A New Cryptography Algorithm to Protect Cloud-based Healthcare Services

Mohammed Aledhari*, Ali Marhoon†, Ali Al-Qaabi‡, and Fahad Saeed*
*Department of Computer Science
Western Michigan University, Kalamazoo, MI, 49008-5466,
Correspondence should be addressed to Fahad Saeed at fahad.saeed@wmich.edu
† Electrical Engineering Department
University of Basrah, Basrah, Iraq
‡ Information and Communication Engineering Department
University of Baghdad, Baghdad, Iraq

Abstract—**The revolution of smart devices has a significant and positive impact on the lives of many people, especially in regard to elements of healthcare. In part, this revolution is attributed to technological advances that enable individuals to wear and use medical devices to monitor their health activities, but remotely. Also, these smart, wearable medical devices assist health care providers in monitoring their patients remotely, thereby enabling physicians to respond quickly in the event of emergencies. An ancillary advantage is that health care costs will be reduced, another benefit that, when paired with prompt medical treatment, indicates significant advances in the contemporary management of health care. However, the competition among manufacturers of these medical devices creates a complexity of small and smart wearable devices such as ECG and EMG. This complexity results in other issues such as patient security, privacy, confidentiality, and identity theft. In this paper, we discuss the design and implementation of a hybrid real-time cryptography algorithm to secure lightweight wearable medical devices. The proposed system is based on an emerging innovative technology between the genomic encryptions and the deterministic chaos method to provide a quick and secure cryptography algorithm for real-time health monitoring that permits for threats to patient confidentiality to be addressed. The proposed algorithm also considers the limitations of memory and size of the wearable health devices. The experimental results and the encryption analysis indicate that the proposed algorithm provides a high level of security for the remote health monitoring system.**

I. Introduction

The security of personal health data is a critical issue when these data are transferred on wireless channels to end users such as doctors, nurses, family members, or other authorized individuals. The data are exchanged wirelessly where cables cannot be used, so all nodes can receive data if they are within range. If the network is not secure, an adversary could read, modify, and inject messages into the network. Such incorrect information, even when not for nefarious reasons, can lead to serious consequences for patients and potentially compromise their safe treatment. Current data encryption methods are not suitable for the cloud-based services such as remote healthcare monitoring due to using heterogeneous devices that use a variety of transfer protocols that belong to different vendors. Observing these facts, we take advantage of the nature of the genomic encryption and the deterministic Chaos Theory to implement a more efficient cryptography algorithm to secure remote healthcare monitoring. When connecting the **W**ireless **S**ensor **N**etwork (WSN) to the Internet, the location of these sensor nodes would not be an important issue for intruders wherein they can attack the WSN from anywhere. Consequently, a powerful security mechanism should be designed with awareness of the resource constraints of the WSN. Most of the security protocols designed for WSN cannot be applied directly in the **W**earable **W**ireless **B**ody **A**rea **N**etwork (WWBASN), since these nodes have limited resources in the power, computation processing, and communication. A powerful method of data encryption is the one-time-pad algorithm [1], where each single piece of data is encrypted individually with a unique key. The disadvantage of this method is that it requires a vast number of keys; a **P**seudo **R**andom **N**umber **G**enerator (PRNG) could be used to generate the required keys, but it is problematic in terms of the key repetition. To eliminate the problem of repetition, the application of Chaos Theory to generate these keys represents a solid approach. The security algorithm of these networks must maintain the resource limitations of the sensor nodes and at the same time provide high security. The new generation of security mechanisms is genomics encryption due to its randomness and complexity. The security algorithm of these networks must maintain the resource limitations of the sensor nodes and at the same time provide high security. As a method of authentication, [2] used a wavelet domain Hidden Markov Model. In addition, information from the **E**lectro**C**ardio**G**raphy (ECG) signal is used as a biometric key. [3] had proposed a **P**hysiological **S**ignal-based **K**ey **A**greement (PSKA) which shares the cryptographic key using physiological signals obtained from the patient. [4] had proposed a simulation of a chaotic block cipher used for wireless sensor networks through which results were compared with RC5 and RC6 block ciphers. An improvement of a **M**essage **A**uthentication **C**ode (MAC) algorithm was proposed using chaos and XOR encryption [5]. The advanced MAC generator has been divided into a sub key generator and the MAC structure. Genomic-based cryptography emerged as a new cryptographic field, in which nucleotide is used as an information carrier, and modern biological technology is used as an implementation tool. [6] had solved **H**amiltonian **P**ath **P**roblems (HPP) by using nucleotide computing, with its inherent advantages, such as vast parallelism and extraordinary information density. [7] had designed a genomics and chaos map to encrypt images. [8] had introduced a new approach to the genomics computing algorithm aims to perform genomics computing with adaptive parameters by using **Q**uantum-behaved **P**article **S**warm **O**ptimization (QPSO). [9] had designed an RGB image encryption algorithm using DNA encoding and a chaos map. In 2015, [10] had designed a DNA-based encryption and decryption algorithm such that it overcame the limitation of the genomics-based cryptography algorithm and incorporated modular arithmetic cryptography at some steps. Also, in 2015 [11] had designed a system that realizes complex access control on encrypted data, where the attributes used to describe a user's credentials will determine a policy as to which recipient will be able to decrypt the provided cipher text. [12] developed a secure authentication and authorization for patient security and privacy medical data for the Internet of Things (IoT)-based healthcare monitoring system. [13] proposed an improvement to the **U**ser **A**uthentication and **K**ey **A**greement **S**cheme (UAKAS) for **H**eterogeneous WSN (HWSN) by improving the security level and enabling the HWSN to dynamically grow, without influencing any part involved in the UAKAS.

