SECURING WIRELESS IMPLANTABLE MEDICAL DEVICES USING ELECTROCARDIOGRAM SIGNALS

by

Guanglou Zheng

Dissertation submitted in fulfilment of the requirements

for the degree of

DOCTOR OF PHILOSOPHY

Department of Computing
Faculty of Science and Engineering
Macquarie University
Sydney, Australia

November 2016

ABSTRACT

Implantable Medical Devices (IMDs), such as pacemakers and cardiac defibrillators, can perform a variety of health monitoring and therapeutic functions. A wireless module has become an intrinsic part of many modern IMDs for parameter configuration and medical data transmission. However, such wireless modules can be manipulated to compromise a patient's safety or privacy by eavesdropping or by sending unauthorized commands. A unique challenge in this scenario is that doctors who are not pre-authorized need to have access to the IMDs in an emergency situation.

In this thesis, we study the use of electrocardiogram (ECG) signals for securing the IMDs. Blood circulation system in the body is regarded as an inborn secure channel to transmit ECG signals to the IMD and to its external programmer simultaneously. Measurements extracted from the ECG signal, e.g., inter-pulse intervals (IPIs) and random binary sequences (BSes), are used for security purposes. In an emergency situation, doctors can gain access to the IMDs by measuring the patient's real-time ECG signal. However, adversaries cannot achieve this as long as they do not have any physical contact with the patient.

We design two ECG-based key distribution schemes based on a fuzzy commitment primitive and a fuzzy vault primitive, respectively. Using the schemes, doctors can obtain a symmetric key by measuring the patient's real-time ECG signal. We also compare these two schemes from different perspectives and discuss their advantages and disadvantages.

In order to provide information-theoretically unbreakable encryption for the IMDs, we design an ECG based Data Encryption (EDE) scheme. This scheme

combines two well-known techniques of classic One-Time Pads (OTPs) and error correcting codes. Meanwhile, in order to improve the efficiency of the BS generation, we develop an ECG Multiple Fiducial-points based Binary Sequence Generation (MFBSG) algorithm. Existing methods solely rely on ECG IPIs to produce BSes and hence introduce unacceptable levels of latency. On the other hand, the proposed algorithm uses five distinct feature values from one heartbeat cycle. By doing this, the time required to generate a BS is reduced, and we achieve the key design goal of low-latency.

In conclusion, this thesis explores the use of ECG signals for securing the IMDs. The proposed ECG-based schemes can solve the unique challenge prevalent in this environment.

STATEMENT OF CANDIDATE

I certify that the work in this thesis has not previously been submitted for a degree nor has it been submitted as part of the requirements for a degree to any other university or institution other than Macquarie University.

I certify that the thesis is an original piece of research and has been written by me.

In addition, I certify that all information sources and literature used are indicated in the thesis.

. . . . . .

Guanglou Zheng

ACKNOWLEDGMENTS

There are a number of people without whom this thesis might not have been written, and to whom I am greatly indebted.

First of all, I would like to express my sincere gratitude to my supervision team who have helped, supervised and educated me in the past three and a half years. Dr. Gengfa Fang hosted me as my principle supervisor in the past 3 years, and helped me from living to studying at Macquarie University. Mehmet A. Orgun, a respectful professor, has a lot of experience in research methods, paper preparation and supervision. His sensitivity in research topics and directions of innovations has impressed me. Dr. Rajan Shankaran, a wise and approachable mentor, spent a lot of time on discussing every idea that I worked on. He went through my papers and thesis carefully, which helped me improve the quality of my writing and article organization.

My family always give me unconditional support without which I could not complete the thesis successfully. My parents have always encouraged me to study well since I was a little boy in a village. My wife, Dr. Li Qiao, has given me full support financially and mentally during my study in Sydney. She continues to inspire me to work on innovative ideas and papers. Also many thanks to my sister and niece who have accompanied my mother during this period.

Thanks to all my friends in the Department of Engineering and the Department of Computing. This research is proudly supported by the Macquarie University Research Excellence Scholarships (MQRES).

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Implantable Medical Devices (IMDs), such as artificial pacemakers, implantable cardiac defibrillators (ICDs), neuro-stimulators, and implantable drug delivery systems, use embedded computer systems to perform a variety of health monitoring and therapeutic functions automatically. IMDs have been widely used for monitoring and treating physiological conditions for patients, such as cardiac arrhythmia, diabetes and Parkinson's disease [5]. Normally an IMD is composed of bio-sensors, a miniaturized computer, and a wireless module. A typical example of the IMD is a cardiac implant, such as pacemakers and ICDs. Cardiac implants are placed into a patient's chest to monitor and correct an abnormal heart rhythm (bradycardia, tachycardia or arrhythmia). IMDs play a critical role in the patient's healthcare. Several examples of IDM applications are shown in Fig. 1.1, with the functionality of each device explained as below.

Fig. 1.1(a) shows an artificial pacemaker, with the image from [6]. It is implanted in the patient's chest, with electrodes that are in contact with the heart muscles for regulating the beating of the heart. The operation is normally done in a cardiology laboratory or an operation theater. It is used to treat patients with arrhythmias. During an arrhythmia, the heartbeat of the patient is too slow, too fast, or with an irregular rhythm.

Fig. 1.1(b) shows a Medtronic InterStim® neuro-stimulation device which is used

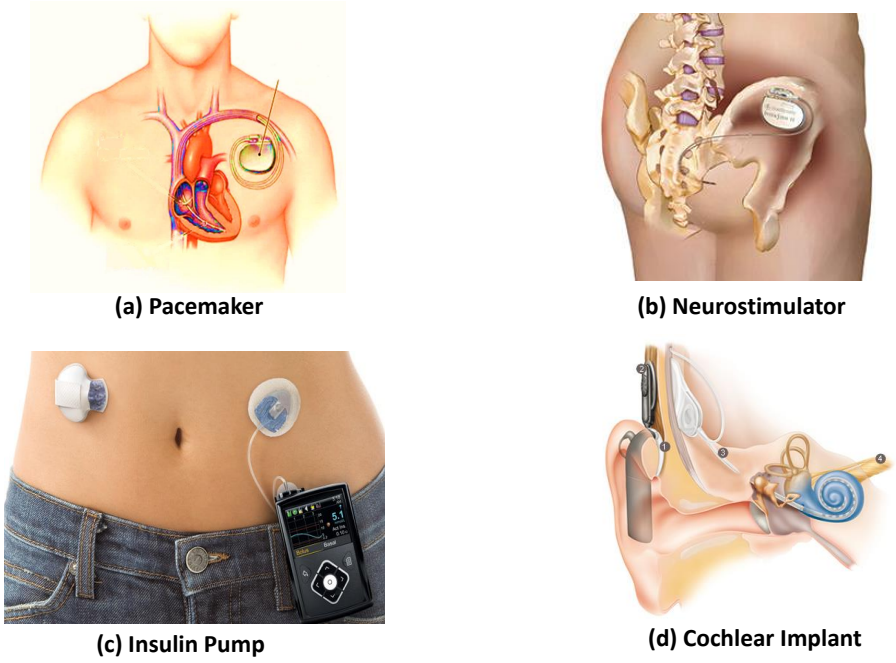


Figure 1.1: Examples of Implantable Medical Devices (IMDs).

for sacral nerve stimulation therapies [7]. This device targets the communication problem between the brain and the nerves that control the bladder. The use of this device provides the patient freedom from the embarrassment of leaks. It may also help improve the patient's sleep as the need to wake up frequently for urinating is reduced.

Fig. 1.1(c) shows a semi-implanted insulin pump. It can be easily carried on a belt, inside a pocket, or even attached to a bra, thus protecting the patient's privacy by making it virtually invisible to others [8]. It monitors the current glucose levels and then delivers precise amount of insulin continuously into the body to closely match the body's needs. It replaces the need for frequent insulin injections and offers better glucose control.

