

PHRs thus provide a solid basis for a number of health related services that require patient data, such as fitness, weight management or stress management services. Very often PHRs are coupled with additional basic services for

0-7695-3102-4/08 $25.00 © 2008 IEEE
DOI 10.1109/ARES.2008.85

their users, which allow them to get preliminary advice on symptoms, find a doctor in their vicinity that is specialized in a particular area or allow the creation of communities of patients that have similar diseases. These services, which are usually run at providers outside the traditional healthcare domain, require access to certain (potentially highly sensitive) parts of a patient's PHR. Therefore, the patient often faces the dilemma of either disclosing sensitive information to a third party or keeping it secret and not obtaining the service. One way to solve this dilemma is to apply cryptographic techniques, such as private profile matching techniques which have recently been introduced in cryptography, so that the service provider gets as little information as possible on the patient's PHR data. In those solutions, two parties often want to compute the degree of "similarity" between data they possess, without disclosing the data to each other. The central tool to construct such schemes are semantically secure public key encryption schemes, which allow calculations to be performed on encrypted data without having to access the data in the clear.

In this paper, we study the use of cryptographic private profile matching techniques in the context of healthcare services based on PHRs. In particular, we discuss three different application scenarios: (1) sharing a health profile of a patient with a service provider to get preliminary advice on his health status, (2) finding a doctor who best matches a patient's health profile and (3) creating a community of fellow patients that suffer from similar health problems. In all scenarios, we show that services can be implemented that provably do not reveal any sensitive patient data to the service provider.

The rest of the paper is organized as follows. Section 2 gives a brief overview of the basic cryptographic mechanisms, while Section 3 outlines the cryptographic protocols that are required to implement privacy preserving services. Section 4 details the three above mentioned PHR service scenarios. Finally, Section 5 concludes the paper.

2 Cryptographic Tools

As a central tool to implement privacy-preserving recommendation systems, we use homomorphic public-key encryption schemes. Such schemes allow to compute linear combinations of encrypted values without need for prior decryption. Formally, a (public key) encryption system $E_{pk}(\cdot)$, where pk denotes the public key, is additively homomorphic, if for any messages x and y taken from the message space of the encryption scheme, we have

$$E_{pk}(x+y) = E_{pk}(x) * E_{pk}(y).$$

Given two encryptions $E_{pk}(x)$ and $E_{pk}(y)$, an encryption of the sum $E_{pk}(x+y)$ can thus directly be computed. Note that this property implies that

$$E_{pk}(c \cdot x) = (E_{pk}(x))^c$$

for every integer constant c. Thus, every additively homomorphic cryptosystem also allows multiplication of an encrypted value with a constant available or known as clear text.

The Paillier cryptosystem [10] provides the required homomorphism, if both addition and multiplication are considered as modular. The encryption of a message $m \in \mathbb{Z}_N$ under a Paillier cryptosystem is defined as

$$E_{pk}(m) = g^m r^N \mod N^2,$$

where $N = pq$, p and q are large prime number, $g \in \mathbb{Z}^*_{N^2}$ is a generator whose order divides N, and $r \in \mathbb{Z}_N$ is a random number (blinding factor). We then easily see that

$$\begin{aligned} E_{pk}(x)E_{pk}(y) &= (g^x r_x^N)(g^y r_y^N) \mod N^2 \\ &= g^{x+y}(r_x r_y)^N \mod N^2 \\ &= E_{pk}(x+y). \end{aligned}$$

The Paillier cryptosystem satisfies a strong security property, called semantic security. Due to the use of randomized encryption (for the computation of each encryption of a message m, a new random value r is chosen), an adversary cannot even see whether two encryptions correspond to the same plaintext.

The homomorphic property allows performing linear operations directly on encrypted values. For example, to obtain the encrypted correlation between an encrypted vector $\mathbf{x}$ and a vector $\mathbf{y}$ known in the clear, one can compute

$$\begin{aligned} E_{pk}(<\mathbf{x},\mathbf{y}>) &= E_{pk}\left(\sum_{i=1}^{M} x_i y_i\right) \\ &= \prod_{i=1}^{M} E_{pk}(x_i y_i) \\ &= \prod_{i=1}^{M} E_{pk}(x_i)^{y_i}. \end{aligned} \quad (1)$$

Thus it is possible to compute an inner product directly in case one of the two vectors is encrypted. One takes the encrypted samples $E_{pk}(x_i)$, raises them to the power of y_i and multiplies all obtained values. The resulting number itself is also in encrypted form. Here we implicitly assume that x_i, y_i are represented as integers in the message space of the Paillier cryptosystem (i.e. $x_i, y_i \in \mathbb{Z}_N$) and that all computations do not overflow when performed modulo N.