The purpose of this paper is to design and implement a hybrid cryptography algorithm to protect cloud-based health services. Moreover, it will introduce a generic concept that can be used by other cloud-based applications to secure data that exchange remotely. The remainder of this paper is organized as follows: Section II presents preliminaries of this work for both: genomic encryption method and the Chaos Theory. Section III discusses the overall architecture of the proposed algorithm and model description. Section IV presents the experiments and results of the proposed cryptography algorithm. Section V discusses the comparative analysis and Section VI presents our conclusions.

II. Preliminaries

A. Genomic-based Cryptography

Deoxyribo Nucleic Acid (DNA) is a biochemical macromolecule that contains genetic information necessary for the functioning of living beings. The genomics include the entire hereditary information about the organism cell and consists of the chromosomes in a cell's nucleus and components of the genome. A genomic molecule consists of a double-stranded nucleotides structure that is obtained by two twisted single-stranded DNA chains, hydrogen bonded together between bases (A-T and G-C) [14]. The double-helix structure is configured by two single, antiparallel strands.

Four kinds of bases are found in two strands: **Adenine (A)**; **Guanine (G)**; **Thymine (T)**; and **Cytosine (C)**. A strand contains a sequence of bases in a specific pattern. The other strand contains the complementary nucleotides of the first strand. The adenine pairs with a thymine by using a double bond (A = T), while the thymine and the cytosine pair with each other by using a triple bond (G = C). The genomic sequencing information is contained in the nucleotide bases.

Genomic-based cryptography emerged as a new cryptographic field in 1994 and has been used as an information carrier and as a modern implementation tool in biological technology. The computational process using genomic-based cryptography produces a sequence of nucleotides: A, T, C, and G, as the encrypted data output. The genomic-based cryptography is done by hybridization of the DNA molecules and is formed by a double helix structure of complementary base pairs to encode data. The DNA addition and subtraction operation rules, described in Tables I and II on page 2, are used to confuse the ECG data values, such that A= 00; T= 01; C= 10; G= 11.

TABLE I: Subtraction operation for the DNA sequence

-	A	T	C	G
A	A	G	C	T
T	T	A	G	C
C	C	T	A	G
G	G	C	T	A

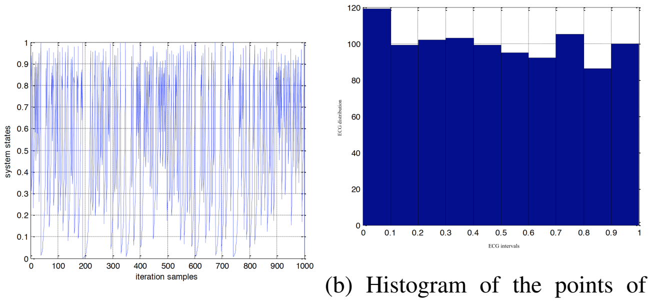
TABLE II: Addition operation for the DNA sequence

+	A	T	C	G
A	A	T	C	G
T	T	C	G	A
C	C	G	A	T
G	G	A	T	C

B. Deterministic Chaos Theory

Chaos functions have mainly been used to develop mathematical models of nonlinear systems. They have attracted the attention of many mathematicians owing to their extremely sensitive nature of initial conditions, as well as their enormous applicability to modeling complex problems of daily life. The sequences produced by such functions have very good randomness and complexity. These functions have an extreme sensitivity to initial conditions. For example, if the initial start value of a chaotic function is modified 10^{-20}, iterative numbers produced after some iterations appear to differ from each other. This extreme sensitivity to initial conditions and some other interesting properties, such as pseudo-randomness, wide spectrum, and good correlation indicate that chaotic functions may serve as a promising alternative to conventional cryptographic algorithms.

