Pers Ubiquit Comput (2017) 21:831–839
DOI 10.1007/s00779-017-1056-7

ORIGINAL ARTICLE

Utilizing fully homomorphic encryption to implement secure medical computation in smart cities

Xiaoqiang Sun[1] · Peng Zhang[1] · Mehdi Sookhak[2] · Jianping Yu[1] · Weixin Xie[1]

Received: 7 Sep 2016 / Accepted: 16 Apr 2017 / Published online: 1 July 2017

Abstract As healthcare is one of major socioeconomic problems in cities, mobile healthcare network becomes one of core components of smart cities, which would improve urban healthcare environment. However, there are wide privacy concerns as personal health information is outsourced to untrusted cloud servers. It is a promising method to encrypt the health data before outsourcing, but how to do diagnosis computations on the encrypted health data remains an important challenge. In this paper, we propose a general architecture of the mobile healthcare network, and define three typical secure medical computations, which include the average heart rate, the long QT syndrome detection, and the chi-square tests. To achieve computations on the ciphertext, we leverage fully homomorphic encryption (FHE) to encrypt the health data. Different from previous related works, we use more efficient Dowlin's FHE scheme to implement above three medical computations. In our implementation of the average heart rate, only one ciphertext is sent back to the receiver, so homomorphic decryption is needed once. We take an efficient l-bits comparator to implement the long QT syndrome detection, which only needs l XOR operations and one homomorphic multiplication. We first implement the chi-square tests by homomorphic additions and homomorphic multiplications, which can be used to study whether varicose veins is relevant to overweight. Extensive simulations and analytical results show the scalability and efficiency of our proposed scheme.

Keywords Smart cities · Mobile healthcare network · Secure medical computation · Fully homomorphic encryption

1 Introduction

Cities with the large and centralized population, relying heavily on energy infrastructures and the quality of public services, are experiencing unprecedented crises, including terrorist attacks, viral transmission, natural disasters, etc. Smart cities [1, 2], proposed by IBM in 2008, are increasingly looking for information and communication technology to optimize the management of city affairs. Within this paradigm, intelligent transportation systems, smart power system, mobile healthcare networks, etc. are forming as components of smart cities. Healthcare is one of major socioeconomic problems, especially in the crowding city, where it needs enormous health expense and labor resources. Mobile healthcare network collects the health information sensed by wearable devices, analyzes/processes for health monitoring and diagnosis, and enables users' social interactions [3]. Compared with traditional hospital-centric healthcare,

✉ Peng Zhang
zhangp@szu.edu.cn

Xiaoqiang Sun
xiaoqiangsun1989@gmail.com

Mehdi Sookhak
m.sookhak@ieee.org

Jianping Yu
yujp@szu.edu.cn

Weixin Xie
wxxie@szu.edu.cn

[1] ATR Key Laboratory of National Defense Technology, College of Information Engineering, Shenzhen University, Shenzhen, China

[2] Department of Systems and Computer Engineering, Carleton University, Ottawa, Canada

they can avoid excessive waiting times in hospitals, enhance disease diagnosis, especially when dealing with the chronic disease or preventing several serious diseases at the early stages, and reduce the heavy burden of the healthcare cost.

In the mobile healthcare network, dedicated wearable devices can continuously measure and collect personal health information (for example, body temperature [4], heart rate [5], blood pressure [6]). Next, the health information collected from wearable devices can be outsourced to cloud servers. And then, doctors can use computers or smart phones to acquire these health data remotely. In case of any emergency, such as a heart problem, wearable devices can automatically report the health data to doctors. However, since the health data is relatively sensitive for the patient, any inappropriate disclosure may violate user privacy and even result in property damage. Cloud servers are untrusted will, no doubt, aggravate users' worries about their critical health data leakage. A feasible and promising approach would be to encrypt the data before outsourcing, and then the encrypted data is stored on cloud servers.

However, cloud servers are not only used to data storage but also data computation. The collected data from wearable devices is raw, so in order to help doctors make a diagnosis, it should be further processed by cloud servers, e.g., the average heart rate and the long QT syndrome detection [7], which are used for the diagnosis of the heart problem, and the chi-square tests, which are helpful to research whether varicose veins is relevant to overweight. Achieving diagnosis computation on the cloud requires operation on the encrypted data. We could leverage fully homomorphic encryption (FHE) [8] techniques to implement computations on encrypted data. FHE enables computations of arbitrary functions on the ciphertext, and thus generates a ciphertext that when decryption matches the result of operations on the plaintext. The encrypted result can be used to protect user privacy and support diagnosis computation at the same time.

The fully homomorphic encryption scheme was first proposed by Gentry [8] based on ideal lattices in 2009. Following Gentry's scheme, number of FHE schemes have been proposed to make FHE more practical. Based on the approximate greatest common divisor problem, Dijk et al. [9] constructed a simple FHE scheme over the integers that only trivial operations are applied. FHE schemes [10–12] based on the learning with errors (LWE) assumption [13] have the advantage without increasing the secret key size after each homomorphic multiplication. Compared to the LWE assumption, the learning with errors over rings (RLWE) assumption [14] has simpler algebraic structure and higher efficiency, which has been used to construct efficient FHE schemes [15–18]. Brakerski et al. [19] proposed the leveled FHE scheme without the bootstrapping procedure, called BGV scheme. It has 2^{λ} security against known attacks. In 2013, Bos et al. [20] constructed a new FHE scheme based on the ref. [21], in which only one ring element is used in the ciphertext. Bos's scheme is scale-invariant without modulus switching, and the ciphertext expansion is eliminated in each multiplicative homomorphic operation. A more practical variant of the FHE scheme [20] is implemented by Dowlin et al. [22].