Fig. 1.1(d) is a cochlear implant which is used to help the patient suffering hearing loss. It does the work of damaged parts of the inner ear (cochlea) to provide sound signals to the brain [9]. Different from hearing aids which make sound louder, cochlear implants can help transfer the sound to hearing nerves in the brain and enable the patient to hear

better.

All these devices need to be implanted in an operation room and be initialized with proper parameters. However, if the patient's health condition is changed after the operation, doctors have to adjust the IMDs accordingly. In order to facilitate this kind of configuration, wireless communication functions have been introduced to the IMD system.

1.1 IMD Wireless Communications

Currently wireless communication capabilities have been adopted as an intrinsic part of many modern IMDs for medical purposes [10]. An external device, named a radiowave programmer, is used to configure parameters to and extract data from these IMDs wirelessly. With its wireless module, the IMD can be configured by using the programmer and monitored remotely by using a home monitor device. For instance, a pacemaker, as a typical IMD, is implanted in a patient's chest or abdomen to help control abnormal heart rhythms. After the operation, the patient needs to see their doctor or clinician regularly to undertake tests to make sure that the IMD is working properly. During the visit, the doctor can fine-tune the IMD according to patient's changed conditions. This is done by using a device programmer to communicate with the IMD through the wireless channel.

Fig. 1.2 shows a three-tier application architecture of an IMD communication system. In the figure, Tier-1 is an IMD implanted in the body for medical purposes. The figure takes a pacemaker as an example, but it could be any IMD in the body. In Tier-2, the IMD communicates wirelessly with a device programmer or a monitor for telemetry purposes. A doctor can use the programmer to fine-tune the IMD parameters wirelessly. The monitor, e.g., the Medtronic CareLink® Home Monitor [11], is used to collect the patient's medical data through the wireless channel. In Tier-3, the monitor sends patient's medical data to a remote clinician for check-up. This telemetry function gives a clinician 24/7 access to

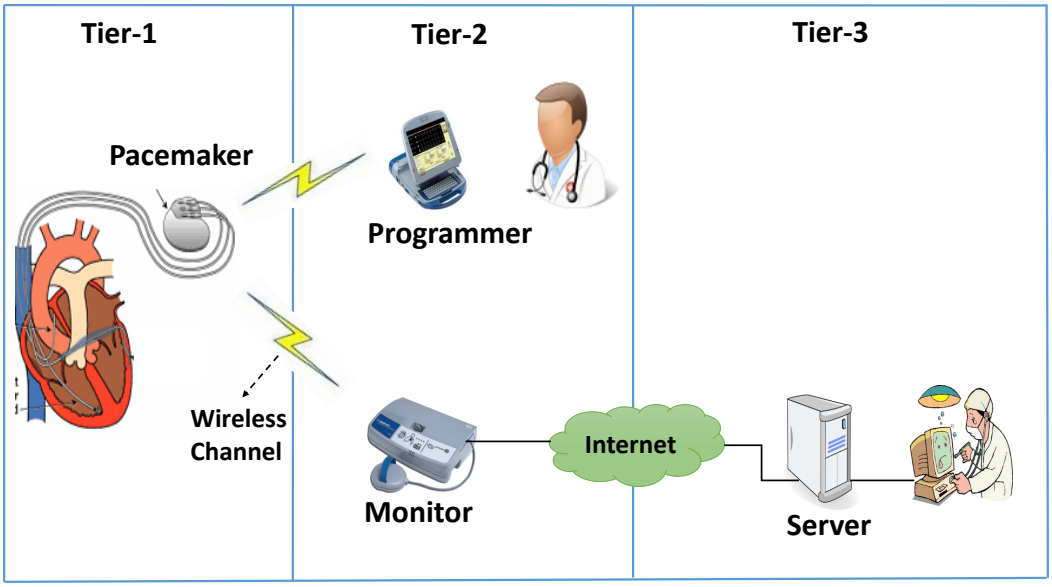


Figure 1.2: A three-tier application architecture of an IMD communication system.

patient's reports which offer information comparable to an in-office visit [11]. Therefore, the wireless module of the IMD provides patients an advanced health care service.

1.2 Research Motivation

Despite of the benefits of using a wireless module in the IMD, recent studies have shown that such wireless module can be manipulated to compromise a patient's privacy or safety by eavesdropping or by sending unauthorized commands. This is because the underlying communication links between the IMD and the programmer are insecure [12, 13, 14, 15]. The attackers can launch these attacks by using a commercial programmer or an off-the-shelf radio and computer equipment [12, 1]. They can easily obtain these attacking devices, e.g., a demonstration of attacking using proper antennas, a Universal Software Radio Peripheral (USRP), a computer, and an oscilloscope [12]. They can even buy a commercial programmer from the ebay (http://www.ebay.com.au/) directly.

Recent studies have demonstrated successful attacks on IMDs [12, 13, 14] in the laboratory. Halperin et al. [12] demonstrated that an adversary, equipped with a commercial programmer or custom equipment, can easily launch an eavesdropping attack or even an active attack on an ICD. The experiment disclosed that, since wireless communications between the ICD and the programmer are in cleartext and hence not protected cryptographically, patient's privacy could be easily breached by intercepting transmitted data, including the patient's name, date of birth, medical ID number, patient history, etc. Furthermore, active attacks can be launched to disclose ICD identification, patient data and cardiac data. These active attacks can even change therapies or induce fibrillation, causing life-threatening consequences to patients. Similarly, as an example, security attacks can be successfully launched on a commercial glucose monitoring and insulin delivery system, as demonstrated by Li et al. [13]. These attacks can compromise both the privacy and the safety of patients, e.g., disclosing the medical condition of the patient, stopping/resuming the insulin injection, or injecting the insulin with a much higher dose than required. These active attacks on both the ICD and the insulin pump can be life-threatening to the patients. Since next-generation IMDs will incorporate greater communication functions and computation capabilities, and be more networked [12], these security issues have to be addressed before a wide spread deployment of these IMDs is possible.

Different from conventional security systems, a unique challenge of providing security solutions for IMDs is that an IMD should be accessed successfully by unauthorized doctors using a programmer in an emergency situation. Meanwhile, it is still protected against adversaries in the patient's daily living [16]. To illustrate, a doctor in any hospital who is not authorized to operate an IMD beforehand may have to access the IMD. This would happen, for instance, when a patient bearing an IMD is admitted to a hospital for an emergency treatment during his/her period of travel to other cities or countries. On the other hand, the security mechanisms present in the IMD have to protect the IMD against

any unauthorized malicious access from adversaries. Therefore, it seems that a pair of contradictory requirements have to be satisfied.

Conventional security schemes that use keys or credentials cannot be directly utilized here. One approach is to obtain a secret key from a server via the Internet. However, we cannot guarantee a reliable access to the Internet anytime, anywhere, especially in developing countries. The underlying network vulnerabilities of the Internet may also result in the key being disclosed to malicious parties. Some methods propose to obtain the key from the patient. However, some patients may forget the key or be unconscious in the emergency, making these methods unreliable. The approach of using a backdoor for the emergency treatment may be exploited by malicious attackers since IMDs have been widely used by millions of patients [5].

Therefore, the IMD security design has unique challenges and is still an open issue. It requires researchers from academia and industries to propose innovative and practical solutions for the IMD devices which are already in the market or will come in the future.

1.3 Contributions of the Thesis

As analyzed in the previous Section 1.2, the IMD security design should balance requirements between security and accessibility, so that doctors without any pre-deployed information can gain access to the IMD for the purpose of an emergency treatment. It is a unique challenge as conventional security key-based solutions cannot be directly used here.

In this thesis, we propose electrocardiogram (ECG) signal-based security schemes for the IMDs. There are two kinds of devices in an IMD system, an IMD and its external device programmer. If we allow the programmer to come in contact with the patient's body, e.g., by using a wearable sensor, then the IMD and its programmer are connected by the

same medium (that is, the patient's body). Physiological signals in the medium (body) can be measured by both the IMD and its programmer, e.g., ECG and photoplethysmogram (PPG). Therefore, these physiological signals can provide common trusted information between the IMD and the programmer. In our proposed schemes, we require both the IMD and the programmer measuring real-time ECG signals. This can prevent attacks from malicious parties by using the patient's previous medical data.

