Privacy-Preserving Statistical Analysis on Ubiquitous Health Data

George Drosatos and Pavlos S. Efraimidis

Electrical and Computer Engineering, Democritus University of Thrace, University Campus, 67100 Xanthi, Greece
{gdrosato,pefraimi}@ee.duth.gr

Abstract. In this work, we consider ubiquitous health data generated from wearable sensors in a Ubiquitous Health Monitoring System (UHMS) and examine how these data can be used within privacy-preserving distributed statistical analysis. To this end, we propose a secure multi-party computation based on a privacy-preserving cryptographic protocol that accepts as input current or archived values of users' wearable sensors. We describe a prototype implementation of the proposed solution with a community of independent personal agents and present preliminary results that confirm the viability of the approach.

Keywords: Ubiquitous health data privacy, Distributed statistical analysis, Personal data, Secure multi-party computation, Mutli-agent system.

1 Introduction

The use of statistical methods is an integral part of medical research. A medical statistic may comprise a wide variety of data types, the most common of which are based on vital records (birth, death, marriage), morbidity (incidence of disease in a population) and mortality (the number of people who die of a certain disease in relation with the total number of people). Other well-known statistical data that are used are the health care costs, the demographic distribution of a disease based on geographic, ethnic, and gender criteria, and data on the socioeconomic status and education of health care professionals.

At the same time, the advances in wearable sensor technology have dramatically increased the amount of health monitoring data that can be efficiently generated, stored and processed. This led to the emergence of Ubiquitous Health Monitoring Systems (UHMS's) [11,19,17] that use these health data. The data from wearable sensors like any health data are sensitive personal data. Thus, the operation of UHMS's systems must ensure the protection of the patients' privacy. Examples of data types that are used in health monitoring as they are reported in [4] are: heart rate, blood pressure, galvanic skin response, skin temperature, heat flux, subject motion, speed and the distance covered. One of the main features of a UHMS is to automatically generate alerts to notify the family or the patient's doctor about a possible health emergency. The need for UHMS

S. Furnell, C. Lambrinoudakis, and G. Pernul (Eds.): TrustBus 2011, LNCS 6863, pp. 24–36, 2011.

systems is expected to continuously rise in the foreseeable future. The population of the developed world is growing older, medical costs are rising, and there are not enough doctors to heal the elderly. UHMS systems are also important for special groups of people of any age who have the need of continuous health monitoring. The (aimed) benefits of a UHMS is both to reduce the number of visits to the hospitals and to support better health services that may lead to saving the lives of patients.

In this work, we examine how ubiquitous health data generated from wearable sensors in a Ubiquitous Health Monitoring System (UHMS) can be used within privacy-preserving distributed statistical analysis. To this end, we propose a secure multi-party computation based on a privacy-preserving cryptographic protocol that accepts as input current or archived values of users' wearable sensors. This distributed computation is performed by a community of personal agents; each patient has a personal agent which is continuously on-line and collects the medical data of its owner. In addition to the data that are obtained by wearable sensors, the agents may also contain other data such as demographic elements about the patient and further information about his health records, as well. Finally, we describe a prototype implementation of the proposed solution with a community of personal data management agents and present preliminary results that confirm the viability of the approach.

Some of the advantages of our approach in comparison to traditional statistical analysis techniques are:

- Performing statistical analysis on real time, up-to-date data.
- Utilizing valuable, sensitive personal data while ensuring privacy.
- Simplifying the process and reducing the time and cost for conducting a statistical analysis.
- Avoiding errors in data entry, which leads to more reliable results.

In the proposed solution, each patient must have a personal agent at his disposal and permanent access to the Internet. The personal agent collects and preserves the personal data of the patient. The computational requirements for the personal agent can be fulfilled with commodity hardware and hence its cost is not high. Thus, it is plausible to assume that patients with a UHMS can afford the extra cost for such an agent.