3 Privacy-Enhancing Recommendation Protocols

In this section, we show how homomorphic public-key encryption can be used to implement privacy-enhancing protocols for profile matching and subset matching.

3.1 Profile Matching

In the case of profile matching, two parties A and B possess two vectors $\mathbf{x}$ and $\mathbf{y}$ respectively. A wants to know the degree of similarity between his vector $\mathbf{x}$ and the vector $\mathbf{y}$; neither party wants to disclose his vector to the other party.

Depending on the similarity measure, different implementations are necessary. Figure 1 shows the implementation of a privacy-preserving profile matching protocol if the similarity between $\mathbf{x}$ and $\mathbf{y}$ is measured by their correlation. This protocol, which utilizes the properties of Equation (1), has repeatedly been applied in the context of privacy protection [9].

If the similarity between $\mathbf{x}$ and $\mathbf{y}$ is measured by the Euclidean distance, $d(\mathbf{x},\mathbf{y}) = \sqrt{\sum_{i=1}^{M}(x_i - y_i)^2}$, then the protocol of Figure 2 can be utilized. In this case, the party A pre-computes encryptions of the squares of the elements of $\mathbf{x}$, which cannot be directly computed under encryption using only homomorphic properties. (This precomputation step increases the communication complexity; however, this is preferable over protocols that require to jointly compute multiplications on encrypted numbers.) Similar protocols have been used in multimedia recommender services [11].

Even though privacy-preserving protocols traditionally add a considerable level of complexity, the protocols of Figure 1 and 2 are extremely efficient: they require only one round of interaction (i.e., A sends data to B and immediately receives the result) and their communication complexity is linear in the length of the vectors. Furthermore, they provably protect the privacy of both A and B: as all operations are performed on encrypted values, the semantic security of E_{pk} immediately implies that B cannot get any information about the vector $\mathbf{x}$, while A obtains only the intended result and cannot derive any further information on B's vector $\mathbf{y}$.

3.2 Subset Matching

In the case of subset matching, two parties A and B encode their preferences as binary vectors $\mathbf{x},\mathbf{y} \in \{0,1\}^M$ of length M. Every entry of $\mathbf{x}$ or $\mathbf{y}$ set to one indicates the preference of a specific condition. A wants to know whether his preference is a subset of the preferences of B, i.e., A needs to check whether the positions marked with ones in his vector $\mathbf{x}$ are also marked with ones in the vector $\mathbf{y}$. We will denote this condition as $\mathbf{x} \subseteq \mathbf{y}$.

The protocol depicted in Figure 3 allows testing the condition $\mathbf{x} \subseteq \mathbf{y}$ in a privacy preserving manner using the principle of secure polynomial evaluation. A starts with encrypting the bits of $\mathbf{x}$ and hands the encryptions over to B, who computes (using homomorphic properties) an encryption of the sum $Z = \sum_{i=1}^{M} x_i(x_i - y_i) = \sum_{i=1}^{M}(x_i - x_i y_i)$. Note that this sum is zero if and only if the positions x_i set to one are also set to one in $\mathbf{y}$. Finally, B blinds the result with a random value and sends the obtained encryption back. A can decrypt the result and declare a match if he obtained a zero.

Again, the protocol is efficient (it requires one round of interaction and linear communication complexity). Furthermore, it provably protects the privacy of both parties: due to the semantic security of the encryption, B cannot access the preferences of A. Furthermore, at the end of the protocol, A only obtains a binary answer.

4 PHR Usage Scenarios

In this section we describe examples of consumer healthcare services that can be implemented in a secure way by using the protocols introduced in Section 3. In particular, we concentrate on three scenarios: getting preliminary health advice, locating a specialist and forming patient communities.

4.1 Health Advice

Nowadays, a number of health services on the Internet offer health advice based on questionnaires submitted by the user. These questionnaires typically include, besides demographic information, subjective and objective health data. The former consists of personal opinions collected from the patients, whereas the latter can be data obtained from measurement devices or retrieved from a PHR.