In 2013, Kocabas et al. [23] used FHE for accessing, analyzing, and displaying the patient's health data, which will not leak any privacy of the health data. Next, a medical cloud computing system [24] is constructed based on the FHE scheme, which could protect user privacy and support some homomorphic computations, for example the average of the heart rate. Recently, based on the ref. [24], Kocabas et al. [7] depicted a general architecture of the medical cyber physical system, which consists of four layers, includes data acquisition, data aggregation, etc. He chose the Paillier scheme [25] and the RLWE-based BGV scheme [19] to calculate the encrypted health data. The Paillier scheme is used for the computation of the average heart rate, and the RLWE-based BGV scheme is used for the long QT syndrome detection. However, the Paillier scheme does not support multiplicative homomorphic operation. As a result, the encrypted accumulated sum and the encrypted number of samples will be returned to the user. These ciphertexts should be decrypted to calculate the average heart rate, which incurs high computation and communication cost on the user and server.

In this paper, we endeavor to study health data security in the mobile healthcare network, which is the core component of smart cities, and focus on implementing typical medical computations securely. Taking into account the privacy and computation of the health data stored on the untrusted cloud, we adopt FHE as the main encryption primitive to carry out our research. Eventually, we make following main contributions:

- We define an architecture of the mobile healthcare network, and three typical secure medical computations, which include the average heart rate, the long QT syndrome detection, and the chi-square tests.
- We use Dowlin's FHE scheme to securely compute the average heart rate, in which only one ciphertext will be returned to the receiver. We also apply a l-bits comparator to the implementation of the long QT syndrome detection, which results in only l XOR operations and one homomorphic multiplication are needed. The chi-square tests are first implemented by homomorphic additions and homomorphic multiplications that can be used to study whether varicose veins is relevant to overweight.
- We implement the proposed secure medical computations on the personal computer, and demonstrate the efficiency of our scheme according to the thorough comparison analysis with the state-of-the-art schemes.

The rest of this paper is organized as follows. Our mobile healthcare network is presented in Section 2. Preliminaries are introduced in Section 3. In Section 4, we show the adopted FHE scheme and serval secure healthcare computations. Secure computations of the average heart rate, the long QT syndrome detection, and the chi-square tests are implemented in Section 5. The detailed simulation results and efficiency analysis are shown in Section 6. Finally, Section 7 concludes the whole paper.

2 Mobile healthcare network

The architecture of our mobile healthcare network [7, 24] is described as in Fig. 1, which consists of four sections: the wearable device section, the preprocessing section, the cloud server section, and the physician diagnosis section. Detailed operations and security analysis of each section are introduced as follows.

2.1 The wearable device section

Health data usually consists of body temperature, heart rate, etc. With the rapid development of sensors and chips, more and more health data can be easily acquired by different kinds of wearable devices, such as smart rings, smart watches and so on. These devices work in low-power computation, communication, and storage modules. Thus, they can only collect the health data. Alternatively, the health data is transmitted to the adjacent preprocessing section.

2.2 The preprocessing section

Wearable devices would transmit the collected data to a cloudlet in order to preprocess the data. A cloudlet is designed to have more powerful computational ability than the wearable device, for example, a cellphone, a router. These wearable devices can be connected to a cloudlet by the Bluetooth or the ZigBee protocol. These wearable devices and the cloudlet can form a kind of Internet of things. The router is the most important proportion to construct an Internet of things, because it can select and set the route according to the situation of the channel automatically, which will enable the weak device to have the strong facility of sending the preprocessed data to the cloud server.

2.3 The cloud server section

The cloud server provides two essential functionalities: storage and computation. To guarantee user privacy, the health data can be encrypted by the traditional public key encryption scheme, then stored on the cloud server. However, the encrypted health data cannot be computed without decryption. The only feasible method for computing on the ciphertext is the FHE scheme. In order to predict the patient's healthcare condition, the encrypted health data should be analyzed by homomorphic operations on the cloud server, such as the average heart rate, the long QT syndrome detection. The ciphertext of the analysis will be returned to the doctor.

2.4 The physician diagnosis section

The doctor could obtain the decryption result by his secret key. He makes an accurate diagnosis for the remote healthcare diagnosis, which belongs to the passive action. An active action indicates that the decryption result can be turned to the activation of an actuator, for example a robotic arm can be used for the robot-assisted surgery. It can be noticed that the medical data is always in encrypted format

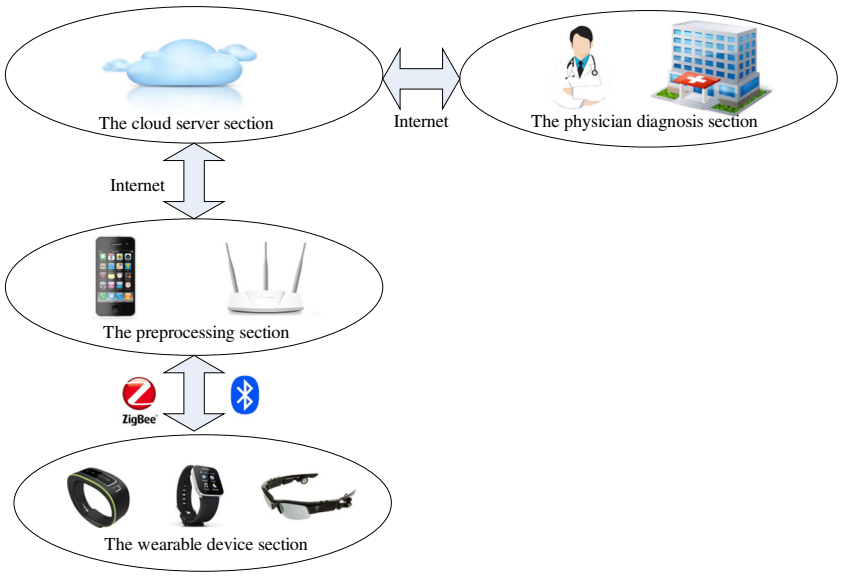

Fig. 1 The model of our mobile healthcare network

until it is decrypted by the doctor, which will leak nothing about user privacy.