The main advantage using Chaos Theory lies in the observation that a chaotic signal looks like noise to unauthorized users. Moreover, generating chaotic values is often low cost with simple iterations, which makes it suitable for the construction of stream ciphers. Therefore,



(a) The chaotic orbit for parameters of $x_0 = 0.6, \mu = 0.8$

(b) Histogram of the points of typical orbit with 1000 points length

Fig. 1: Locations of ECG Leads on the Human Body

cryptosystems can provide a secure and fast means of data encryption, which is crucial for data transmission in many applications. Generally speaking, chaotic stream ciphers use chaotic systems to generate pseudorandom key streams to encrypt the data (one-by-one). In this work, a 1-D tent logistic map is employed for key generation. The map generates chaotic sequences in the interval [0, 1], assuming the following formula [15]:

$$x_{n+1} = \begin{cases} \frac{x_n}{\mu} & 0 \le x_n \le \mu \\ \frac{1-x_n}{1-\mu} & \mu < x_n \le 1 \end{cases}, \tag{1}$$

where x_n refers to the state variable of the system that belongs to the interval [0, 1], $\mu \in$ (0,1], and $x_0 \in$ [0,1] is the initial condition. A typical orbit with initial condition $x_0 = 0.6$ and control parameter $\mu = 0.8$ is shown in Figure 1 (a) on page 3. The distribution of the points of the orbit with a length of 1000 points is shown in Figure 1 (b) on page 3. The histogram plot is a graphical representation similar to the bar chart where it shows the distribution of data with ranges of the data grouped into intervals.

C. One-Time Pad Encryption Method

The one-time pad encryption mechanism is simple: encrypt each sample of the data by the addition modular, which gets a bit or character from a random key generator. For example, the key sequence generated by a random generator is as follows:

$$pad = k_i = k_1, k_2, k_3, ..., k_n, \quad k_i \in [0, 1]. \tag{2}$$

The original message which will be encrypted by the pad keys is as follows:

$$message = m_i = m_1, m_2, m_3, ..., m_n, \quad m_i \in [0, 1]. \tag{3}$$

Then the cipher is as follows:

$$c_i = m_i \oplus k_i. \tag{4}$$

To decrypt the cipher in the receiver side, the following function is used:

$$m_i = (m_i \oplus k_i) \oplus k_i. \tag{5}$$

III. Proposed Encryption Algorithm

This section provides step-by-step details of the proposed algorithm for the data encryption by combining a DNA-based encryption technique and chaotic logistic maps. Limitation of the wearable sensor node memories causes each sample of the ECG signal to be encrypted alone. The sample data is a 16-bit length (the most 4-bit are considered to be zeros when the 12-bit analog to digital converter is used). Each sample is divided and encoded into 8 DNA bases, with each consisting of a two-bit length. A chaotic sequence is generated by using a tent chaotic map with the initial condition xo1 and constant parameter μ_{01}. The sequence is scaled to [0, 65536] and converted into 8 DNA bases, with the most 4-bit equating to zeros. The 8 new DNA bases are generated by combining the encoded ECG sample and encoded chaotic random key, as shown in Tables I and II on page 2.

Two of 1-D tent chaotic maps with different initial conditions x_{02} and x_{03} and constant parameters μ_{02} and μ_{03} are used to construct the 2-D encoding matrix, which will be used as a complement (or not) to the combination of the ECG signal and the first chaotic map. After that, we will obtain the six encryption keys: x_{01}, x_{02}, x_{03}, μ_{01}, μ_{02}, and μ_{03}. The computational precision of the 64-bit double precision number is 2^{52}, according to the IEEE standard for the floating-point arithmetic(IEEE 754). Therefore, the total number of different keys used is $(2^{52})^6 = 2^{312}$. Such a large key space is efficient and sufficient for reliable practical use. The proposed algorithm of the ECG sensory data was initially introduced by the work of [1] and [7], where they used it for image encryption. A modification of this algorithm has been made to be applicable for WWBASN, such that each sample of the ECG data is encrypted alone, since it works on limited resources in terms of memory and computation. The proposed algorithm works in 10 simple steps as follows:

Step 1:

Convert the signal samples into a binary sequence as a $n * m$ binary matrix (m = 16 bits).

Step 2:

Encode the binary sequence into a matrix of nucleotides (DNA sequence matrix) to get the encoding matrix $\frac{n*m}{2}$ such that: A = 00, T = 01, C = 10, G = 11.