In the following we consider a simplified version of a health recommendation system, which compares the user's health data with a number of reference profiles. For each reference profile, the similarity with the profile submitted by the patient will be computed. After finding the best matching profile, advice corresponding to the health status is given to the patient. Without any privacy protection in place, the service learns the user's health data. However, as this data is detailed and sensitive, privacy-aware patients may hesitate to use such services. To avoid disclosure of patient's health data, the PHR service can download the reference profiles from the service and make the comparison itself. However, the reference profiles are the most valuable asset of the service provider, who may therefore be reluctant to disclose them.

To avoid both undesirable scenarios, we suggest applying the algorithms for private computations introduced in

Private Input of A: $\mathbf{x} = (x_1, \ldots, x_M)$
Private Input of B: $\mathbf{y} = (y_1, \ldots, y_M)$
Private Output for A: $\mathbf{x} \cdot \mathbf{y}$

A and B engage in the following protocol:

- A generates a pair of public and private keys pk/sk of the Paillier encryption system.
- A encrypts $\mathbf{x}$ to obtain $E_{pk}(\mathbf{x}) = (E_{pk}(x_1), \ldots, E_{pk}(x_M))$ and sends this vector to B.
- B uses his vector $\mathbf{y}$, computes $E_{pk}(\mathbf{x} \cdot \mathbf{y}) = \prod_{i=1}^{M} E_{pk}(x_i)^{y_i}$ and sends the result to A.
- A decrypts the received value to obtain $\mathbf{x} \cdot \mathbf{y}$.

Figure 1. Securely computing the inner product of two vectors.

Private Input of A: $\mathbf{x} = (x_1, \ldots, x_M)$
Private Input of B: $\mathbf{y} = (y_1, \ldots, y_M)$
Private Output for A: $d(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{i=1}^{M} (x_i - y_i)^2}$

A and B engage in the following protocol:

- A generates a pair of public and private keys pk/sk of the Paillier encryption system.
- A encrypts $\mathbf{x}$ to obtain $E_{pk}(\mathbf{x}) = (E_{pk}(x_1), \ldots, E_{pk}(x_M))$. Furthermore, he encrypts all squares of values in $\mathbf{x}$ to obtain $(E_{pk}(x_1^2), \ldots, E_{pk}(x_M^2))$. He sends both vectors to B.
- B uses his vector $\mathbf{y}$ and computes encryptions $E_{pk}(\mathbf{y}) = (E_{pk}(y_1), \ldots, E_{pk}(y_M))$. Finally, B evaluates $E_{pk}(d(\mathbf{x}, \mathbf{y})^2) = \prod_{i=1}^{M} E_{pk}(x_i^2) E_{pk}(x_i)^{-2y_i} E_{pk}(y_i^2)$ and sends the result to A.
- A decrypts the received value and takes the square root to obtain $d(\mathbf{x}, \mathbf{y})$.

Figure 2. Securely computing the Euclidean distance of two vectors.

Private Input of A: $\mathbf{x} = (x_1, \ldots, x_M) \in \{0,1\}^M$
Private Input of B: $\mathbf{y} = (y_1, \ldots, y_M) \in \{0,1\}^M$
Private Output for A: TRUE iff $\mathbf{x} \subseteq \mathbf{y}$

A and B engage in the following protocol:

- A generates a pair of public and private keys pk/sk of the Paillier encryption system.
- A encrypts $\mathbf{x}$ to obtain $E_{pk}(\mathbf{x}) = (E_{pk}(x_1), \ldots, E_{pk}(x_M))$ and sends both vectors to B.
- B uses his vector $\mathbf{y}$ and computes $E_{pk}(Z) = \prod_{i=1}^{M} (E_{pk}(x_i) E_{pk}(x_i)^{-y_i})$. Furthermore, B picks a random value r, multiplicatively blinds the encryption $E_{pk}(Z)$ with r and sends the result $E_{pk}(rZ) = E_{pk}(Z)^r$ back to A.
- A decrypts the received value and declares a match if the decrypted value is zero, otherwise he will declare a mismatch.

Figure 3. Securely matching two binary vectors.