2 Related Work

The problem of distributed statistical analysis of this work is a secure multi-party computation (MPC) on extremely critical personal medical data. The general model of a MPC was firstly proposed by Yao [20] and later was followed by many others. In general, a MPC problem concerns the calculation of a function with inputs from many parties, where the input of each participant is not disclosed to anyone. The only information that should be disclosed is the output of the computation. The general solution for MPC presented in [20] is powerful but commonly leads to impractical implementations.

A secure two-party computation (S2C) for the calculation of statistics from two separate data sets is presented in [8]. Each data set is owned by a company and is not disclosed during the computation. Similar results are shown in [9], this time focusing on linear regression and classification and without using cryptographic techniques. Two indicative works from the related field of privacy-preserving data mining are [15,10]; A major difference of our work from the above is that in our approach every participant is in control of his health data and that the distributed computation is performed by the community of the personal agents.

Another approach for statistics on personal data is anonymization, i.e., the sanitization of a data collection by removing identifying information. The data anonymization approach and some of its limitations are discussed for example in [2,16]. Data anonymization applies to collections data in central databases and is not directly comparable to our decentralized approach. Finally, an example of an efficient privacy-preserving distributed computation is given in [7], where personal agents of doctors execute a distributed privacy-preserving protocol to identify the nearest doctor to an emergency. The focus of the present work is on privacy-preserving distributed statistical analysis using a massive number of participants.

3 The Proposed Solution

We propose a system for performing privacy-preserving statistical analysis. The system is build on top of a UHMS, and more precisely, on top of the privacy-enhanced UHMS presented in [6]. An overview of the architecture of the statistical analysis system is shown in Figure 1 with emphasis on the extra components that have to be added to a UHMS; the Network Community of Personal Agents and the Statistical Analysis Service (SAS). More analytically, all the data that are obtained by users' wearable sensors are sent and stored in their personal agents. The personal agent of each patient manages his personal data, provides controlled access to these data, and has the ability to participate in distributed computations. In addition to the medical data obtained by the wearable sensors, the personal agent may also contain other personal data such as demographic information, medical drugs and health record data of the patient.

The personal agents are organized into a virtual topology, which may be a simple ring topology or a more involved topology for time-critical computations. On the other hand, the SAS is a server that initiates the distributed computation on the users' medical data and collects the aggregate results. Each researcher who wishes to carry out a statistical research can submit his task to the SAS.

4 The Main Steps of the Calculation

The main steps of the proposed statistics calculation procedure are:

– Initially, the researcher who wishes to carry out a statistical analysis on the critical medical data submits his request to the SAS.