3 Preliminaries

3.1 Basic notation

For any real number z, let $\lceil z \rceil$, $\lfloor z \rfloor$ and $\lfloor z \rceil$ denote the rounding of an up, down or the nearest integer respectively. Namely, $\lceil z \rceil \in [z, z+1)$, $\lfloor z \rfloor \in (z-1, z]$ and $\lfloor z \rceil \in (z-1/2, z+1/2]$.

Given n-dimensional vectors $\vec{a} = (a_0, a_1, \cdots, a_{n-1})$ and $\vec{b} = (b_0, b_1, \cdots, b_{n-1})$, the inner product $< \vec{a}, \vec{b} >$ is defined as $< \vec{a}, \vec{b} >= \vec{a} \cdot \vec{b}$. Let the prime modulus $q \geq 2$, $l = \lceil \log_T(q) \rceil$, T is a power of 2 and $N' = m \cdot l$. Let $BitDecomp(\vec{a}) = (a_{0,0}, a_{0,1}, \cdots, a_{0,l-1}, \cdots, a_{n-1,0}, a_{n-1,1}, \cdots, a_{n-1,l-1})$, where $a_{i,j}$ is the jth component of a_i and ordered from the least significant bit to the most significant bit. Let $Powersof2(\vec{a}) = (a_0, 2a_0, \cdots, 2^{l-1}a_0, \cdots, a_{n-1}, 2a_{n-1}, \cdots, 2^{l-1}a_{n-1})$, and output a N'-dimensional vector.

Let $f(x) = x^n + 1 \in \mathbb{Z}[x]$, where the security parameter $n = 2^k$, $k \in \mathbb{Z}^+$. Let $R = \mathbb{Z}/ < f(x) >$ be the ring modulo $f(x)$. Let $R_q = \mathbb{Z}_q[x]/ < f(x) >$ be the ring modulo $f(x)$ and the prime modulus q, where $q \geq 2$. For any $\beta > 0$, denote $D_\beta(x) = (1/\beta) \cdot exp(-\pi(x/\beta)^2)$ to be the density function of the Gaussian distribution over the real domain. For any $q \geq 2$, the distribution $\overline{\psi_\beta}(q)$ on $\mathbb{Z}_q$ obtained by drawing $y \leftarrow D_\beta$ and output $\lfloor q \cdot y + 1/2 \rfloor$. Let χ denote an error distribution whose coefficients are randomly chosen from $\overline{\psi_\beta}(q)$.

3.2 Learning with errors over rings

The assumption of learning with errors over rings was first proposed by Lyubashevsky et al. [14].

Definition 1 (RLWE) The $RLWE_{n,q,\chi}$ problem is to distinguish following two distributions: (1) $(a, b = a \times b + e) \in R_q \times R_q$, where a and s are generated uniform randomly from R_q, and e is an error randomly selected from χ. (2) $(a, c) \in Unif(R_q \times R_q)$, where $Unif$ represents uniformly random.

Lyubashevsky et al. [14] has proved that the RLWE assumption is hard over ideal lattices (see Theorem 1), and $(a, b = a \times s + e)$ is pseudorandom.

Theorem 1 ([14]) *Given the security parameter $n = 2^k$, where $k \in \mathbb{Z}^+$, the prime modulus $q \equiv 1 mod\, n$. There is an efficient distribution χ outputting a ring element $r \in R$ overwhelmingly with maximum length B. Hence, if there exists an efficient algorithm for $RLWE_{n,q,\chi}$, there is a quantum algorithm to solve the approximate SVP problem* [26] *over the ring R on ideal lattices under the worst case.*

Given a lattice $\wedge$ and a positive number $s > 0$, the discrete Gaussian distribution over $\wedge$ and s is denoted as $D_{\wedge,s}$, which assigns a probability proportional to $exp(-\pi \|x\|^2/s^2)$ for each $s \in \wedge$. The discrete Gaussian $D_{\mathbb{Z}^n,s}$ is just the product distribution of n independent copies of $D_{\mathbb{Z},s}$, where $\wedge = \mathbb{Z}^n$.

3.3 Fully homomorphic encryption

Definition 2 (Homomorphic Encryption [8]) A homomorphic encryption scheme HE consists of four probabilistic polynomial algorithms, which is described as follows.