Let's assume the signal data vector

$$s = \begin{pmatrix} 2055 \\ 1250 \\ 3590 \end{pmatrix}$$

Then the binary sequence would be:

$$s = \begin{pmatrix} 0000100000000111 \\ 0000010011100010 \\ 0000111000000110 \end{pmatrix}$$

So, we get the DNA sequence as follows:

$$s = \begin{pmatrix} AACAAATG \\ AATAGCAC \\ AAGCAATC \end{pmatrix}$$

Step 3:

Divide the DNA sequence matrix into 8 sub-matrices each $\frac{n*m}{8}$ as follows:

DNA sub-matrix$_1$ is $s_1 = \begin{pmatrix} A \\ A \\ A \end{pmatrix}$, DNA sub-matrix$_2$ is $s_2 = \begin{pmatrix} A \\ A \\ A \end{pmatrix}$

DNA sub-matrix$_3$ is $s_3 = \begin{pmatrix} C \\ T \\ G \end{pmatrix}$, DNA sub-matrix$_4$ is $s_4 = \begin{pmatrix} A \\ A \\ C \end{pmatrix}$

DNA sub-matrix$_5$ is $s_5 = \begin{pmatrix} A \\ G \\ A \end{pmatrix}$, DNA sub-matrix$_6$ is $s_6 = \begin{pmatrix} A \\ C \\ A \end{pmatrix}$

DNA sub-matrix$_7$ is $s_7 = \begin{pmatrix} T \\ A \\ T \end{pmatrix}$, DNA sub-matrix$_8$ is $s_8 = \begin{pmatrix} G \\ C \\ C \end{pmatrix}$

Step 4:
Generate a chaotic sequence vector n1 through a 1-D chaotic map with initial condition x_{01} and constant parameter μ_{01}. Scale the chaotic sequence to [0, 65536].

Step 5:
Apply steps 1 to 3 to the chaotic sequence.

Step 6:
Add the DNA sub matrices of the original signal to the DNA sub matrices of the chaotic sequence, according to the rules shown in Tables I and II on page 2.

Step 7:
Recombine the sub matrices generated from the **Step 6** to form a new binary sequence matrix $c(n * m)$.

Step 8:
Generate two chaotic sequences, $c_1(n * 1)$ and $c_2(1 * m)$, along with another initial condition x_{02}, x_{03}, and constant parameters μ_{02} and μ_{03}. After that, multiply the two vectors to produce a matrix $w(n * m)$ with range [0, 1]. Map the value of w into (0, 1) by mod $(w, 1)$. Then use the following threshold function $f(x)$ to get the binary sequence matrix:

$$f(x) = \begin{cases} 0, & 0 < w(i, j) \leq 0.5 \\ 1, & 0.5 < w(i, j) \leq 1 \end{cases}, \tag{6}$$

Step 9:
Get the complement to the matrix $w(i, j) = 1$, then $c(i, j)$ is complemented. Otherwise, it is unchanged. For example: if the first row of the
w is $\big(1001100110111001\big)$,
c is $\big(1010100100101010\big)$,
c' is $\big(0011000010010011\big)$
This would produce a new encoding matrix that is $c'(n * m)$. Rescaling this matrix produces the encrypted ECG sample, which would be transmitted through the wireless channels.

Step 10:
Apply the inverse process of **Step 2** and **Step 1** for the sequence matrix c', then we will get a real value of the matrix **D** that represents the encrypted data signal.

In the decryption process, the reverse steps are applied from the **Step 10** to the **Step 1**. Also, use the subtraction operation instead of the addition operation as shown in Tables I and II on page 2.