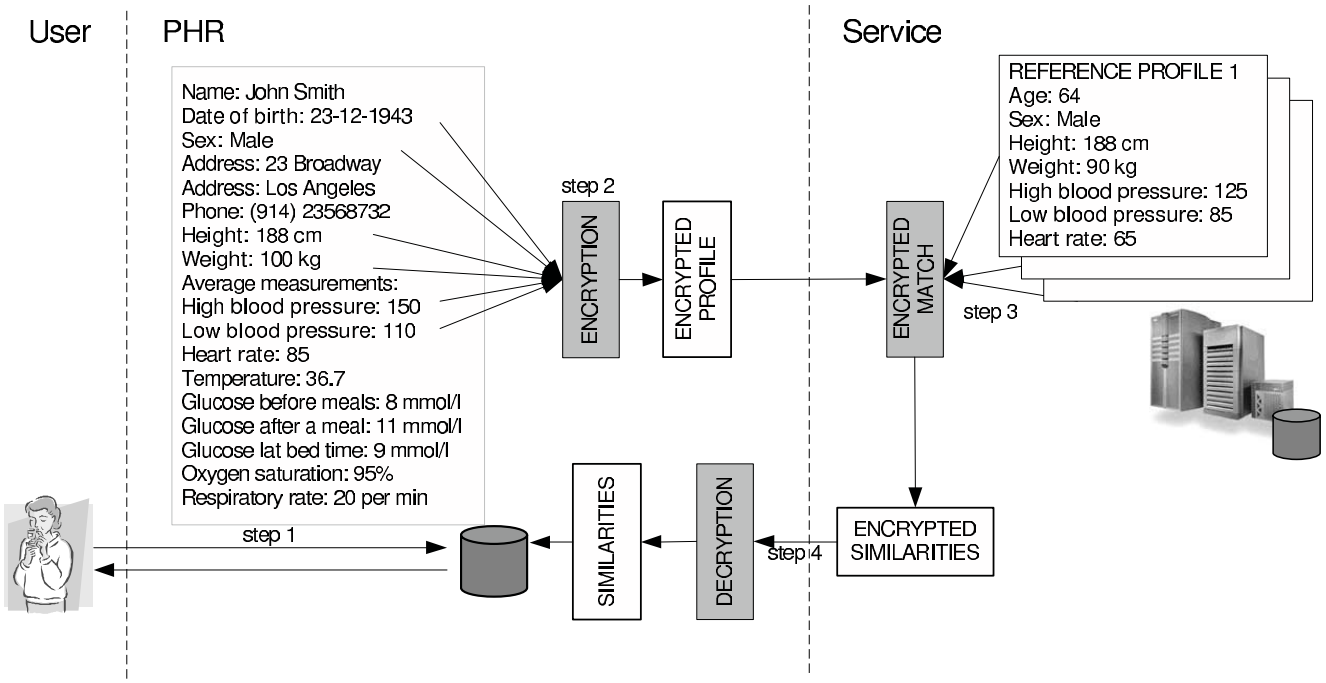

Figure 4. Privacy-preserving health advice service.

Section 3, allowing both the patient and the service provider to keep their data protected. The architecture of the proposed privacy-enhanced advice service is depicted in Figure 4. After receiving a request from the user (step 1), the PHR system generates a public/private key pair. It selects and encrypts required health data from the user's record before sending it to the third party service (step 2 in the figure). The service now performs the similarity calculations directly on the encrypted data; depending on the type of the data that needs to be compared, the service can use one of the protocols from Section 3.1 to compute a distance measure between the user profile and all available reference profiles. This process yields one encrypted value for each reference profile; these encrypted results are sent back to the PHR system (step 4). Finally, the PHR system decrypts the results and shows the degree of similarity with the reference profiles to the user. Additionally, an advice related to the profile that gives the best match will be presented to the user; the similarity ratings will also be added into his PHR for future reference.

Alternatively, if the user does not trust his PHR system, he may choose to store only encrypted values in the PHR. Only a special client software installed at the patient's home computer will be able to access and decrypt the information, while the PHR acts as a pure data storage system. In this case the above described protocol is still applicable; however, the user's trusted client has to decrypt results obtained from the service and find the best matching profile.

4.2 Locating a Specialist

Current "find a doctor" services, such as the one from the American Medical Association [1] or WebMD [8], allow a patient to find general practitioners located in the vicinity or physicians specialized in certain fields (e.g. cardiology, pediatrics, oncology, etc.). The patient is also able to see which medical school granted a doctor a degree, as well as where he did his training and what his major professional activity is.