In order to address issues and challenges in the ECG-based IMD security design, we make a number of contributions in this thesis. They are highlighted in the following sub-sections.

1.3.1 ECG-based Key Distribution

We propose ECG-based key distribution schemes for the IMD security by using the fuzzy commitment primitive in Chapter 3 and using the fuzzy vault primitive in Chapter 4, respectively. Using these two schemes, the security key of the IMD can be distributed from the IMD to its programmer securely in an emergency situation. Besides this, our contributions in the design of these two schemes are as follows:

- A fuzzy commitment-based key distribution scheme is proposed in Chapter 3 for the IMD security, including its communication model and performance evaluation via simulation. In order to generate two random bit strings with a high matching performance, we design an ECG bit string generation algorithm which will be executed by the IMD and the programmer, respectively.

- In Chapter 4, we explore the key distribution for the IMD by using a different fuzzy vault primitive. In this scheme, points on a polynomial curve is sent into the wireless channel, and a large number of chaff points are added to hide these real polynomial points. However, in our scheme, we propose to calculate hash values

of X coordinates but not add chaff points, which can help to reduce memory and communication overhead of the IMD.

- To provide guidance to researchers, we conduct a comparative analysis of these two schemes in order to identify their similarities and differences, and contrast their relative merits and demerits. The comparison is done in terms of ECG measurements, error correcting codes, key concealing and revealing processes, and their communication and computation overhead.

1.3.2 Encryption with Modified One-Time Pads

We design an ECG-based data encryption scheme in Chapter 5 which uses modified one-time pads for the IMD security. Unlike schemes proposed in Chapter 3 & 4, security keys in our scheme are generated from ECG signals and are used to encrypt secret data directly. Compared to conventional symmetric key-based encryption systems, this scheme has following advantages:

- It combines two well-known techniques of classic one-time pads and error correcting codes to achieve a cryptographic primitive for IMDs. It inherits the property of perfect secrecy from one-time pads, and even has an ability to resist brute-force attacks.

- This scheme does not require a cryptographic infrastructure to support key predistribution, storage, revocation and refreshment. This is because keys are generated from ECG signals by each sensor dynamically before each round of encryption. This scheme does not need to protect random seeds either, since the ECG signal is used as a natural random source to generate keys.

1.3.3 Efficient ECG Binary Sequence Generation

In order to reduce the latency of generating binary sequences from ECG signals, we review inherent characteristics of an ECG signal and propose a multiple fiducial-points based binary sequence generation algorithm. This algorithm uses characteristics of an ECG signal, including RR intervals, RQ intervals, RS intervals, RP and RT intervals within one heartbeat cycle. The contributions of this algorithm are summarized as below:

- We describe an overview of the algorithm which has two major processes in stages: the ECG wavelet process and the binary sequence generation process. The technique of discrete wavelet transforms is adopted to precisely detect ECG fiducial points and obtain time intervals between them. In order to reflect variations of ECG features, an adaptive binary sequence generation process is proposed to extract random bits from these intervals.

- We conduct a series of experiments to evaluate this algorithm. We analyze the performance of wavelet-based ECG signal processing, randomness of ECG features, randomness and distinctiveness of generated binary sequences, and the complexity of this algorithm.

- This algorithm uses multiple fiducial points to obtain five feature values from one heartbeat cycle. By exploiting more components of an ECG signal, we demonstrate that the time required to generate random binary sequences is reduced significantly. Thus, it achieves the design goal of low-latency, a basic requirement in a communication system.

1.4 Organization of the Thesis

In this thesis, we explore the use of ECG signals to address unique challenges in the IMD security design. By using ECG signals, the IMD security system can provide access to unauthorized doctors in an emergency situation. Thus, it balances design goals between security and safety of IMDs. We study ECG aided key distribution schemes for the IMDs by using different security primitives. An overview of the organization of the thesis is shown in Fig. 1.3. Four small circles in the figure represent four main chapters in the thesis which study the use of ECG signals for the IMD security. Specifically, each chapter of the thesis is summarized as below.

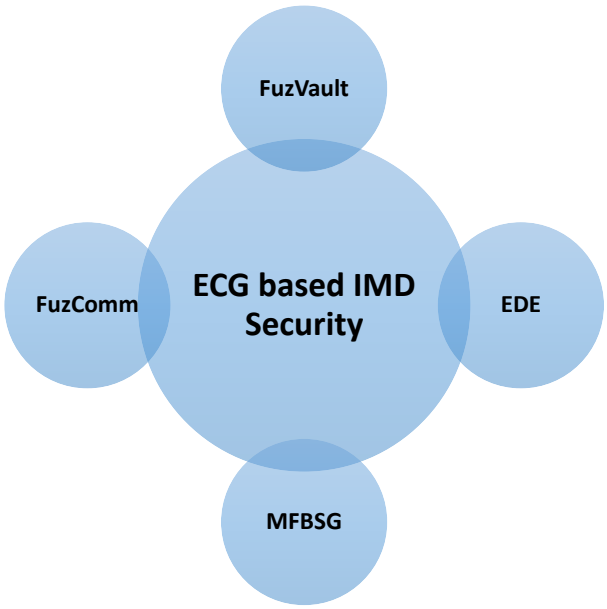


Figure 1.3: An overview of the organization of the thesis.

In Chapter 2, we conduct a comprehensive literature review in order to have a solid understanding of issues in the IMD security and current progress done in this area. The review analyzes the main threats and challenges in the IMD security design, and exam-

ines the advantages and disadvantages of current solutions proposed in the literature, pertaining to the design of secure and trustworthy IMDs.

We propose a fuzzy commitment based key distribution scheme (*FuzComm*) in Chapter 3. In this scheme, random binary sequences are generated from measured ECG signals and then are used to distribute the key from the IMD to the programmer in a secure way.

Chapter 4 presents a key distribution scheme based on the fuzzy vault primitive (*FuzVault*). The scheme uses inter-pulse intervals of ECG signals for key distribution. A polynomial is constructed in the IMD and the symmetric key is embedded into its coefficients. The programmer retrieves the key by reconstructing the polynomial.

In Chapter 5, an ECG based Data Encryption (*EDE*) scheme is proposed for the IMD security. This scheme combines two well-known techniques of one-time pads and error correcting codes. Binary sequences generated from ECG signals are used as keys for encryption and decryption directly in the one-time pads.

In order to reduce the latency involved in the process of the ECG binary sequence generation, we propose an ECG Multiple Fiducial-points based Binary Sequence Generation (*MFBSG*) algorithm in Chapter 6. It uses multiple ECG feature values to generate random binary sequences, which can be up to five times faster than previous methods that solely rely on inter-pulse interval information of ECG signals to generate such sequences.

Chapter 7 concludes the thesis by presenting a summary of the contributions of the thesis and discussing considerations and future work in the IMD security design.

1.5 List of Publications

The publications of the author during his PhD study are listed as follows.

- **Journal Papers**

[1] **G. Zheng**, G. Fang, R. Shankaran, M.A. Orgun, J. Zhou, L. Qiao and K. Saleem, "Multiple ECG Fiducial Points based Random Binary Sequence Generation for Securing Wireless Body Area Networks", *IEEE Journal of Biomedical and Health Informatics*, vol. PP, no. 99, pp. 11, 2016. (**ranked A* by CORE, Impact Factor= 2.093, SCIE indexed**, corresponding to Chapter 6).

[2] G. Fang, M.A. Orgun, R. Shankaran, E. Dutkiewicz, **G. Zheng**, "Truthful Channel Sharing for Self Coexistence of Overlapping Medical Body Area Networks", *PLoS One*, vol. 11 no. 2, p. e0148376, 02 2016 (**ranked A by CORE, Impact Factor= 3.230, SCIE indexed**).