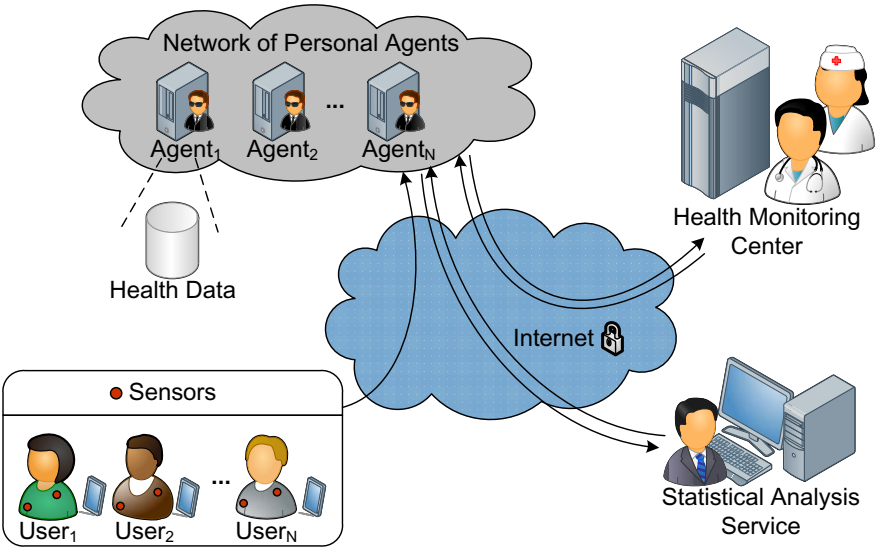

Fig. 1. The general architecture of our system

- The SAS accepts the request after verifying the credentials of the researcher.
- The SAS picks one of the personal agents which will serve as the root node for the specific computation and submits the request to it.
- The root-node coordinates a distributed computation that calculates the specified statistical function.
- At the end of the distributed computation, the SAS and the researcher will only learn aggregate results of the computation without any additional information of the actual personal data.

5 The Secure Distributed Protocol

In this section, we present the main idea of the cryptographic protocol that is used in the privacy-preserving statistical computations. The protocol is secure in the Honest-But-Curious (HBC) model (see Section 7), where the users' agents participating in the computation follow the protocol steps but may also try to extract additional information. During the calculation the actual users' personal data are not disclosed in any stage of the process but only the aggregate results are revealed at the end. An instance of a statistical computation consists of:

- **N patients** $P_1, P_2, \ldots, P_N$ and their personal data.
- **The statistical computation:** The agents of all patients perform a distributed privacy-preserving computation.
  - **Input:** The type of the statistical function/s and its parameters. In addition, selectivity constraints for the data set may also be specified.

- **Output:** The necessary values (e.g. w_x, u_x, z_{xy} and n) that are needed to calculate the statistical function/s.

Assume the following statistical computation instance: Computing the average of the female patients' age in a city. Given the computation instance, the SAS chooses a node from the network of the users' agents as the root-node for the particular computation. Then, the SAS sends the type of the requested computation and its parameters to the root node. The parameters of the computation, i.e., the female gender and the city name, are used to filter the data set. Each personal agent, decides privately if it participates in the statistic research.

A simple topology for the personal agents is a virtual ring topology that contains as nodes all agents (Figure 2.b). For time-critical computations, more complex topologies like a virtual tree can be used (Figure 2.a). The tree topology is used in [7]. At the end of the execution, the root-node collects the results of the calculation as an encrypted message and sends it to the SAS. The message is encrypted with the public key of the SAS, which should be known to all nodes. In this way, the protocol ensures k-anonymity (see Definition 4), where $k = N$ and N is the number of all the nodes in the network.

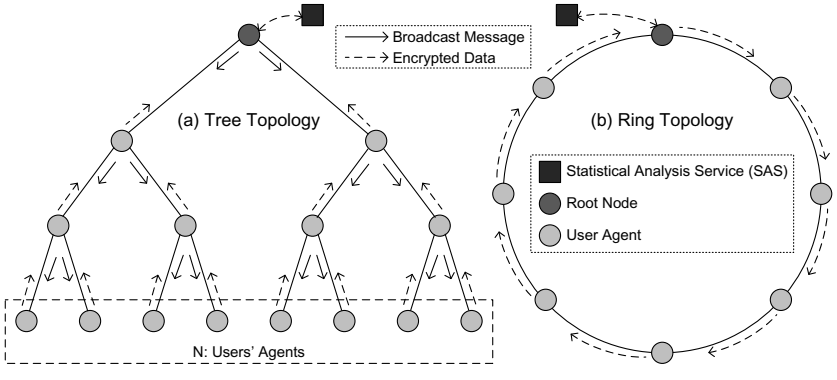

Fig. 2. Possibles network topologies

We use the Paillier public key cryptosystem [18] for the proposed cryptographic protocol. An important feature of the Paillier cryptosystem is its homomorphic property.