Typically, a patient is not only interested in finding any doctor, but in finding a good or even the best doctor in a certain area of expertise, related to a very specific medical problem he has. Currently, it is rather difficult for patients to obtain quantitative data (such as statistics of doctor's successfulness, education, scientific standing or mortality rates) which would allow them to objectively select 'the best' doctor for treatment. An intelligent "find a doctor" service could obtain this data at a large scale, interpret it and make ratings or recommendations of specialists available to the general public. The service may offer the option to select the best matching specialist, given certain criteria selected by the patients. However, as in the previous scenario of obtaining health advice, privacy-cautious users may be reluctant to use the service, as the query performed

by the patient potentially leaks information about their medical condition. On the other hand, doctors usually do not want to make raw statistics publicly available.

Therefore, we propose a privacy-preserving find-a-doctor service that safeguards both the privacy of the patients and the confidentiality of the knowledge on which the service is based. The setup of the protocol is similar to that in Figure 4, except that the search criteria and the profiles are encoded differently. The patient encodes, with the help of a PHR server, the required medical expertise as a binary vector: each position in the binary vector corresponds e.g. to a disease or a specific symptom recognized by the patient. In a similar way, the service provider encodes the expertise of all the recommended doctors. The patient (or his PHR service) initiates the protocol by encrypting his query vector and sending it to the service, alongside with some information encoding the region where the doctor's practice should be located. The server performs, for each doctor matching the given location, the subset matching protocol of Section 3.2 to determine whether the doctor has at least the expertise required by the patient. Finally, the service sends the encrypted results, together with the contact details of the recommended doctors, back to the user. The client software (or PHR service) can decrypt, check the result for a match and display the resulting recommendations.

4.3 Community Creation

Some healthcare portals allow the formation of patient communities: patients who suffer from similar diseases can engage in discussions, exchange experiences, or get specific advice. Up to now, these communities usually rely on web forums or mailing lists to disseminate information. A central problem of community creation is the ability of a patient to search for and join a community. A web service can simplify this process: the patients can indicate their intention to join a community and submit their preferences to the service provider, which in turn manages a list of all available patient communities and searches for an appropriate one. However, as in the case of Section 4.2, the patient preferences leak sensitive information about the patient's health and should thus be protected from untrusted remote services.

Using again the privacy-preserving subset matching protocol of Section 3.2, patient communities can be identified without disclosure of sensitive health information. The architecture of the system is depicted in Figure 5(a). A patient who wants to join a patient community encodes his preferences in a vector (again, each position in the vector encodes a disease or a specific symptom), encrypts the data and submits it to the service provider. This provider in turn retrieves a list of available patient communities together with their encoded profiles and engages for each of them in a subset matching protocol. This process will yield to one encrypted value for each patient community registered at the service, indicating whether the community matches the patient's profile. Finally, the service provider submits all obtained encryptions together with the contact information to the client software running on the patient's PC. This software decrypts all received outputs, checks for a match and displays the contact information of all community services that match the given patient profile.

Again, the protocols assure that the service provider cannot access the patient profiles in the clear (the services learn the patient identities only after they have decided to become actual members of the community). Note that the same approach can be used to construct a patient community in a peer-to-peer fashion, based on the arbitration of a PHR server, rather than relying on an external service, as indicated in Figure 5(b). In this case, the patients who are willing to participate in the P2P community register at a PHR server, which maintains a location index of all interested patients. In case a patient indicates his willingness to join a community, the PHR server can distribute the request to all other interested patients, who can then perform the matching process in a bilateral fashion. (Both scenarios assume semi-honest parties, who perform the matching honestly on their input data; this must be assured through external mechanisms, such as reputation management).

5 Conclusions

In this paper we discussed how cryptographic privacy enhancing protocols can be used to secure sensitive patient data in upcoming consumer e-health services, which are offered by parties that are not bound to traditional strict healthcare privacy laws. We showed that private matching protocols can be successfully applied to services offering health advice, help finding specialists or allow creation of patient communities.

Acknowledgements. This work was partially funded by the European Commission through the IST Programme under Contract IST-2006-034238 SPEED. The information in this paper is provided as is, and no guarantee or warranty is given or implied that the information is fit for any particular purpose. The user thereof uses the information at its sole risk and liability.