- $HE.KeyGen(1^\lambda)$: Take as input a security parameter λ. Output the public key pk and the secret key sk, and note $(pk, sk) \leftarrow HE.KeyGen(1^\lambda)$.
- $HE.Enc(pk, m)$: Take as input the public key pk and the plaintext m. Output the ciphertext c, and note $c \leftarrow HE.Enc(pk, m)$.
- $HE.Dec(sk, c)$: Take as input the secret key sk and the ciphertext c. Output the decryption result m', and note $m' \leftarrow HE.Dec(sk, c)$.
- $HE.Eval(pk, f, c_1, c_2, \cdots, c_l)$:Take as input the public key pk, function f and ciphertexts c_i, where $i = 1, 2, \cdots, l$, and c_i corresponds to the plaintext m_i. Output the ciphertext c_f, and note $c_f \leftarrow HE.Eval(pk, f, c_1, c_2, \cdots, c_l)$, namely $HE.Dec(sk, c_f) = f(m_1, m_2, \cdots, m_l)$, where f is an operation circuit over the plaintext space.

Definition 3 (Permitted Circuit [8]) Let HE be a homomorphic encryption scheme, and f be a function with l variables. For any key pair (pk, sk), any plaintext $M = (m_1, m_2, \cdots, m_l)$ and any ciphertext $C = (c_1, c_2, \cdots, c_l)$, if $Dec(sk, Eval(pk, f, c_1, c_2, \cdots, c_l)) = f(m_1, m_2, \cdots, m_l)$, then f is the permitted circuit of HE, and the corresponding circuit C_{HE} is called the permitted circuit.

Definition 4 (Augmented Decryption Circuit [9]) The augmented decryption circuit of the homomorphic encryption scheme HE is obtained by connecting two decryption circuits with the addition circuit or the multiplication circuit, called D_{HE}.

Definition 5 (Fully Homomorphic Encryption [19]) Let HE be a homomorphic encryption scheme, if the decryption circuit of HE and the set of the augmented decryption

circuit D_{HE} are in the permitted circuit set C_{HE}, then HE is a FHE scheme.

3.4 Dowlin's FHE scheme

An efficient fully homomorphic encryption scheme $FHE = (Setup, KeyGen, Encrypt, Decrypt, Add, Multi)$ proposed by Dowlin [22] is described as follows:

- $FHE.Setup(1^n)$: Input the security parameter $n = 2^k$, where $k \in \mathbb{Z}^+$. Then choose a sufficiently large prime modulus q, where $q \bmod n \equiv 1$. The modulus t satisfies the condition that $1 < t < q$, where t is a small plaintext modulus. Coefficients of distributions χ_{key} and χ_{err} are randomly chosen from $\overline{\psi_\beta}$. Let $params = (n, q, t, \chi_{key}, \chi_{err})$.
- $FHE.KeyGen(params)$: Input the parameter $params$, choose f' and g from the key distribution χ_{key} randomly, namely $f', g \leftarrow \chi_{key}$. Set $f = [1 + tf']_q$, generate a new f if it has not the inverse element f^{-1}. Generate vectors $\vec{e}, \vec{s} \in R^l$ randomly, and each component is chosen from χ_{err}. Set $\gamma = [Powersof2(f) + \vec{e} + h\vec{s}]_q$, where $h = [tgf^{-1}]_q$. Output the public key $pk = h$, the secret key $sk = f$ and the evaluation key $evk = \gamma$.
- $FHE.Encrypt(pk, m \in R_t)$: To encrypt a plaintext m, generate the ciphertext as follows:

  $$c = [\lfloor q/t \rfloor m + e_2 + hs_2]_q,$$

  where error terms e_2 and s_2 are randomly chosen from χ_{err}.
- $FHE.Decrypt(sk, c)$: Given the ciphertext c and the secret key sk, the plaintext m can be recovered as follows:

  $$m = [\lfloor t/q \cdot [fc]_q \rfloor]_t.$$

- $FHE.Add(c_1, c_2)$: Given two ciphertexts c_1 and c_2, output the fresh ciphertext $c_{add} = [c_1 + c_2]_q$.
- $FHE.Multi(c_1, c_2)$: Given two ciphertexts c_1 and c_2, firstly compute $c_{temp} = [\lfloor t/q(c_1 \cdot c_2) \rceil]_q$, then switch c_{temp}'s secret key as follows

  $$c_{mul} = [< BitDecomp(c_{temp}), evk >]_q.$$

  We can also use Gentry's idea of raising the modulus to switch the secret key. The modulus reduction technique can be applied to reduce the noise.

4 Secure medical computation using fully homomorphic encryption

Wearable devices, such as smart wristwatch, smart bracelet, or smart glass, could be widely used to provide healthcare service. The patient's heart rate, long QT syndrome detection, blood pressure, or blood sugar [27] can be obtained by different kinds of wearable devices. To protect the patient's health data from leaking, fully homomorphic encryption is used to encrypt the data, and it allows computation on the encrypted data without decryption. In this paper, we choose efficient Dowlin's scheme [22] for the computation of the encrypted health data.

Electrocardiogram (ECG) is one of most common clinical detection methods for diagnosing myocardial ischemia and myocardial infarction. And signal of weight is another important factor which could reflect the patient's body appearance. When the patient's signals of ECG and weight are collected by wearable devices, they will be encrypted and stored on the cloud server by Dowlin's scheme [22].

In our system, secure computations including the average heart rate, the long QT syndrome detection, and the chi-square tests are defined. Statistics of the patient's average heart rate can be calculated by homomorphic addition and homomorphic multiplication from the patient's signals of ECG. The secure average computation is shown in Section 5.1.

ECG signals can be used to detect the long QT syndrome. The purpose of the long QT syndrome detection is to observe whether QT_{result} exceeds a clinical threshold, where QT_{result} represents a patient's heartbeats. And 500 is a usual clinical threshold. If $QT_{result} > 500$, the patient can be considered to have the long QT syndrome. We can use the formula $QT/\sqrt{RR}$ [28] to calculate QT_{result} from intervals of QT and RR in an ECG signal. Intervals of QT and RR are represented in Fig. 2. The secure comparison computation is shown in Section 5.2.