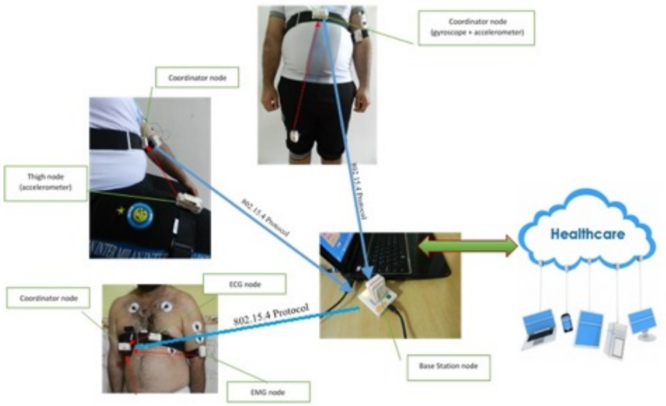
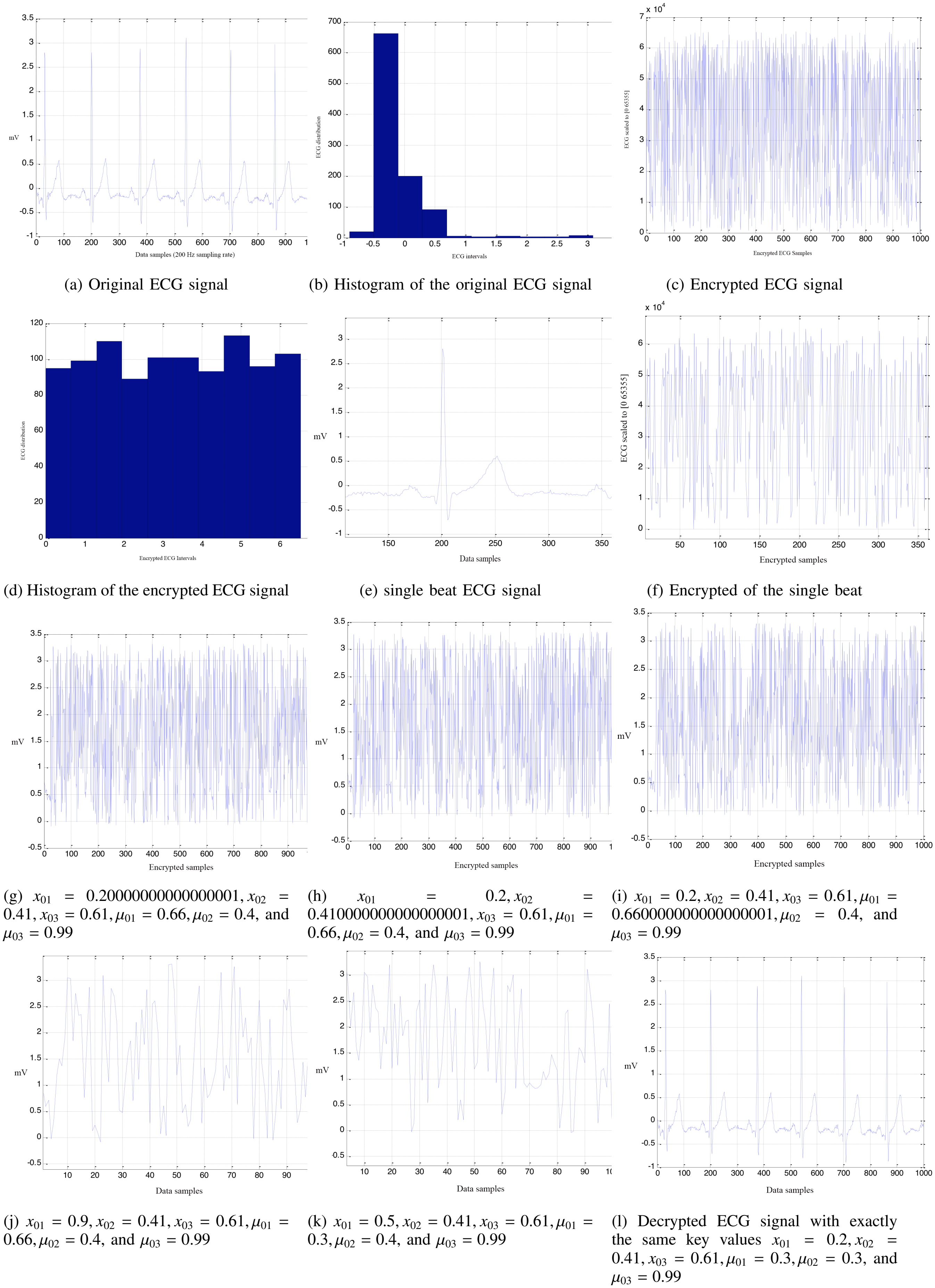


Fig. 2: Architecture of the Propose System

IV. Experiments and Results

In this work, we used the SHIMMER platform [16] as the embedded sensor system. From the hardware viewpoint, this platform includes the following:
A low-power 16-bit microcontroller (Texas Instrument MSP430F1611), a low-power radio supported with 802.15.4, an extension module for the ECG, **E**lectro**M**yo**G**raphy (EMG), and **G**alvanic **S**kin **R**esponse (GSR), and built-in triaxial accelerometer.
The MSP430 microcontroller runs at 8 MHz, has 10 KB of RAM, 48 KB of flash memory, and includes a fast hardware multiplier. As a practical implementation, five sensor nodes were used as a body area network, which includes the ECG, EMG, and accelerometers in the human chest, thigh, and leg. The chest node was used as a coordinator for the body area network, while the encryption algorithm was implemented only inside the ECG node. The nodes were programmed using the TinyOs, and the data were transmitted to the base station node using Python 2.7 under a Linux operating system to decrypt the data. There is a significant problem that effects the performance of the decryption process by reducing the transmitted packets in the time unit when using a collision-free (MAC) protocol. The proposed algorithm was implemented and tested using a SHIMMER sensor nodes platform, a python programming language, and a TinyOs 2.1.2 that supports the floating-point computation. The proposed system is designed to support multiple patients (up to 256), using a single base-station node. Also, it can support different groups of patients: each group belongs to a different base-station node. In this mode of the operation, different scenarios could be considered, such as intensive care unit, hospital rooms, rehabilitation units, or patient homes. The proposed system has been designed to be secure, scalable, effective, and easy to use, as shown in Figure 2 on page 4. The sensory data packet from different types of sensor nodes, which are attached to the patient's skin, will send information to the coordinator node, using the IEEE 802.15.4 protocol. The coordinator node will forward the packets to the base-station node, using the same protocol. The original