[3] **G. Zheng**, G. Fang, R. Shankaran, M.A. Orgun, "Encryption for Implantable Medical Devices Using Modified One-Time Pads", *IEEE Access*, vol.3, pp.825-836, 2015 (**Impact Factor= 1.249, SCIE indexed**, corresponding to Chapter 5).

[4] **G. Zheng**, R. Shankaran, M.A. Orgun, L. Qiao, K. Saleem, "Ideas and Challenges for Securing Wireless Implantable Medical Devices: A Review", *IEEE Sensors Journal*, accepted in November 2016. (**Impact Factor= 1.889, SCIE indexed**) (corresponding to Chapter 2).

- **Peer Reviewed Conference Papers**

[5] **G. Zheng**; G. Fang; M.A. Orgun; R. Shankaran, "A Non-key based Security Scheme Supporting Emergency Treatment of Wireless Implants," *in 2014 IEEE International Conference on Communications (ICC'2014)*, pp.647-652, Sydney Australia, 10-14 June 2014 (**ranked B by CORE**, corresponding to Chapter 2).

[6] **G. Zheng**; G. Fang; R. Shankaran; M.A. Orgun; E. Dutkiewicz, " An ECG-based Secret Data Sharing Scheme Supporting Emergency Treatment of Implantable Medical

Devices," *in the 17th International Symposium on Wireless Personal Multimedia Communications (WPMC'2014)*, Sydney Australia, September 2014 (corresponding to Chapter 3).

[7] **G. Zheng**; G. Fang; M.A. Orgun; R. Shankaran, "A Comparison of Key Distribution Schemes Using Fuzzy Commitment and Fuzzy Vault within Wireless Body Area Networks"', *in the 26th Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC'2015)* Hong Kong, August 30 - September 2, 2015 (**ranked B by CORE**, corresponding to Chapter 4).

[8] **G. Zheng**; G. Fang; M.A. Orgun; R. Shankaran; E. Dutkiewicz, " Securing Wireless Medical Implants Using an ECG-Based Secret Data Sharing Scheme," *in the 14th International Symposim on Communications and Information Technologies (ISCIT'2014)*, Incheon Korea, September 2014 (corresponding to Chapter 4).

[9] K. Saleem, M.A. Orgun, **G. Zheng**, R. Shankaran, H. Abbas, J.A. Muhtadi, Empirical Studies of ECG Multiple Fiducial-points Based Binary Sequence Generation (MFBSG) Algorithm in E-Health Sensor Platform, *in the IEEE Workshop on Networks of Sensors, Wearable, and Medical Devices*, Dubai, United Arab Emirates, 7-10 November 2016.

[10] **G. Zheng**; R. Shankaran; G. Fang; M.A. Orgun, "An Improved Binary Sequence Generation for Securing Wireless Body Area Networks", *in the 2015 IEEE International Conference on Data Science and Data Intensive Systems*, Sydney, Australia, December 11-13, 2015 (corresponding to Chapter 6).

Chapter 2

Literature Review – Ideas and Challenges for Securing IMDs

The IMD security is an emerging research topic and has attracted a wide range of interests from academia and industry because of its importance to maintain patients' safety and privacy from attackers. As a medical device, IMD security mechanisms should support access to the IMD by emergency doctors without any pre-deployed information [12]. Compared to devices that are generally used in a general Wireless Sensor Network (WSN) environment, the implants have different operation conditions and more restricted resources in terms of battery and computation capabilities. Therefore, security solutions for WSNs cannot be readily used for the IMD environment. In the literature, many scholars have proposed security solutions from different perspectives [2]. In this chapter, we review the literature organized as follows:

- The first three sections, Sections 2.1, 2.2, & 2.3 analyze requirements in the IMD security design, including appropriate threat modeling, regulation requirements from government agencies, and trade-offs that we need to balance.
- Section 2.4 analyzes security solutions proposed for balancing the trade-off between

security and accessibility in an emergency situation.

- In Section 2.5, the proposed solutions for patients' normal check-up visit are analyzed mainly in terms of their resource consumption.
- Section 2.6 focuses on the analysis of security solutions for addressing IMD resource constraints, including low-power security algorithms and countering battery depletion schemes.

2.1 Threat Modeling

Proper threat modeling identifies and defines fundamental requirements in security scheme design. So this section discusses potential threats that IMDs need to deal with. An IMD communicates with an external programmer or a monitor through wireless channel infrequently, as shown in Fig. 1.2. The external device modifies the parameters in the IMD or transmits data from the IMD to a remote server for check-up by a doctor. In this context, we identify the adversaries who can compromise the IMD as follows:

2.1.1 Passive Eavesdroppers

A passive eavesdropper, listening to an IMD's wireless transmissions, can capture and decode transmitted data by using off-the-shelf or custom-built radio equipment. It does not interfere with IMD's communications. Recent studies demonstrated that such attacks on ICDs and insulin pumps could compromise privacy and confidentiality of patient's medical data [12, 13]. The eavesdropping attack on the ICD may disclose a wide range of patient data, including the patient' name, date of birth, medical ID number, and patient history. It is also easy to find the name and phone number of the treating physician, the model, the dates, and the serial number of the ICD and the leads [12]. The attack on the

insulin pump privacy would expose the existence of the therapy, the medical condition of the patient, the device type, and the device PIN [13].

2.1.2 Active Adversaries

An active adversary extends an eavesdropper's capabilities.It can replay recorded control commands, or generate new radio commands, to an IMD, aiming at modifying IMD's settings or triggering data transmissions actively. The adversary may use a commercial IMD programmer acquired from a hospital or elsewhere. Alternatively, the adversary may use his own software radio equipment to send commands. In this way, they can send the command at a higher power than the one originated from from a commercial programmer. The active adversary is more harmful to patients than the passive eavesdropper. As demonstrated by Halperin et al. [12], it can not only disclose patient private data, but also change therapies and deliver electric shocks to patients which would be life-threatening to the patient. Meanwhile, Li et al. [13] showed that the insulin pump could also be controlled by such adversaries. It may be programmed to stop the required insulin injection or inject more than required dose to the patient.

These adversaries can even launch a power Denial of Service (DoS) attack which has more severe impact on the IMDs than the WSNs. An active adversary can launch a battery depletion attack on IMDs by forcing the IMD to continually engage in wireless communications. When a programmer attempts to communicate with an IMD, a security algorithm in the IMD has to perform an authentication process. If the request is from an attacker, the authentication will fail and the attacker cannot access the IMD. However, this process consumes resources in the IMD, including energy and memory storage. By launching this attack repeatedly, the IMD's battery life would be decreased or even be depleted [12, 17]. This will cause a power DoS attack. It requires the IMD to be replaced by an operation when its battery goes flat. An unexpected battery depletion incident has

a serious threat to the patient's life.

2.2 Regulations

The protection of medical data privacy and security is mandated by laws and regulations, e.g., the Health Insurance Portability and Accountability Act (HIPAA) [18] and the European Union Directive 2002/58/EC [19]. In October 2014, U.S. Food and Drug Administration (FDA) issued guidance for manufacturers to consider cybersecurity risks during the design and development of the medical device, including IMDs [20]. Manufacturers should establish design inputs for their device related to cybersecurity and establish a cybersecurity vulnerability and management approach for their devices.

The need for effective cybersecurity for medical devices has become more important, as more medical devices are interconnected and more medical device-related health information are exchanged through wired or wireless channel. In July 2015, FDA issued an alert to warn users of Hospira Symbiq infusion systems due to cybersecurity vulnerabilities with this medical device [21]. An unauthorized user can control the device and change the dosage delivered by the pump maliciously. Although currently no patient injuries or deaths associated with the IMD cybersecurity are reported, the news that the U.S. Vice President Dick Cheney disabled the wireless telemetry interface of his implanted pacemaker can be a telling example of this kind of risks [22].