The chi-square tests are one of most common hypothesis testing methods, which can be used to study whether varicose veins is relevant to overweight. The detailed calculation process is shown as follows. Formulas $\chi^2 = \sum^{\nu} \frac{(A-T)^2}{T}$ and $\chi^2 = \frac{(ad-bc)^2 n}{(a+b)(c+d)(a+c)(b+d)}$ can be used for

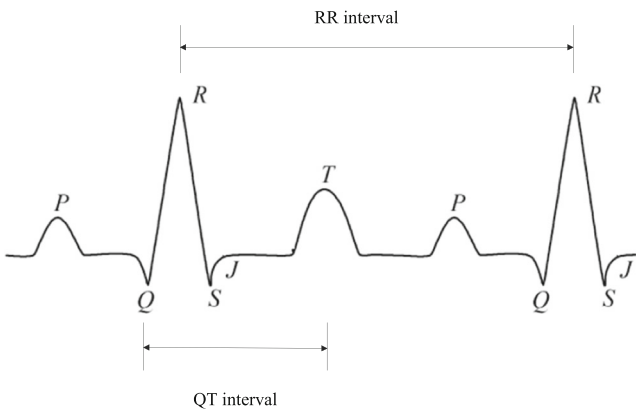

Fig. 2 Intervals of QT and RR in an ECG signal

Table 1 The basic form of the fourfold table

Frequency	Group	Qualified number	Unqualified number	Total
Actual frequency	Experimental group	a	b	$a+b$
	Control group	c	d	$c+d$
	Total	$a+c$	$b+d$	$n=a+b+c+d$
Theoretical frequency	Experimental group	T_{11}	T_{12}	$T_{11}+T_{12}$
	Control group	T_{21}	T_{22}	$T_{21}+T_{22}$
	Total	$T_{11}+T_{21}$	$T_{12}+T_{22}$	$n=T_{11}+T_{12}+T_{21}+T_{22}$

the chi-square tests of the data as described in Table 1, which consists of varicose veins and overweight without correction when $n \geq 40$, $T \geq 5$, where the degree of freedom $v = (\textit{the number of rows} - 1)(\textit{the number of columns} - 1)$, A denotes the actual frequency, T denotes the theoretical frequency. Set $T_{11} = \frac{(a+b)(a+c)}{n}$, $T_{21} = \frac{(c+d)(a+c)}{n}$, $T_{12} = \frac{(a+b)(b+d)}{n}$ and $T_{22} = \frac{(c+d)(b+d)}{n}$. Formulas $\chi_c^2 = \sum^{v} \frac{(A-T-0.5)^2}{T}$ and $\chi_c^2 = \frac{(|ad-bc|-\frac{n}{2})^2 n}{(a+b)(c+d)(a+c)(b+d)}$ can be used for updating when $n \geq 40$, $1 \leq T < 5$. From above equations, it can be noticed that the calculation of χ^2 can be converted to homomorphic additions and homomorphic multiplications. The secure computation of the chi-square tests is shown in Section 5.3.

5 The computation details

Because Dowlin's scheme [22] is fully homomorphic and efficient, we could use it to calculate the average heart rate, the long QT syndrome detection, and the chi-square tests that whether varicose veins is relevant to the patient's overweight.

5.1 Computation of the average heart rate

Reference [7] uses Paillier scheme to compute the average heart rate. However, it is additive homomorphic rather than multiplicative homomorphic, so two ciphertexts of the accumulated sum and the number of ECG samples will be returned to the receiver. Then the receiver needs to decrypt two ciphertexts and calculate their ratio as the average heart rate. In our scheme, based on the Dowlin's scheme, the average heart rate can be calculated by using homomorphic addition and homomorphic multiplication. And then, the cloud server sends the encryption of the average heart rate to the receiver. So, the receiver could obtain the final result through decryption only once.

Table 2 The implementation of the average heart rate

input: Ciphertexts $C_1, C_2, \cdots, C_n$ and $C_{n+1} \leftarrow FHE.Enc(pk, 1/n)$;
output: The fresh ciphertext C_{fresh};
$X = (X_1, X_2, \cdots, X_l)$;
$Y = (Y_1, Y_2, \cdots, Y_l)$;
$Z_1 = X_1 + Y_1$;
$T_1 = X_1 \cdot Y_1$;
$S = C_1$;
for $i = 2$ to n do
$S = add(S, C_i)$;
The algorithm $add(X, Y)$ can be defined as follows:
function $add(X, Y)$ {
for $i = 2$ to l do
{ $Z_i = X_i + Y_i + T_{i-1}$;
$T_i = X_i \cdot Y_i + (X_i + Y_i) \cdot T_{i-1}$; }
return $(Z_i)_{i=1,2,\cdots,l}$ as the sum of ciphertexts X and Y; }
$C_{fresh} = S \cdot C_{n+1}$;

The implementation details of the average heart rate are described in Table 2. Given n encrypted heart rates $C_1, C_2, \cdots, C_n$ and the encryption of the floating number $1/n$. $1/n$ keeps four bits of precision after the binary point. To get the encrypted average heart rate C_{fresh}, firstly we calculate the sum S of encrypted heart rates homomorphically, then multiply S by the ciphertext of $1/n$. The summation processing is achieved through the iteration function $add(X, Y)$, which is used to calculate the sum of X and Y. It works as follows. Defined $X = (X_1, X_2, \cdots, X_l)$ and $Y = (Y_1, Y_2, \cdots, Y_l)$. Suppose their sum is $(Z_1, Z_2, \cdots, Z_l)$. Initial values $Z_1 = X_1 + Y_1$ and $T_1 = X_1 \cdot Y_1$. For $i = 2, 3, \cdots, l$, compute $Z_i = X_i + Y_i + T_{i-1}$ and $T_i = X_i \cdot Y_i + (X_i + Y_i) \cdot T_{i-1}$. Finally, return Z_i as the output of $add(X, Y)$, where $i = 1, 2, \cdots, l$.