Definition 1. Paillier Cryptosystem: *The Paillier cryptosystem is a probabilistic asymmetric algorithm for public key cryptography that is based on the Diffie-Hellman key agreement.*

Definition 2. Homomorphic Encryption: *The homomorphic encryption is a form of encryption where one can perform a specific algebraic operation on the plaintext by performing a (possibly different) algebraic operation on the ciphertext.*

The additive homomorphic property of the Paillier cryptosystem is shown in the following equation:

$$\begin{aligned}\mathcal{E}(x_1) \cdot \mathcal{E}(x_2) &= (g^{x_1} \cdot r_1^{n_p}) \cdot (g^{x_2} \cdot r_2^{n_p}) \\ &= g^{[x_1+x_2 \bmod n_p]} \cdot (r_1 r_2)^{n_p} \bmod {n_p}^2 \\ &= \mathcal{E}([x_1 + x_2 \bmod n_p])\end{aligned}$$

where

- x_1 and x_2 are two plain messages such that $x_1, x_2 \in \mathbb{Z}_{n_p}$,
- (n_p, g) is the Paillier public key,
- r_1 and r_2 are two random numbers such that $r_1, r_2 \in \mathbb{Z}^*_{n_p}$, and
- $\mathcal{E}(m) = g^m r^{n_p} \bmod {n_p}^2$ is the encryption of message m.

6 The Computations

In this section, we use our approach to calculate representative statistical functions with a distributed privacy-preserving computation. Wherever it is necessary, the expression of the statistical function is brought to a form that is appropriate for the distributed computation.

6.1 Arithmetic Mean

The arithmetic mean of a variable X (with sample space $\{x_1, \ldots, x_n\}$) is computed by the following equation:

$$\bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i$$

We use the additive homomorphic property of Paillier to calculate the value of the terms $u_x = \sum_{i=1}^{n} x_i$ and n. The calculation is privacy-preserving; no single x_i information is disclosed. Once the SAS learns the values of the terms u_x and n, it can compute the arithmetic mean. More analytically, using the homomorphic property of Paillier, the two terms u_x and n can be transformed into the following form:

$$E_{pk}(u_x) = \prod_{i=1}^{n} E_{pk}(x_i) \quad \text{and} \quad E_{pk}(n) = \prod_{i=1}^{n} E_{pk}(1)$$

where the E_{pk} indicates that the message is encrypted with the current public key of SAS for the specific statistical analysis. Each agent i that participates in the statistical analysis, calculates the $E_{pk}(x_i)$ and $E_{pk}(1)$ and multiplies the current two encrypted values that are calculated from the above two products. Agents that do not participate in the statistical computation (because for example they do not satisfy some selection criterion) multiply each of the above two products with a different encryption of zero $E_{pk}(0)$.

6.2 Frequency Distribution

The frequency distribution is a tabulation of the values that one or more variables take in a sample. Each entry in the table contains the frequency or count of the occurrences of values within a particular group or interval, and in this way the table summarizes the distribution of values in the sample. The graphical representation of frequency distribution is the well known histogram. Figure 3 indicates how the frequency distribution would become by using ciphertext as counters in each range, where each ciphertext is represented by the following equation:

$$E_{pk}(n_v) = \prod_{i=1}^{n} E_{pk}(m) \text{ , where } m = \begin{cases} 1, & x \in [x_{v-1}, x_v) \\ 0, & x \notin [x_{v-1}, x_v) \end{cases}$$

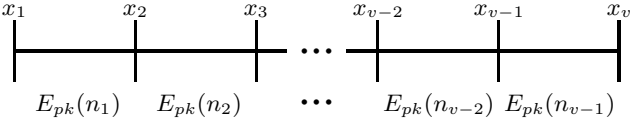

Fig. 3. Representation of a frequency distribution

6.3 Linear Regression

The linear regression of a dependent variable Y of the regressors X is given by the equation $y = a + bx$, where a and b are parameters. The determination of a and b gives an approximate line, which connects the values of Y with the corresponding values of X. This line can be constructed by using the method of least squares and the parameters a and b are given by the following equations:

$$b = \frac{n\sum_{i=1}^{n} x_i y_i - \sum_{i=1}^{n} x_i \sum_{i=1}^{n} y_i}{n\sum_{i=1}^{n} x_i^2 - \left(\sum_{i=1}^{n} x_i\right)^2} \text{ and } a = \frac{1}{n}\sum_{i=1}^{n} y_i - b\frac{1}{n}\sum_{i=1}^{n} x_i$$