2.3 Trade-offs in Security Design

Since the IMD is a tiny electronic device implanted in the body for medical purposes, its security design needs to consider several trade-offs among different requirements. These trade-offs are listed as below.

2.3.1 Security vs. Accessibility

The design of IMD security safeguards should balance requirements between security and device accessibility under an emergency situation. An IMD can work in two modes: normal and emergency [23]. In a normal mode, the patients bearing one or more IMDs visit their clinic or hospital regularly. Doctors with pre-authorized programmers have access to IMDs and perform treatment, including extracting patient data, modifying IMD settings or disabling it. However, in the emergency mode, a patient may be admitted into a different hospital where there are no authorized programmers. Doctors may not be able to get security keys from the patient because the patient would be unconscious or may forget to bring his security token. Then doctors cannot gain access to the IMD for emergency treatment. This is a specific and one of the most critical requirements in the design of security mechanisms for IMDs. Currently several security schemes have been proposed to achieve this balance and will be discussed in the following section.

2.3.2 Emergency Access vs. Normal Access

The design for solving the trade-off between security and accessibility becomes prominent when the patient is in need of emergency treatment and we envisage that this will not happen frequently. However, as a chronic patient bearing an IMD, they need to visit the clinic regularly for check-up and get their IMDs fine-tuned according to their changed health conditions, where the IMD is working in a non-emergency mode. Using a conventional security scheme, a doctor can have access to the IMD by using his security key (token). Compared with the conventional scheme, schemes proposed for supporting emergency access normally have more overhead. For instance, security schemes that use an external base station as an authentication proxy [2, 23] require a secure channel established between the IMD and the external base station, and the IMD needs to sense the existence of the base station regularly. Biometric-based authentication schemes need

to measure complicated biological characteristics, e.g. fingerprints and iris [24], electrocardiograph [25] for supporting emergency access to the IMD. All these security schemes consume more IMD resources and time than the conventional one. Therefore, design of a secure key distribution scheme is necessary in order to support an IMD patient's regular visit.

2.3.3 Strong Security vs. Limited Resources

The IMD security design should achieve a trade-off between a strong security requirement and limited resources of the IMD. A strong security mechanism, which has capabilities of authentication, encryption, non-repudiation, authorization, etc., can consume plenty of resources within the IMDs. However, as a tiny wireless device, the IMD is restrained in resources in terms of computation, communications, memory and battery lifetime. Much different from generic wireless sensors, the IMD battery is non-rechargeable and non-replaceable, and are normally expected to last 5-10 years [26]. When its battery runs out, a procedure is required to replace the IMD. Although several studies proposed to charge the battery wirelessly, this design could harm organs close to IMDs [27, 28]. Moreover, increasing use of resources for IMD security may hinder resources supporting IMD utility, and even amplify the effects of power DoS attacks.

In summary, specific characteristics of IMDs (tiny, implanted, for medical purpose, long lifetime, etc.) create unique challenges for its security design. A suitable security scheme for IMDs has to support access to the IMD in both the emergency situation and the normal check-up environment as well. Besides, it needs to balance requirements between strong security and limited resources, especially battery lifetime. The algorithm should be resource-conserving to prolong the lifetime of the IMD. The design goals are shown in Fig. 2.1. Currently, a few schemes have been proposed to address these challenges, and will be discussed in the following sections.

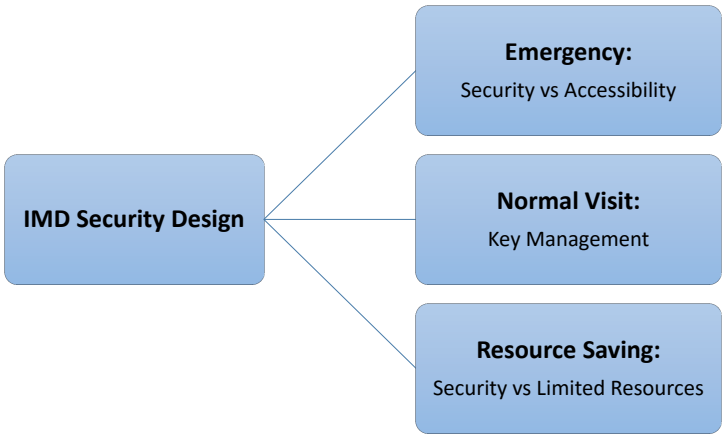


Figure 2.1: Requirements and tradeoffs in IMD security design.

2.4 Security Solutions for Supporting Emergency Access

In this section, we discuss security mechanisms that have been proposed to support emergency access. As discussed above, an IMD needs to be accessible to doctors during emergency treatment in a hospital setting where no security tokens or keys are predeployed. In order to address this challenge, several schemes have been proposed from different perspectives, e.g. external base station as a proxy [2, 1, 23, 29], biometric-based authentication [24, 25], proximity-based access control [3], and secure key distribution based schemes [30, 31, 32, 33]. Their advantages and disadvantages are studied and summarized as below.

2.4.1 External Proxy-based Solutions

The concept of using an external security proxy device, named Communication Cloakers, to protect an IMD was first proposed in [16], with following implementation protocols proposed in [2, 1, 23, 29] . Motivations and advantages of this concept are listed as below:

- It provides a fail-open access in order to achieve the tradeoff between security and accessibility. In patient's normal, everyday activities, an external proxy will protect the IMD from malicious attacks. In emergency treatment when there are no pre-distributed keys available, doctors with unauthorized programmers can simply shut down or remove the proxy, thereby gaining immediate emergency access to the IMD.

- The use of an external security proxy requires little or no modifications to the IMD. This design benefits patients who already have an IMD implanted in the body. In addition, making significant changes to the IMDs have to be approved by government administration agencies. It also increases the risk of IMD recalls. Thus, adopting this design concept helps IMD manufacturers to get their medical devices approved, thereby mitigating the risks associated with recalls.

- This design can mitigate battery draining attacks, since the majority of security operations are delegated to the external proxy. The battery of the proxy device can be easily charged and replaced. However, the battery draining attack is difficult to safeguard against, if the adversary launches intensive attacks on the device.

A common drawback of this approach is that, the patient needs to remember wearing the proxy device all times. If the proxy is not present, e.g., forgotten, lost, broken, out of battery, or stolen, then the IMD is vulnerable to attacks. This design cannot address attacks on the IMD availability, since active adversaries may simply block the IMD's wireless channel, making further wireless communication sessions infeasible. Besides, using an external device constantly reminds the patient of his/her ongoing medical conditions.

IMD Shield. An IMD Shield was designed to implement this concept [1], as illustrated in Fig. 2.2. It is a jammer-cum-receiver which synchronously receives and jams all wireless signals directed to and from the IMD. Only when a programmer is verified as a genuine party, it will act as a mediator to relay messages between the IMD and the

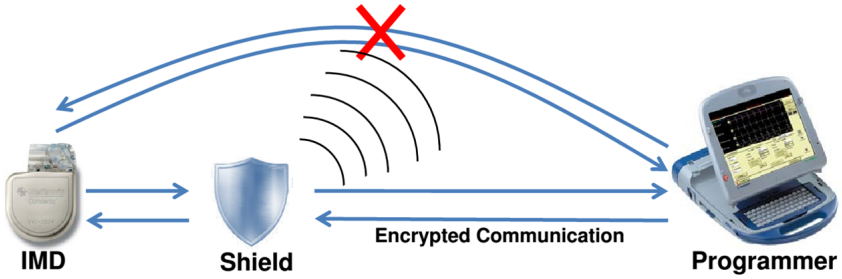


Figure 2.2: An external security Shield that authenticates programmers [1].

programmer. It establishes a secure channel between itself and the genuine party. But the channel between the Shield and the IMD is insecure. An attacker can intercept messages sent out between the Shield and the IMD, making the IMD susceptible to the man-in-the-middle attack [34]. However, establishing an authenticated, encrypted channel between the two will require modifications to the IMD.