5.2 Computation of the long QT syndrome detection

Whether $QT/\sqrt{RR} > threshold$ is required for the long QT syndrome detection, and $threshold = 500$ is a usual clinical threshold. In order to implement the comparison conveniently, we also convert the formula $QT/\sqrt{RR} > 500$ to $QT^2 > 500^2 \cdot RR$. Firstly, we introduce a l-bits comparator [29] to implement the comparison homomorphically. Given two l-bits plaintexts $X = (X_0, X_1, \cdots, X_{l-1})$ and $Y = (Y_0, Y_1, \cdots, Y_{l-1})$ represent QT^2 and $500^2 \cdot RR$ respectively, we can compare X and Y from the most significant bit, where X_i and Y_i are ith components of X and Y, X_{l-1} and Y_{l-1} are the most significant bits of X and Y respectively. If $X_{l-1} > Y_{l-1}$ or $X_{l-1} < Y_{l-1}$, it means that

Table 3 The implementation of the long QT syndrome detection

input: Ciphertexts C_X=(C_{X_0},C_{X_1},$\cdots$,$C_{X_{l-1}}$),C_Y=(C_{Y_0},C_{Y_1},$\cdots$,$C_{Y_{l-1}}$);
output: The result of the long QT syndrome detection;
for $i = l - 1$ to 0 do
{ $C_{S_i} = C_{X_i} + C_{Y_i} - 2(C_{X_i} \times C_{Y_i})$;
if $S_i == 1$
{
$C_{M_i} = C_{X_i} \cdot (C_{X_i} - C_{Y_i})$;
if $M_i == 1$
{
It means that $X_i > Y_i$, hence $X > Y$;
break;
}
else
{
It means that $X_i < Y_i$, hence $X < Y$;
break;
}
} }

$X > Y$ or $X < Y$. Otherwise, we compare next bits X_{l-2} and Y_{l-2}, and so on.

Given ciphertexts C_X and C_Y corresponding to plaintexts X and Y respectively, where $C_X = (C_{X_0}, C_{X_1}, \cdots, C_{X_{l-1}})$, $C_Y = (C_{Y_0}, C_{Y_1}, \cdots, C_{Y_{l-1}})$, we use the above method to compare C_X and C_Y. However, C_{X_i} and C_{Y_i} cannot be compared directly. Hence, we first calculate $C_{S_i} = C_{X_i} \bigoplus C_{Y_i} = C_{X_i} + C_{Y_i} - 2(C_{X_i} \times C_{Y_i})$, if $S_i = 1$, it means that $X_i \neq Y_i$. Then we should compute $C_{M_i} = C_{X_i} \cdot (C_{X_i} - C_{Y_i})$, if $M_i = 1$, it means that $X_i > Y_i$, otherwise $X_i < Y_i$. The detailed comparison process is shown in Table 3.

The homomorphic comparison in the ref. [7] needs $l - 1$ XOR operations and $2l$ homomorphic multiplications, our method only needs at most l XOR operations and one homomorphic multiplication.

5.3 Computation of the chi-square tests

The calculation of the chi-square tests is mainly to research whether varicose veins is relevant to overweight homomorphically. Suppose we observe 146 pairs of brothers, and in any pair one is obese and the other one is normal body mass. The incidence of varicose veins is described in Table 4, where C_* represents the ciphertext of $*$, which is encrypted by Dowlin's scheme. For example, C_8 represents 8 pairs of brothers suffer varicose veins, including the obese ones and the normal ones. Then we can calculate the encrypted data homomorphically.

The concrete analysis process is as follows. Firstly, we establish tests hypotheses, namely H_0 and H_1, where H_0 represents varicose veins relevant to overweight, H_1 is the opposite of H_0. The inspection level is determined by parameters $\alpha = 0.05$, $\nu = 1$. For convenience, we use Dowlin's scheme [22] to encrypt floating numbers 1/146, 1/18, 1/128, 1/40, 1/106, which keeps four bits of precision after the binary point. As the value of χ^2 can be computed by $C_{\chi^2} = (C_8 \cdot C_{96} - C_{10} \cdot C_{32})^2 \cdot C_{146} \cdot C_{1/146} \cdot C_{1/18} \cdot C_{1/128} \cdot C_{1/106}$, namely $\chi^2 \approx 2.9996$. Next, we compute $C_{T_{11}}$, $C_{T_{21}}$, $C_{T_{12}}$, and $C_{T_{22}}$ as above. The cloud server sends $C_{T_{11}}$, $C_{T_{12}}$, $C_{T_{21}}$, and $C_{T_{22}}$ to the receiver, and these ciphertexts can be decrypted by the receiver's secret key. Because $1 < T_{11} < 5$, we use the formula $C_{\chi_c^2} = (|C_8 \cdot C_{96} - C_{10} \cdot C_{32}| - C_{73})^2 \cdot C_{146} \cdot C_{1/18} \cdot C_{1/128} \cdot C_{1/40} \cdot C_{1/106}$ for updating. The doctor can get $\chi_c^2 \approx 2.1017$ by his secret key, and the updating value $P = 0.1471$ can be obtained by the function $CHIDIST$ in the microsoft office excel or the chi-square boundary table. Because $P = 0.1471 > \alpha$, we should accept H_0 and reject H_1.