The unknown terms that are required to calculate the parameters of line y with the help of the homomorphic property of Paillier are the $w_x = \sum_{i=1}^{n} x_i^2$, $u_x = \sum_{i=1}^{n} x_i$, $u_y = \sum_{i=1}^{n} y_i$, $z_{xy} = \sum_{i=1}^{n} x_i y_i$ and n, by taking the following form:

$$E_{pk}(w_x) = \prod_{i=1}^{n} E_{pk}(x_i^2), \; E_{pk}(u_x) = \prod_{i=1}^{n} E_{pk}(x_i),$$

$$E_{pk}(u_y) = \prod_{i=1}^{n} E_{pk}(y_i), \; E_{pk}(z_{xy}) = \prod_{i=1}^{n} E_{pk}(x_i y_i) \text{ and } E_{pk}(n) = \prod_{i=1}^{n} E_{pk}(1) \, .$$

6.4 Covariance

The covariance $cov(X, Y)$ of two random variables X and Y is a measure of the strength of the correlation between the two variables and is defined as:

$$cov(X,Y) = \frac{1}{n}\sum_{i=1}^{n} x_i y_i - \frac{1}{n}\sum_{i=1}^{n} x_i \cdot \frac{1}{n}\sum_{i=1}^{n} y_i = \frac{1}{n}\sum_{i=1}^{n} x_i y_i - \frac{1}{n^2}\sum_{i=1}^{n} x_i \sum_{i=1}^{n} y_i$$

The unknown terms that are required to calculate the covariance with the help of the homomorphic property of Paillier are the $u_x = \sum_{i=1}^{n} x_i$, $u_y = \sum_{i=1}^{n} y_i$, $z_{xy} = \sum_{i=1}^{n} x_i y_i$ and n, by taking the following form:

$$E_{pk}(u_x) = \prod_{i=1}^{n} E_{pk}(x_i), \;\; E_{pk}(u_y) = \prod_{i=1}^{n} E_{pk}(y_i)$$

$$E_{pk}(z_{xy}) = \prod_{i=1}^{n} E_{pk}(x_i y_i) \;\text{ and }\; E_{pk}(n) = \prod_{i=1}^{n} E_{pk}(1)$$

6.5 Comments

From the analysis of the above statistical functions, we conclude that, except the frequency distribution, all other can be simultaneously calculated by computing once the required unknown terms. Moreover, it is clear that the proposed solution can be used to calculate also other statistical functions, such as the variance, the linear correlation coefficient and so on.

7 The Protocol's Security

In this section, we demonstrate that the proposed protocol of a distributed statistical analysis in a UHMS does not violate the privacy of participants. The security holds for the model of Honest-But-Curious (HBC) users.

Definition 3. Honest-But-Curious (HBC): *An honest-but-curious party (adversary) [1] follows the prescribed protocol properly, but may keep intermediate computation results, e.g. messages exchanged, and try to deduce additional information from them other than the protocol result.*

The security of the Paillier cryptosystem and its homomorphic property ensures that the personal medical data are not disclosed and cannot be associated with any particular patient. We will use the concept of k-anonymity.