IMD Guardian. Xu et al. proposed an IMD Guardian which performs authentication on behalf of the IMD [2]. A proxy named Guardian, as shown in Fig. 2.3, is worn on the wrist. When a request from a programmer comes to the IMD, the Guardian will also receive this request at the same time. Then the IMD starts a timer to wait for authentication results from the Guardian. If the result is *YES*, the IMD will communicate with the programmer directly. This is different from the IMD Shield scheme where all messages between the IMD and the programmer have to be relayed by the Shield. In an emergency situation, the doctor will remove the Guardian to gain access to the IMD without authentication. Compared to the IMD Shield scheme, there is a secure channel established between the IMD and the Guardian with the aid of a symmetric key. By measuring electrocardiogram (ECG) signals by the IMD and the Guardian synchronously, two random binary sequences (BSes) are extracted, respectively. These two BSes can then be used to construct a symmetric key without the need to share any knowledge a priori. Nonetheless, as discussed in [35, 36, 25], it is still a challenging task to make these

two BSes exactly the same. Besides, this design requires modifications to be made to the IMD.

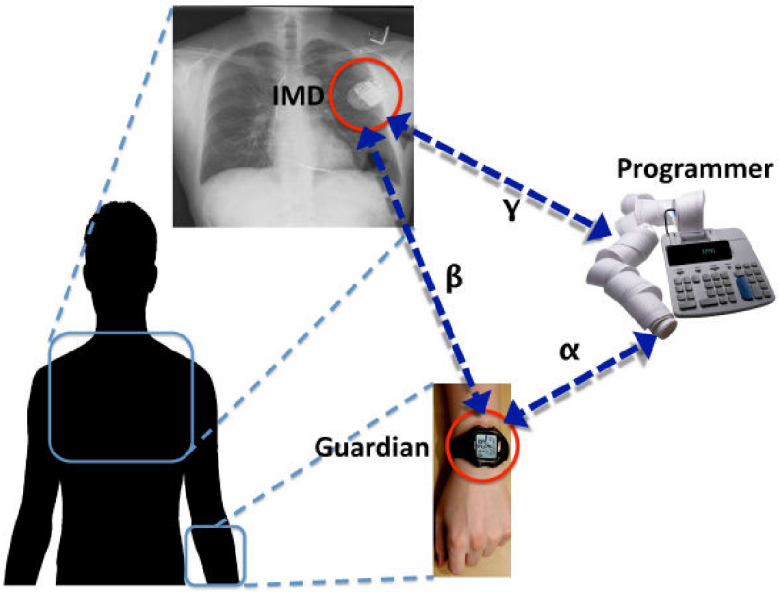


Figure 2.3: An IMD Guardian is worn on the wrist and used as a security proxy to protect the IMD [2].

MedMon A MedMon scheme [29] uses an external security monitor to snoop on all messages to and from the IMD and to detect potentially malicious transactions. It detects physical anomalies by observing signal characteristics, such as the received signal strength indicator (RSSI), time of arrival (TOA), differential TOA, and angle of arrival (AOA). It is also designed to detect behavioral anomalies, such as a malicious command that orders repeated or large-dose drug injections to the patient. This scheme can provide a non-invasive protection. It does not require any hardware or software modifications to the IMD. However, this scheme cannot protect the IMD privacy, since eavesdroppers may wiretap messages coming to and from the IMD in the channel.

User Alert. A user alert is an external device which can notify the patient of particular events happening on their IMD via an alarm signal (e.g. a loud sound or vibration). The events are normally suspicious , such as a malicious programmer attempting to access

the IMD or repeating this authentication procedure to drain the IMD resources. The notification from this alert is informative only, and cannot prevent attacks from adversaries. However, this alert function can be embodied in an existing security proxy, such as the IMD Sheild or the IMD Guardian. The MedMon scheme uses this alert function to notify patients of malicious transactions [29].

Table 2.1: A comparative analysis of external proxy based security solutions for IMDs.

Scheme	Security functions	Techniques	Modifications to the IMD	Resource Consumption
IMD Shield	Authentication, Encryption	Physical layer, Jamming	No	Minor
IMD Guardian	Authentication, Encryption	Authentication protocols, Jamming	Yes	Yes
MedMon	Anomaly detection	Detection algorithm, Jamming	No	Minor

In summary, these schemes use an external proxy device to prevent malicious programmers by authentication or anomaly detection. Emergency access is provided by shutting down or removing the proxy. The IMD Shield scheme is a physical layer solution. All schemes use the jamming technique to block wireless transactions in order to counter active attackers. The IMD Guardian scheme needs to add ECG signal processing and BS generation functions to the IMD. On the contrary, the IMD Shield scheme and the MedMon scheme do not require any modifications to the IMD. Therefore, the IMD Guardian scheme consumes more resources of the IMD than others. The comparative analysis results are shown in Table 2.1.

2.4.2 Biometric-based Access Control

Biometrics measure unique identifiable attributes of people for identification and access control, such as fingerprint, iris print, hand geometry, and voice. An enrolled template of a selected biometric is stored in the device. In the verification process, a probe biometric is measured and matched against the template. There is always slight difference among measurements of a given biometric. So the similarity between the template and the probe is computed and a decision is made based on the similarity level and its relationship with a pre-determined threshold [37].

Two Level-AC. Hei et al. proposed a two-level access control (Two Level-AC) for IMDs which can support emergency access [24]. Level-one uses patient's fingerprints, iris color and height as biometric parameters to control access to the IMD. Its second level uses an iris verification scheme. Initially, a reference iris code (a fixed length binary code) is generated from a high quality iris image of the patient and pre-loaded into the IMD. In emergencies, a sample iris image is captured and converted into a sample iris code. A programmer uses this sample code as a security token to gain access to the IMD.

The use of biometrics for IMD access control does not require the patient bringing the token or remembering the password [24]. Compared to the aforementioned external proxy-based solutions, the patient does not need to worry about the proxy device being forgotten, lost, broken, or stolen. However, a security flaw in biometrics is that, the selected biometric feature, e.g., the fingerprint, is normally unchangeable and an attacker may gain access to its template. A conventional password-based scheme can mitigate this risk by storing a non-invertible cryptographic hash value of the password rather than the password itself. But this strategy is not applicable here due to variations in biometric measurements at different points of time [38].

Heart-to-Heart (H2H). A Heart-to-Heart scheme makes use of ECG signals for the IMD authentication [25]. It ensures that the IMD can only be accessed by a programmer

which is in physical contact with the patient. The IMD and the programmer measure ECG signals synchronously and generate random BSes, k_{ai} and k_{bi}, respectively. These two simultaneously generated BSes are similar with each other only when the ECG signals are measured from the same body at the same time. This characteristic ensures that an attacker cannot have access to the IMD either by using ECG signals from the patient's past or a different person. Compared to the two-level access control scheme, this scheme does not store any biometric template in the IMD. So the template leaking risk is avoided. However, the IMD needs to measure and process ECG signals in each verification attempt, which is energy consuming. Adding an ECG measurement and processing functionality to the programmer using an external device is a trivial task.

The H2H scheme assumes the existence of a secure channel between the IMD and the programmer in which a BS is transmitted to another entity. However, this prerequisite may not be practically achieved in an emergency situation. As we have discussed above, a patient may be admitted into a hospital where its programmer does not share any information with the patient's IMD beforehand. In this case, the secure channel cannot be established. Nonetheless, we can assume that the doctor operates the programmer in a safe environment where attackers hardly exist, e.g. an emergency room in a hospital. If this assumption is normally valid, then the BSes can be sent out to each other in plain-text. This simplified process avoids the prerequisite of a secure channel between two entities and conserves power of the IMD.