Table 4 The incidence of varicose veins in 146 pairs of brothers

		Obese		
		Happen	Not happen	Total
Normal body mass	Happen	C_8	C_{10}	C_{18}
	Not happen	C_{32}	C_{96}	C_{128}
	Total	C_{40}	C_{106}	C_{146}

6 Simulation and analysis

The average heart rate, the long QT syndrome detection, and the chi-square tests are carried out on the same experimental environment. It is described as follows: the operating system is microsoft windows 7, featuring two Intel (R) Core (TM) i5-3470 CPU processors, running at 3.20 GHz, with 8.00 GB RAM, and the virtual machine is allocated 4.00 GB internal storage, single processor with the operating system Ubuntu 12.04. Our implementation uses the simple encrypted arithmetic library [22], which can be used for bioinformatic, genomic, or other researches. For conveniently, we set the security parameter ranging from 200 to 400, and each test has five iterations, and datum shown in the following tables are averages of them.

Table 5 Implementation time of the average heart rate and the chi-square tests in our scheme (unit: millisecond)

Security parameter	200	250	300	350	400
The average heart rate	80.24	106.94	147.03	183.61	237.87
The chi-square tests	469.37	650.55	918.95	1244.81	1601.81

Table 6 Implementation time of the long QT syndrome detection in Kocabas's scheme [7] and our scheme (unit: millisecond)

Security parameter	200	250	300	350	400
Kocabas's scheme [7]	2059.9	2252.57	2381.59	2395.01	2425.67
Our scheme	635.42	907.986	1254.71	1721.49	2241.55

Tables 5 shows the implementation time of the average heart rate and the chi-square tests. Because the average heart rate in Kocabas's scheme [7] is implemented by Paillier scheme, which is somewhat homomorphic. The chi-square tests are first implemented by homomorphic additions and homomorphic multiplications. So, we do not compare the implementation time of the average heart rate and the chi-square tests with other schemes. As seen from the Table 6, our scheme's implementation times of the long QT syndrome detection decreases a lot compared to Kocabas's scheme [7] under the same security parameter. Datum in Table 6 are described with Fig. 3, which shows two scheme's efficiency of the long QT syndrome detection. Obviously, Fig. 3 shows our scheme's implementation time of the long QT syndrome detection is much better than that of Kocabas's scheme [7] with the increasing of the security parameter.

From the above analysis, it can be known that two different encryption schemes will be used for calculations of the average heart rate and the long QT syndrome detection in Kocabas's scheme [7], which may block some homomorphic operations between heart rate and interval of RR or QT. Fortunately, this disadvantage does not exist in our scheme.

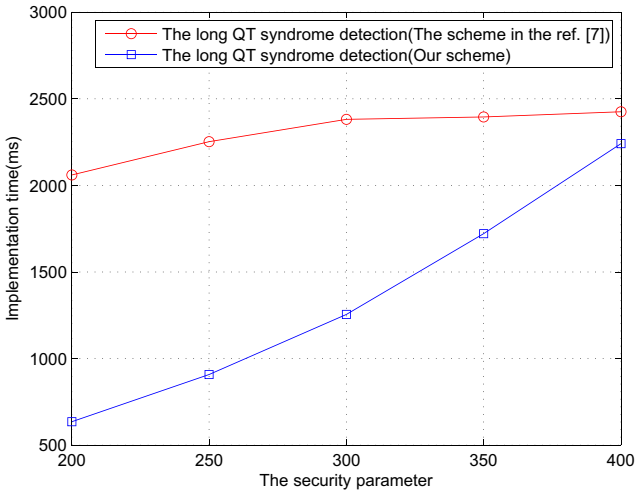

Fig. 3 Efficiency comparison of the long QT syndrome between Kocabas's scheme [7] and our scheme

7 Conclusion

In this paper, we propose a four-layer mobile healthcare network consisting of the wearable device section, the preprocessing section, the cloud server section, and the physician diagnosis section. Then, we define three secure medical computations: the average heart rate, the long QT syndrome detection, and the chi-square tests. Because Dowlin's scheme is more efficient than the BGV scheme with a small plaintext moduli, we use it to implement three secure medical computations. In our implementation of the average heart rate, only one ciphertext will be returned to the receiver, compared with two ciphertexts of the accumulated sum and the number of samples sent to the receiver in Kocabas's scheme [7]. To improve the efficiency of the computation of the long QT syndrome, we adopt a more efficient l-bits comparator which only needs l XOR operations and one homomorphic multiplication. However, l-bits comparator needs $l-1$ XOR operations and $2l$ homomorphic multiplications in Kocabas's scheme [7]. And simulation results show that the computation of the long QT syndrome is more efficient than that of Kocabas's. We first implement the chi-square tests homomorphically to study whether varicose veins is relevant to overweight.

Acknowledgments This work was supported by the National Natural Science Foundation of China (61171072,61602316), the Science and Technology Innovation Projects of Shenzhen (ZDSYS20140430164957660, JCYJ20140418095735596, and JCYJ20160307150216309).