(a) Original ECG signal

(b) Histogram of the original ECG signal

(c) Encrypted ECG signal

(d) Histogram of the encrypted ECG signal

(e) single beat ECG signal

(f) Encrypted of the single beat

(g) $x_{01} = 0.20000000000000001, x_{02} = 0.41, x_{03} = 0.61, \mu_{01} = 0.66, \mu_{02} = 0.4$, and $\mu_{03} = 0.99$

(h) $x_{01} = 0.2, x_{02} = 0.410000000000000001, x_{03} = 0.61, \mu_{01} = 0.66, \mu_{02} = 0.4$, and $\mu_{03} = 0.99$

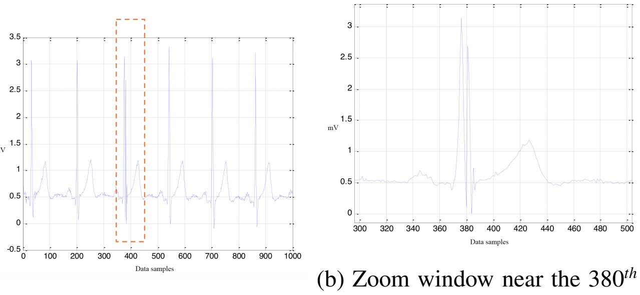
(i) $x_{01} = 0.2, x_{02} = 0.41, x_{03} = 0.61, \mu_{01} = 0.660000000000000001, \mu_{02} = 0.4$, and $\mu_{03} = 0.99$

(j) $x_{01} = 0.9, x_{02} = 0.41, x_{03} = 0.61, \mu_{01} = 0.66, \mu_{02} = 0.4$, and $\mu_{03} = 0.99$

(k) $x_{01} = 0.5, x_{02} = 0.41, x_{03} = 0.61, \mu_{01} = 0.3, \mu_{02} = 0.4$, and $\mu_{03} = 0.99$

(l) Decrypted ECG signal with exactly the same key values $x_{01} = 0.2, x_{02} = 0.41, x_{03} = 0.61, \mu_{01} = 0.3, \mu_{02} = 0.3$, and $\mu_{03} = 0.99$

Fig. 3: Different decrypted ECG signal with different keys values



(a) ECG signal with some lost samples

(b) Zoom window near the 380^{th} sample shows the effect of samples loss

Fig. 4: Locations of ECG Leads on the Human Body

ECG sensed signal is shown in Figure 3 (a) on page 5, while the histogram of the original signal is shown in Figure 3 (b) on page 5, where most of the ECG data are between -0.5 and 0.5 mV. The Figure 3 (c) on page 5 is shown the encrypted ECG signal according to the proposed encryption algorithm, where the signal is detected as a noise due to the nature of the chaos function. The key values used were $x_{01} = 0.2, x_{02} = 0.41, x_{03} = 0.61, \mu_{01} = 0.66, \mu_{02} = 0.4$, and $\mu_{03} = 0.99$. Figure 3 (d) on page 5 is depicts the histogram of the encrypted ECG signal. Here, the contribution of the encrypted data is uniform in contrast to the Figure 3 (b) on page 5. The single beat ECG signal is illustrates in Figure 3 (e) on page 5, while Figure 3 (f) on page 5 displays a zoom window of the decrypted signal. Figure 3 (g, h, and i) on page 5 show different changes in key values that demonstrate the power of the proposed algorithm. This is because of the sensitivity of the chaos function to any changes in the initial conditions. The decrypted signals that use the same encryption keys are shown in Figure 3 (j) on page 5, while Figure 3 (k and l) on page 5 shows a sample window of the encrypted ECG signal after changing the initial condition x_{01}. These figures show the difference of the initial values of the decrypted signals. These values are as follows : $\frac{x_{01}}{\mu_{01}}$ and use the equation 1.