2.4.3 Proximity-based Security Schemes

The core idea of proximity-based security schemes is that, they determine whether an external programmer is legitimate or not according to the distance between the IMD and the programmer. Only when this distance is within a secure and acceptable range, the programmer is allowed to have access to the IMD. The IMD telemetry interface can

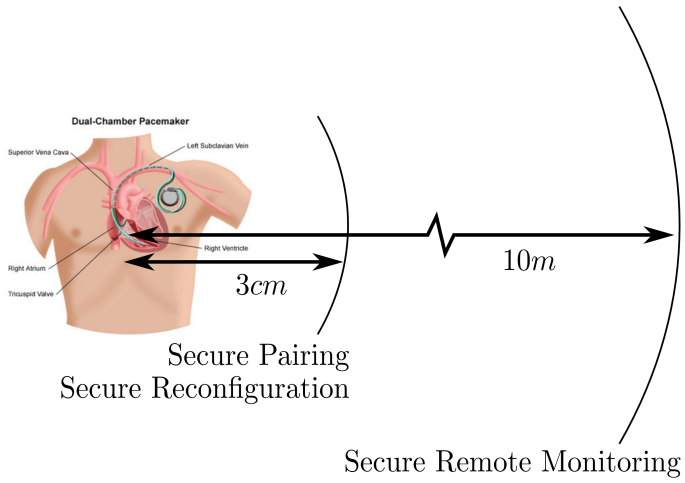


Figure 2.4: A proximity-based access control scheme for the IMD [3].

be used for two purposes: remote monitoring and critical operations. The security range for each operation is defined differently, as shown in Fig. 2.4. Critical operations, e.g. fine-tuning the IMD, is more important than remote monitoring, and requires a more stringent security policy. Thus it should use a security range much smaller than that used for remote monitoring.

Ultrasonic-AC. The ultrasonic based access control (Ultrasonic-AC) scheme, proposed by Rasmussen et al. [3], combines proximity with security credentials. The protocol uses ultrasonic distance bounding technique to measure the range between the IMD and the programmer. The patient carries a security token that shares a secret key with the IMD. In the normal operation mode, the doctor places the programmer within a security range and uses the token from the patient to gain access to the IMD. In the emergency mode when the token is not available, the IMD will generate a temporary secret key and share it with a programmer which is within its security range. However, this proximity-based security scheme could be breached if the adversary can get close to the patient, e.g. in the public transportation or other public area. In addition, the adversary may

use a technique to simulate it by being present within the acceptable security range of the patient. By sending a radio signal to the IMD, it will induce a current in the audio receiver circuit. This is similar to that of receiving a sound signal in the IMD.

The proximity based access control concept is currently adopted in the IMD design by using technologies, such as magnetic fields and short-range communication methods. For instance, an ICD, with a magnetic switch inside, will turn on its wireless telemetry only when there is a strong magnetic field in proximity [12]. Some communications technologies restrain wireless communications within a short range and have been adopted in the IMD design, e.g. RFID [39], Medical Implant Communications (MICS) [40], Bluetooth [41]. However, these methods are quite insecure. Any strong magnetic field can trigger the magnetic switch and turn on wireless transmissions. Even a magnet from headphones may interfere the patient's pacemaker and ICD and de-activate it [42]. Security schemes using short range communications may be compromised by adversaries who use powerful and sensitive transceivers and high-gain antennas to extend its communication range [12, 3].

2.4.4 Key Distribution Supporting Emergency Access

The trade-off between security and accessibility is enhanced due to the limitation of carrying out key distribution that are prescribed by conventional schemes in an emergency situation. If a symmetric key could be deployed between the IMD and the programmer, the conventional security schemes would be deployed in the IMD security design. Several techniques can be used to deploy keys for supporting emergency access to the IMD, e.g., public key cryptography, physiological signal based key distribution.

Direct-KD. A direct Key Distribution (Direct-KD) method can be used to provide the key straightaway during the emergency situation by printing the key on a bracelet or the patient's skin. Emergency doctors can read the key and obtain access to the IMD. The key may be engraved on a bracelet [31], or stored in a smart card [43]. The patient

has to wear the bracelet or bring the smart card and present it to the doctor in their normal visits or in case of an emergency. Alternatively, a visible or UV-visible tattoo can be printed on the skin to represent a scannable password, so the patient does not need to wear anything [31, 32]. However, the use of this method makes it very hard to revoke or re-issue the key. Some damage to the skin may hinder access to the IMD in emergencies. Another possible solution is to use a universal key for IMDs with the same model. But the adversary can also discover the key through side-channel attacks or by hacking into the doctor's computers [44].

Public Key Cryptography. With a public key infrastructure, a certificate with a trusted party's public key can be deployed in the IMD initially [45]. In emergencies, a programmer contacts the party and obtains a valid certificate which is later used to establish a symmetric key between the IMD and the programmer. Nonetheless, this scheme requires at least a national wide deployment of the Certificate Authority (CA) and the availability of Internet access anywhere, anytime for emergency doctors. Besides, the public key cryptography is too expensive in terms of computation and energy consumption [46], so it is inappropriate for medical sensor devices.

ECG-KD. ECG signal based Key Distribution (ECG-KD) have been studied for key distribution in wireless body area networks (WBANs) and IMDs [47, 35, 30, 48, 49]. The IMD and the programmer have to measure ECG inter-pulse-intervals (IPIs) synchronously at the beginning. The PSKA scheme is based on a fuzzy vault theory and uses a polynomial to convey the key securely from one sensor to another [47, 50]. But the polynomial computation and construction is computationally expensive for an IMD with limited resources. Using a fuzzy commitment theory [51], a symmetric key is encrypted by a random BS generated from ECG signals, and decrypted in another WBAN sensor by a synchronously generated BS [35]. Because of uncertainty of measurements from physiological signals, there are bit errors between two BSes, k_{ai} and k_{bi}. So, the challenge

underlying these schemes is how to generate two random BSes with a highly matching pattern [30, 36].

Recently, some researchers propose to extract the symmetric key from wireless channels [33, 52, 53]. These approaches use the symmetrical characteristic of the wireless channel between two wireless sensor nodes. If we use this technique in the IMD settings, then the IMD will measure a signal sent from the programmer while the programmer measures the same signal sent from the IMD. As this signal passes the same channel with the same fading, the received signals by two devices are highly correlated and two high-matching binary sequences can be generated by each device. This is similar to the sequence generation from synchronously measured ECG signals.

2.4.5 Comparative Analysis and Summary

A common underlying assumption for all these schemes is that the IMD will provide open access to programmers (even not authorized) in an emergency situation. This is because utility and safety of the IMD overrides its security and privacy [5, 30]. These security schemes are proposed from different perspectives, but some schemes use common techniques. For instance, the technique of ECG signal processing is used in schemes of IMD Guardian [2], H2H [25] and ECG-KD [47, 30]. Several schemes propose to use the criticality-aware method for the IMD security [25, 24, 30, 17]. It provides open access to all unauthorized programmers when the IMD detects that the patient is in a critical condition.

There are three important aspects of an IMD security scheme:

1) IMD Modifications. The IMD security design is for current commercial medical devices. As aforementioned, making less changes to current IMDs has benefits as below. First of all, it can reduce risks of recalls after the security function is embedded. Normally the IMD is a complicated medical device and implanted in the body. Making less changes

to it can reduce potential problems introduced to current IMD functions. Secondly, it can help manufacturers to get the IMD design approved by government administration agencies. A new IMD product, as a medical device, needs to go through severe regulations by government agencies. Ideally, if no changes are required, this security scheme can even be used for IMDs which have already been implanted in the patient's body.

2) IMD Resource Consumption. The IMD, as implanted in the body, has extreme size and power constraints. Some IMDs, such as pacemakers and ICDs, have to last 5-10 years. Therefore, the IMD security module should not affect its safety and utility functions. It requires the security module to consume as less resources of the IMD as possible. The resource consumption of a security scheme within an IMD is mainly determined by the functionalities that are added to the IMD.