References

1. Hollands RG (2008) Will the real smart city please stand up. City 12(3):303–320
2. Lee J, Lee H (2014) Developing and validating a citizen-centric typology for smart city services. Gov Inf Q 31:93–105
3. Zhang K, Yang K, Liang X et al. (2015) Security and privacy for mobile healthcare networks: from a quality of protection perspective. IEEE Wirel Commun 22(4):104–112
4. Yamamoto S, Yamazaki S, Shimizu T et al (2016) Body temperature at the emergency department as a predictor of mortality in patients with bacterial infection. Medicine 95(21):e3628
5. Kiyono K, Struzik ZR, Aoyagi N, Yamamoto Y (2006) Multiscale probability density function analysis: non-gaussian and scale-invariant fluctuations of healthy human heart rate. IEEE Trans Biomed Eng 53(1):95–102
6. Baumert M, Baier V, Truebner S, Schirdewan A, Voss A (2005) Short-and long-term joint symbolic dynamics of heart rate and blood pressure in dilated cardiomyopathy. IEEE Trans Biomed Eng 52(12):2112–2115
7. Kocabas O, Soyata T, Aktas MK (2016) Emerging security mechanisms for medical cyber physical systems. IEEE/ACM Trans Comput Biol Bioinf 13(3):401–416
8. Gentry C (2009) A fully homomorphic encryption scheme. Ph. D. thesis, Stanford University
9. Van Dijk M, Gentry C, Halevi S, Vaikuntanathan V (2010) Fully homomorphic encryption over the integers. Advances in cryptology-EUROCRYPT 2010. Springer, pp 24–43

10. Brakerski Z, Vaikuntanathan V (2014) Efficient fully homomorphic encryption from (standard) LWE. SIAM J Comput 43(2):831–871
11. Brakerski Z, Vaikuntanathan V (2011) Efficient fully homomorphic encryption from (standard) LWE. In: 2011 IEEE 52nd annual symposium on foundations of computer science. IEEE, pp 97–106
12. Brakerski Z (2012) Fully homomorphic encryption without modulus switching from classical GapSVP. Advances in Cryptology-CRYPTO 2012. Springer, pp 868–886
13. Regev O (2005) On lattices, learning with errors, random linear codes, and cryptography. In: Proceedings of the 37th annual ACM symposium on theory of computing. ACM, pp 84–93
14. Lyubashevsky V, Peikert C, Regev O (2013) On ideal lattices and learning with errors over rings. J ACM 60(6):1–23
15. Zhang X, Xu C, Jin C, Xie R, Zhao J (2014) Efficient fully homomorphic encryption from RLWE with an extension to a threshold encryption scheme. Futur Gener Comput Syst 36:180–186
16. Brakerski Z, Vaikuntanathan V (2011) Fully homomorphic encryption from ring- LWE and security for key dependent messages. Advances in Cryptology- CRYPTO 2011. Springer, pp 505–524
17. Chen H, Hu Y, Lian Z (2015) Double batch for RLWE-based leveled fully homomorphic encryption. Chin J Electron 24 (3):661–666
18. Ducas L, Micciancio D (2015) FHEW: bootstrapping homomorphic encryption in less than a second. In: Annual international conference on the theory and applications of cryptographic techniques. Springer, pp 617–640
19. Brakerski Z, Gentry C, Vaikuntanathan V (2012) (Leveled) fully homomorphic encryption without bootstrapping. In: Proceedings of the 3rd innovations in theoretical computer science conference. ACM, pp 309–325
20. Bos JW, Lauter K, Loftus J, Naehrig M (2013) Improved security for a ring-based fully homomorphic encryption scheme. Cryptography and Coding. Springer, pp 45–64
21. Stehlé D, Steinfeld R (2011) Making NTRU as secure as worst-case problems over ideal lattices. In: International conference on the theory and applications of cryptographic techniques. Springer, pp 27–47
22. Dowlin N, Gilad-Bachrach R, Laine K, Lauter K, Naehrig M, Wernsing J (2015) Manual for using homomorphic encryption for bioinformatics. Technical report MSR-TR-2015-87, Microsoft Research
23. Kocabas O, Soyata T, Couderc J-P, Aktas M, Xia J, Huang M (2013) Assessment of cloud-based health monitoring using homomorphic encryption. In: 2013 IEEE 31st international conference on computer design (ICCD). IEEE, pp 443–446
24. Kocabas O, Soyata T (2015) Utilizing homomorphic encryption to implement secure and private medical cloud computing. In: 2015 IEEE 8th international conference on cloud computing. IEEE, pp 540–547
25. Paillier P (1999) Public-key cryptosystems based on composite degree residuosity classes. Advances in cryptology-EUROCRYPT 1999, Springer, pp 223–238
26. Gentry C, Peikert C, Vaikuntanathan V (2008) Trapdoors for hard lattices and new cryptographic constructions. In: Proceedings of the 40th annual ACM symposium on theory of computing. ACM, pp 197–206
27. Mohanty S, Asfour I, Mohanty P, Trivedi C, Gianni C, Gokoglan Y, Bai R, Burkhardt J, Horton R, Sanchez J et al (2016) Baseline fasting blood sugar predicts long-term outcome of catheter ablation in atrial fibrillation. J Am Coll Cardiol 67(13_S):797–797
28. Bazett HC (1997) An analysis of the time-relations of electrocardiograms. Ann Noninvasive Electrocardiol 2(2):177–194
29. Elmehdwi Y, Samanthula BK, Jiang W (2014) Secure k-nearest neighbor query over encrypted data in outsourced environments. In: 2014 IEEE 30th international conference on data engineering (ICDE), pp 664–675