Definition 4. k-anonymity: *An informal definition of k-anonymity in the context of this work is that no less than k individual users can be associated with a particular personal value. For a more general definition of k-anonymity that is also valid in databases see [5].*

The main security features of the protocol are:

- Each agent that receives a message from the previous node cannot obtain information about the contents of the message, because the ciphertexts are encrypted with the Paillier encryption.
- Each node alters the ciphertexts of the computation. Even the nodes that do not participate in the statistical function multiply the ciphertexts with an encrypted number "0", which is the neutral element of the additive homomorphic property of Paillier.
- At the end of the protocol, only the variables that are needed for a particular statistical function are revealed. As a result, no individual can be associated with the value that he had used in the computation. Consequently, the proposed protocol preserves k-anonymity for $k = N$, where N is the number of all agents in the network.

Another criterion for protecting privacy is the concept of differential privacy.

Definition 5. ϵ-Differential privacy [12]: *A randomized function $\mathcal{K}$ gives ϵ-differential privacy if for all data sets D_1 and D_2 differing on at most one element, and all $S \subseteq Range(\mathcal{K})$, the following holds:*

$$Pr[\mathcal{K}(D_1) \in S] \leq exp(\epsilon) \times Pr[\mathcal{K}(D_2) \in S]$$

The probability is taken is over the coin tosses of $\mathcal{K}$.

In our solution, the differential privacy is meaningful only if the SAS may find out the identities of participants in the distributed computation. Otherwise, the privacy is guaranteed by k-anonymity. If the SAS knows or may find out the identify of participants, then the concept of differential privacy applies and the common technique to assure differential privacy is to add appropriate additive random noise to the results [13]. In addition, for statistical computations on dynamic data such as the wearable sensors' data, the data which are used in the calculations contain by default some kind of random noise and this enhances the differential privacy.

8 Experimental Results

To evaluate our solution, we developed a prototype that carries out distributed statistical analysis on medical data. The application is implemented in Java and for the cryptographic primitives the Bouncycastle [3] library is used. The personal agents of the Polis platform developed in [14] are used as the personal data management agents of the patients. For this approach, the Polis agents were suitably modified so as to be able to manage both health records and health data that would actually be collected through a secure communication channel by the patients' wearable sensors. The community of the personal agents are organized as a Peer-to-Peer network. At this stage of development of the prototype, the backbone of the topology is a virtual ring topology. The ring offers

a simple and reliable solution for the interconnection of the agents. For time-critical calculations or even real-time calculations of statistics a more involved topology like a virtual tree should be used.

The personal agents use production-ready cryptographic libraries and employ 1024 bits RSA X.509 certificates. The communication between agents is performed over secure sockets (SSL/TLS) with both client and server authentication. Below we describe an experiment of a distributed statistical analysis with 6 agents and the SAS. The requested statistic:

– *The arithmetic mean of the current body temperature of patients who are aged between 55 and 65 years old and their gender is female.*

For the needs of the experiment, each agent generated random values for the age and the gender, and for the current body temperature as well. In brief, the process has as follows. Initially, the SAS randomly chooses a node from the agents' network, in this case the agent 'Patient2', as the root-node and forwards the description of the statistical computation to it. The values of each agent which are related to the computation are shown in Table 1. The last two columns show the aggregate values that are encrypted after the corresponding agent applies its values to the results. Since the homomorphic property of Paillier applies to integers, the body temperature should be rounded to a number with at most two decimal digits and then multiplied by 100 to become an integer.

Table 1. Example of computation, where the agents in gray rows did not take part in computation

Agent	Curr. Temp.	Age	Gender	$E_{pk}(u_x) = \prod_{i=1}^{n} E_{pk}(x_i)$	$E_{pk}(n) = \prod_{i=1}^{n} E_{pk}(1)$
Patient2	36.68 ^oC	51	Female	$E(0)$	$E(0)$
Patient3	36.50 ^oC	56	Female	$E(3650)$	$E(1)$
Patient4	37.70 ^oC	60	Female	$E(7420)$	$E(2)$
Patient5	38.10 ^oC	65	Female	$E(11230)$	$E(3)$
Patient6	37.12 ^oC	59	Male	$E(11230)$	$E(3)$
Patient1	36.20 ^oC	63	Female	$E(14850)$	$E(4)$

At the end of the computation, the agent 'Patient2' as the root-node collects the results and sends them back to the SAS. Finally, the SAS decrypts the results and eventually finds that the average of the question which was submitted is 37.125 ^oC. A snapshot of the application during the execution of the experiment is shown in Figure 4.