References

[1] American medical association, doctorfinder for patients. http://webapps.ama-assn.org/doctorfinder/html/patient.html.

[2] Capmed. http://www.phrforme.com/index.asp.

[3] Continua. http://www.continuaalliance.org.

[4] Lifesensor. https://www.lifesensor.com/en/us/.

[5] Medkey. http://www.medkey.com/.

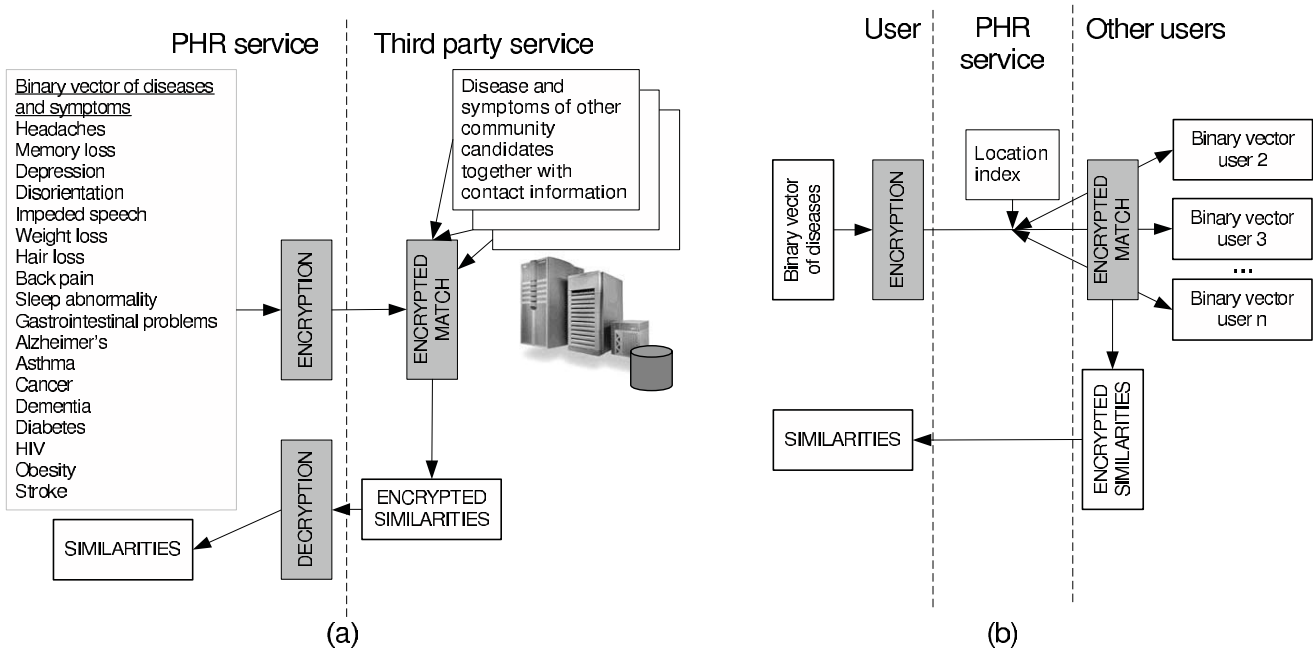

Figure 5. Privacy-preserving patient community creation.

[6] Microsoft healthvault. `http://search.healthvault.com/`.

[7] Webmd. `http://www.webmd.com`.

[8] Webmd physician directory. `http://doctor.webmd.com/physician_finder/home.aspx?sponsor=core`.

[9] B. Goethals, S. Laur, H. Lipmaa, and T. Mielikainen. On Private Scalar Product Computation for Privacy-Preserving Data Mining. *Proceedings of the 7th International Conference on Information Security and Cryptology–ICISC*, pages 104–120, 2004.

[10] P. Paillier. Public-key cryptosystems based on composite degree residuosity classes. In J. Stern, editor, *Advances in Cryptology - EUROCRYPT 1999*, number 1592, pages 223–238, 1999.

[11] W. Verhaegh, A. van Duijnhoven, P. Tuyls, and J. Korst. *Intelligent Algorithms in Ambient and Biomedical Computing*, chapter Privacy Protection in Collaborative Filtering by Encrypted Computation, pages 169–184. Springer, 2006.