Figure 4 (a) on page 6 reveilles the real-time encryption process when sending the ECG sensor node packets to the body area network. Figure 4 (b) on page 6 depicts a zoom around the 380^{th} samples, where the packets were lost. The decryption process assumes these samples (20 successive samples) are zeros. The whole decryption process was not affected by the lost samples since the encryption process decrypts each sample alone. The fluctuation shown in the loss. occurs due to the chaotic nature of the decryption process. The correlation coefficients are important features, and they are calculated based on the correlation between the encrypted and the original signals. The main points that can be obtained from the correlation coefficients are listed as follows:

1) When it is close to the value of **1**, then there is a positive linear relationship between the two vectors.
2) When it is close to the value of **-1**, then there is a negative linear relationship between the two vectors.
3) When it is close to the value of **0**, then there is no linear relationship between the two vectors.

The following equations are used to determine the correlation coefficient [17]:

$$E(x) = \frac{1}{N}\sum_{i=1}^{N} x_i, \tag{7}$$

$$D(x) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E(x))^2, \tag{8}$$

$$cov(x) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E(x))(y_i - E(y)), \tag{9}$$

$$r_{xy} = \frac{cov(x,y)}{\sqrt{D(x)}\sqrt{D(y)}}. \tag{10}$$

The correlation coefficient between the decrypted ECG signal and the original ECG signal was 1, and 0.0215 between the original ECG and the encrypted signals.

V. Comparative Analysis

Since wearable wireless sensor networks have limited storage and computational resources, most of the encryption algorithms are not feasible on these platforms. Connecting this network to the Internet makes the system more vulnerable, since it would be easy for an intruder to gain access to the patients' data, especially in case the physical distance would not be a problem. Hence, a powerful encryption algorithm is required to maintain the privacy of patient data with the limited resources of such a network.

There are many effective crypto-systems of traditional encryption algorithms used for information security, such as **D**ata **E**ncryption **S**tandard (DES); **T**riple DES (3DES); Blow fish; and **A**dvanced **E**ncryption **S**tandard (AES). DES suffers from the key size that is (**56**-bit), and it uses a **64**-block ciphering. There are some potential issues that can occur, especially when encrypting several gigabytes of data using the same key. The use of the 3DES enables the reuse of the DES implementation by cascading three instances of DES (with distinct keys). This algorithm is secure up to **2186** key spaces, but it is slow. Blow fish is a symmetric block cipher that uses a variable length of a key between **32** to **448** bits. It uses key-dependent lookup tables. Hence, performance depends on the resources that are available with the platform used. AES accepts keys of **128**; **192** and **256**-bit length, and uses **128**-bit blocks. It is an efficient algorithm from the software and hardware perspectives [18]. Table III on page 7 summarizes different encryption algorithms in terms of key length, block ciphering, number of keys, and applicability in a wireless sensor network. A real-time DNA-based encryption algorithm and Chaos Theory for secure healthcare information, has been proposed for the ECG signal encryption. The proposed algorithm has been designed to be a context-aware algorithm, where all computations are performed on the sensor node. The algorithm was designed to work with limited resources of the computational sensor nodes, where other encryption methods cannot be implemented with these resources.

The use of Chaos Theory, as a key generator, is more powerful than the pseudorandom generator. The DNA-based encryption is a solid approach that can be used for data encoding, since it requires a minimum of computations. Furthermore, the nature of the one-time pad to encrypt each sample individually makes it an appropriate technique to be used in a wireless sensor network, where it minimizes the required memory space. Finally, the algorithm has been implemented successfully in a real-time body area network for healthcare monitoring and is suitable to be used in the presence of a collision in wireless communication. For future work, it is important to develop an algorithm to predict the sample loss and to design a collision-free MAC (media access control) protocol.

TABLE III: Encryption Algorithm Comparisons

Algorithm	Key space (bit)	Number of keys (bit)	Block cipher	Applicable in WSN
DES	56	1	yes	no
3DES	168	3	yes	no
Blow fish	32-448	huge	yes	no
AES	128, 192, 256	1	yes	no
Proposed	312	each sample has it's a unique key	no	yes

VI. Conclusion

Smart connected wearable medical devices have gained the attention of both patients and professional healthcare providers due to several characteristics, such as the fact that they are lightweight, small in size, easy to use, and inexpensive. Also, the development of communication protocols and techniques, such as cloud-based services, enables many healthcare providers to deliver their services to patients remotely. However, the remote-based and cloud-based services have security issues, due to connecting heterogeneous devices that come from different vendors. Use of traditional cryptography algorithms does not protect health data over the network because the data are transferred through multiple devices and protocols. In this paper, we designed and implemented a novel encryption algorithm that relies on the utilization of genomics encryption and deterministic chaos to protect the remote healthcare monitoring system. The proposed algorithm protects the health data from the main threats, such as a key theft, man-in-the-middle attack, and brute force attack. Also, the proposed algorithm is designed in a way that considers the limitations of device size, memory capacity, power consumption, and cost. Moreover, the proposed algorithm does not require complex computations to encrypt the data. The practical implementation of the proposed encryption algorithm and result analysis prove that it is ideally suitable for remote health monitoring services in terms of data security, computations, and power consumption.