We evaluated the efficiency of our solution with a series of experiments on a gradually increasing number of up to 300 agents. The corresponding running times are shown in Figure 5. For this experiment, a network of 30 computer workstations with Intel Core 2 Quad Q8300 CPU's at 2.5 GHz, 2 GB RAM and a 100 Mbps network, was used. Each computer was shared by at most 10 agents, to ensure that no single workstation will be overloaded; an overloaded workstation would become a bottleneck that could significantly delay the execution of the whole protocol.

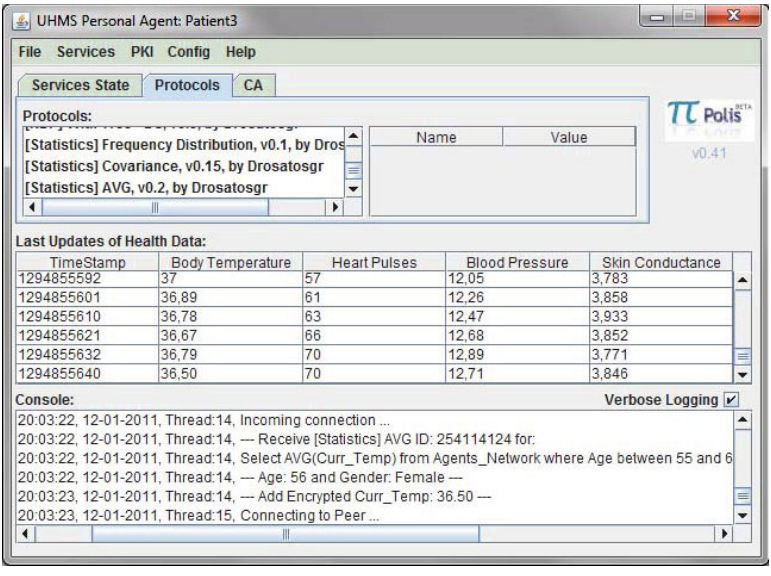

Fig. 4. A snapshot of the agent 'Patient3'

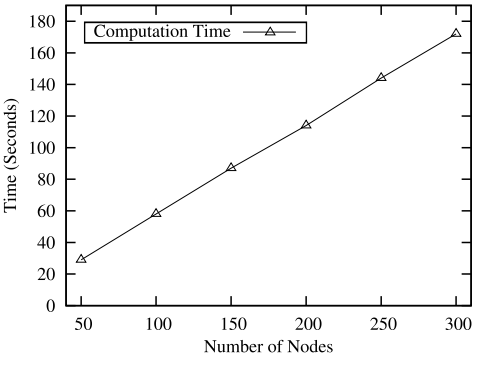

Fig. 5. Computation times of the protocol with respect to the number of agents

9 Conclusion

In this paper, we proposed the use of the ubiquitous health data that are obtained by the wearable sensors in a UHMS for carring out statistical researches. The proposed scheme utilizes the users' personal data while ensuring their privacy. The protection of privacy is achieved by using cryptographic techniques and performing a distributed computation within a network of personal agents. We described how representative statistical functions can be executed distributedly

by using the proposed cryptographic protocol. Finally, we developed a prototype implementation and confirmed the viability and the efficiency of the proposed solution.

The present work can be extended to support more complex statistical functions like nonlinear regression and possibly to conduct privacy-preserving time series analysis on sensors' data. Another interesting direction would be to investigate the optimal adaptation of the differential privacy criterion (and the required random noise) in the context of our distributed computation.

References

1. Acquisti, A., Gritzalis, S., Lambrinoudakis, C., De Capitani di Vimercati, S.: Digital privacy. Auerbach Publications, Taylor & Francis Group (2008)
2. Aggarwal, C.C.: On k-anonymity and the curse of dimensionality. In: VLDB 2005, pp. 901–909 (2005)
3. Bouncycastle Java Library (January 2011), `http://www.bouncycastle.org/`
4. Camous, F., McCann, D., Roantree, M.: Capturing personal health data from wearable sensors. In: SAINT 2008, pp. 153–156. IEEE, Los Alamitos (2008)
5. Ciriani, V., Capitani di Vimercati, S., Foresti, S., Samarati, P.: κ-anonymity. In: Secure Data Management in Decentralized Systems. Advances in Information Security, vol. 33, pp. 323–353. Springer, Heidelberg (2007)
6. Drosatos, G., Efraimidis, P.S.: Privacy-enhanced management of ubiquitous health monitoring data. In: PETRA 2011. ACM, New York (2011)
7. Drosatos, G., Efraimidis, P.S.: A privacy-preserving protocol for finding the nearest doctor in an emergency. In: PETRA 2010, pp. 18:1–18:8. ACM, New York (2010)
8. Du, W., Atallah, M.: Privacy-preserving cooperative statistical analysis. In: ACSAC 2001, pp. 102–112. IEEE, Los Alamitos (2001)
9. Du, W., Chen, S., Han, Y.S.: Privacy-preserving multivariate statistical analysis: Linear regression and classification. In: SDM 2004, pp. 222–233 (2004)
10. Duan, Y., Youdao, N., Canny, J., Zhan, J.Z.: P4P: practical large-scale privacy-preserving distributed computation robust against malicious users. In: USENIX Security Symposium, pp. 207–222 (2010)
11. Durresi, A., Durresi, M., Barolli, L.: Secure ubiquitous health monitoring system. In: Takizawa, M., Barolli, L., Enokido, T. (eds.) NBiS 2008. LNCS, vol. 5186, pp. 273–282. Springer, Heidelberg (2008)
12. Dwork, C.: Differential privacy: a survey of results. In: Agrawal, M., Du, D.-Z., Duan, Z., Li, A. (eds.) TAMC 2008. LNCS, vol. 4978, pp. 1–19. Springer, Heidelberg (2008)
13. Dwork, C., McSherry, F., Nissim, K., Smith, A.: Calibrating noise to sensitivity in private data analysis. In: Halevi, S., Rabin, T. (eds.) TCC 2006. LNCS, vol. 3876, pp. 265–284. Springer, Heidelberg (2006)
14. Efraimidis, P.S., Drosatos, G., Nalbadis, F., Tasidou, A.: Towards privacy in personal data management. J. IMCS 17(4), 311–329 (2009)
15. Kantarcioglu, M., Kardes, O.: Privacy-preserving data mining in the malicious model. Int. J. IJICS 2(4), 353–375 (2008)

16. Muntés-Mulero, V., Nin, J.: Privacy and anonymization for very large datasets. In: CIKM 2009, pp. 2117–2118. ACM, New York (2009)
17. Otto, C., Milenkovic, A., Sanders, C., Jovanov, E.: System Architecture of a Wireless Body Area Sensor Network for Ubiquitous Health Monitoring. J. JMM 1, 307–326 (2006)
18. Paillier, P.: Public-key cryptosystems based on composite degree residuosity classes. In: Stern, J. (ed.) EUROCRYPT 1999. LNCS, vol. 1592, pp. 223–238. Springer, Heidelberg (1999)
19. Yamazaki, A., Koyama, A., Arai, J., Barolli, L.: Design and implementation of a ubiquitous health monitoring system. Int. J. Web Grid Serv. 5, 339–355 (2009)
20. Yao, A.C.C.: Protocols for secure computations (extended abstract). In: FOCS 1982, pp. 160–164. IEEE, Los Alamitos (1982)