Acknowledgment

We would like to express our special thanks and deepest gratitude to **Dr. Marianne Di Pierro**, for her excellent guidance, caring, patience, and providing us with an excellent atmosphere for doing the research.

References

[1] M. Babaei, "A novel text and image encryption method based on chaos theory and dna computing," *Natural computing*, vol. 12, no. 1, pp. 101–107, 2013.

[2] H. Wang, H. Fang, L. Xing, and M. Chen, "An integrated biometric-based security framework using wavelet-domain hmm in wireless body area networks (wban)," in *Communications (ICC), 2011 IEEE International Conference on*. IEEE, 2011, pp. 1–5.

[3] K. K. Venkatasubramanian, A. Banerjee, and S. K. S. Gupta, "Pska: Usable and secure key agreement scheme for body area networks," *IEEE Transactions on Information Technology in Biomedicine*, vol. 14, no. 1, pp. 60–68, 2010.

[4] Y. Liu, S. Tian, W. Hu, and C. Xing, "Design and statistical analysis of a new chaotic block cipher for wireless sensor networks," *Communications in Nonlinear Science and Numerical Simulation*, vol. 17, no. 8, pp. 3267–3278, 2012.

[5] B. En-Jian, Z. Jun-Jie, and W. Liang-Cheng, "Wsn message authentication code based on chaos and xor-encryption," *Sensors &amp; Transducers*, vol. 156, no. 9, p. 161, 2013.

[6] L. M. Adleman, "Molecular computation of solutions to combinatorial problems," *Nature*, vol. 369, p. 40, 1994.

[7] Q. Zhang, L. Guo, and X. Wei, "Image encryption using dna addition combining with chaotic maps," *Mathematical and Computer Modelling*, vol. 52, no. 11, pp. 2028–2035, 2010.

[8] M. Karakose and U. Cigdem, "Qpso-based adaptive dna computing algorithm," *The Scientific World Journal*, vol. 2013, 2013.

[9] M. A. Mokhtar, S. N. Gobran, and E.-S. A. El-Badawy, "Colored image encryption algorithm using dna code and chaos theory," in *Computer and Communication Engineering (ICCCE), 2014 International Conference on*. IEEE, 2014, pp. 12–15.

[10] N. H. UbaidurRahman, C. Balamurugan, and R. Mariappan, "A novel dna computing based encryption and decryption algorithm," *Procedia Computer Science*, vol. 46, pp. 463–475, 2015.

[11] C. Gritti, W. Susilo, T. Plantard, and K. T. Win, "Privacy-preserving encryption scheme using dna parentage test," *Theoretical Computer Science*, vol. 580, pp. 1–13, 2015.

[12] S. R. Moosavi, T. N. Gia, A.-M. Rahmani, E. Nigussie, S. Virtanen, J. Isoaho, and H. Tenhunen, "Sea: a secure and efficient authentication and authorization architecture for iot-based healthcare using smart gateways," *Procedia Computer Science*, vol. 52, pp. 452–459, 2015.

[13] M. S. Farash, M. Turkanović, S. Kumari, and M. Hölbl, "An efficient user authentication and key agreement scheme for heterogeneous wireless sensor network tailored for the internet of things environment," *Ad Hoc Networks*, vol. 36, pp. 152–176, 2016.

[14] M. E. BORDA, O. TORNEA, and T. HODOROGEA, "secret writing by dna hybridization," *Acta Technica Napocensis-Electronica-Telecomunicatii (Electronics and Telecommunications)*, vol. 1, no. 50, pp. 21–24, 2009.

[15] R. Ye and W. Guo, "A chaos-based image encryption scheme using multi modal skew tent maps," *Journal of Emerging Trends in Computing and Information Sciences*, vol. 4, no. 10, pp. 800–810, 2013.

[16] A. Burns, B. R. Greene, M. J. McGrath, T. J. O'Shea, B. Kuris, S. M. Ayer, F. Stroiescu, and V. Cionca, "Shimmer$^{\text{TM}}$–a wireless sensor platform for noninvasive biomedical research," *IEEE Sensors Journal*, vol. 10, no. 9, pp. 1527–1534, 2010.

[17] J. D. Gibbons and S. Chakraborti, *Nonparametric statistical inference*. Springer, 2011.

[18] J. Thakur and N. Kumar, "Des, aes and blowfish: Symmetric key cryptography algorithms simulation based performance analysis," *International journal of emerging technology and advanced engineering*, vol. 1, no. 2, pp. 6–12, 2011.