3) Patient Values. As recipients and consumers of IMD devices, patient's values play an important role in the IMD design. Some technically viable systems would be undesirable to patients. Patient's values that affect the IMD design include perceived privacy, psychological welfare, convenience, cultural and historical associations, self-image, etc. [31]. For instance, several schemes are based on an external device, such as the IMD Shield, the IMD Guardian, and a bracelet with passwords engraved on its back. But some patients disliked this kind of design due to its breach of patient's privacy and inconvenience. Patients were concerned that by wearing this device would let others know of their medical conditions. It is also inconvenient since the patient needs to remember to wear it and charge it constantly. Besides, wearing something constantly reminds the patients of their underlying medical conditions. Patient values can be reflected in many aspects [31, 54].

Table 2.2 compares main IMD security schemes from these aspects. The schemes in the table are classified according to the level of modifications within the IMD. The analysis of the table is described as below.

- Schemes in Group (1) do not require any modifications to the IMD, and thus do not consume IMD resources. But they may breach patient values as they require the patient wearing an external device constantly.

- Schemes in Group (2) require adding ECG signal processing and feature extraction functions into the IMD. The ultrasonic-AC scheme requires the IMD measuring the distance by using the ultrasonic method.

- The difference between schemes in Group (2) and Group (4) is that schemes in Group (2) require the IMD measuring and processing real-time ECG signals while those in Group (4) stores pre-processed biometric features. So the two level-AC scheme consumes less IMD resources than schemes in Group (2).

- Schemes in Group (5) use conventional security solutions. The key is carried to the emergency doctors by the patient. So, they do not require extra IMD sources or time to distribute the key for emergency purposes.

2.5 Security Solutions for Supporting Normal Check-up Visit

The person bearing an IMD is normally a chronic patient. They have to visit their doctor regularly for checking their health conditions and fine-tuning their IMDs. Unlike an emergency situation, providing access to the IMD in this normal environment should avoid extra resources and energy consumption of the IMD. Ideally, we may design a scheme that can support both the emergency access and this normal access to the IMD. This section analyzes normal access method provided by schemes summarized in Section 2.4.

Some IMD security schemes, summarized in Section 2.4, can support access to the IMD in the environment of patient's normal visit to the hospital, even though their main goal is to support emergency access to the IMD. However, some schemes require consuming extra IMD resources even in the normal environment, e.g. the H2H scheme requires the IMD to measure ECG signals and generate ECG features before authentication process [25]. Therefore, if there is a key pre-distributed in the hospital that the patient regularly visits, this additional resource consumption at the IMD could be avoided.

In a routine visit to the regular hospital by the patient, we can assume that a security key of the IMD is present. It can be achieved by two ways. The patient may bring their security token to the hospital when they go for regular check-up. Or the key may be pre-deployed to the hospital by a secure key distribution scheme. Both ways are reasonable, although the latter is more convenient for the patient. In this section, we analyze how schemes in Section 2.4 support normal access to the IMD. These schemes are classified according to their operations in the normal access mode, as shown in Table 2.3. Detailed analysis is provided as below.

1. Group (1) are schemes based on an external proxy device. In the patient's regular check-up, the doctor can simply turn off or remove the proxy, forcing the IMD to enter the emergency operation mode. Alternatively, they can use a pre-distributed key to gain access to the IMD. The resource consumption of these two access methods are comparable and not burdensome to the IMD.

2. Group (2) schemes propose using physiological (ECG) signals for IMD emergency access. If we use the emergency access method to access the IMD, the IMD needs to measure ECG signals and extract ECG features for the authentication or key distribution process. When compared to the direct access method that uses a pre-deployed key, this method apparently consumes much more resources.

3. In the Ultrasonic-AC scheme , a proximity aware device pairing protocol is run even when the doctor obtains a security credential in the normal mode of operation. Since the legitimate credential is present, we may avoid running this pairing protocol and save energy consumption at the IMD end.

4. In the Two Level-AC scheme, the doctor needs to measure and obtain the patient's biometric features, e.g., fingerprints, iris. This process is done in an external device, and the IMD has pre-loaded reference features. Although this process will not consume IMD resources, the whole process is time consuming. In addition, the iris verification algorithm in the IMD is complicated and introduces some overheads.

5. Schemes in Group (5) allow the doctor to read the key from the patient's bracelet or tattoo. This could be done easily as long as the hospital provides suitable devices.

In summary, schemes listed in Table 2.3, which are mainly proposed for emergency access, normally consume extra resources and time if they are used in the patient's regular check-up environment. For conserving the IMD resources, it is essential to have a key management scheme for these medical devices. Then the doctor can obtain access to the IMD directly by using a pre-deployed key.

2.6 Security Schemes for Addressing IMD Resource Constraints

As aforementioned, IMDs have restricted resources, especially its battery. So security schemes for the IMD should be efficient and must not be manipulated by adversaries to drain its battery. Approaches proposed to address the resource constraint requirement of the IMD and also to counter its battery depletion attacks are summarized as below.

2.6.1 Lightweight Security Algorithms

A practical security function should cost as less energy of the IMD as possible. So, the use of ultra-lightweight security algorithms and protocols is necessary. Hosseini-Khayat [55] proposed a lightweight security protocol to provide data confidentiality and authentication between the IMD and its base station. It uses a lightweight block cipher and an improved protocol that does not use the challenge-response, a nonce or a stream cipher, aiming at reducing its overhead. Strydis et al. [56] studied a number of symmetric (block) ciphers in terms of various metrics, such as power consumption, total energy budget, encryption rate and efficiency, program-code size and security level. A performance and power simulator, XTREM [57], is used to evaluate the ciphers. According to the experiments, their findings indicate the best-performing ciphers to be MISTY-1 [58], IDEA [59] and RC6 [60]. A block cipher based security protocol, proposed by by Beck et al. [61], is designed for the Artificial Accommodation System (AAS). The AAS is a micro-mechatronic implant which can enable persons suffering from cataract and presbyopia to regain sight without wearing additional corrective lenses. This security protocol runs in two modes: a stream mode and a session mode. The stream mode is designed for transmitting short messages, such as short control commands, while the session mode is used for the exchange of highly sensitive information, e.g., adjusting IMD parameters and updating the IMD software.

2.6.2 Energy Harvesting

Harvesting energy for the IMD is a potential method to counter attacks on batteries. One approach to counter the battery depletion attack is to use the Radio Frequency (RF) based energy harvesting technique to power security circuitry [62, 12]. Halperin et al. [12] proposed a zero-power defense platform which does not use energy from the IMD primary battery. A Wireless Identification and Sensing Platform (WISP), with an attached

piezo-element, harvests energy from the wireless channel when it senses signals from a programmer. The WISP uses this energy for security purposes, e.g., notifying the patient, running authentication or key exchange protocols. Ellouze et al. [62] combine this WISP with the biometric key generation technique in order for the two communication parties to agree on the same key in the emergency situation. Both the IMD and the programmer measure ECG signals simultaneously and generate the same key, respectively. Besides, there are other energy harvesting and remote powering techniques for the implants, including kinetic harvesters, thermoelectric effect based harvesters, bio-fuel cells, infrared radiation based harvesters, harvesters using low-frequency magnetic fields and inductive links [63, 64]. Among these techniques, the technique of inductive links based energy harvesting is adopted to power a commercial IMD product, named "RestoreUltra SureScan MRI Neurostimulator" produced by the Medtronic [65]. In summary, the power constraints of the IMD will be mitigated when its battery can be recharged using an appropriate energy harvesting technique without resulting in any harm to body tissues.

2.6.3 Supporting Security in an External Device

Like the external proxy based security solutions [2, 16], the security computation process can be shifted to an external device. This external device informs the IMD of computation results, e.g. whether the third party (programmer) is trustable or not. The study in [17] proposed to use a cell phone to run an IMD access-pattern based detection scheme. It informs the IMD when it detects a malicious access. So, the IMD will not perform the authentication process, but instead will go to sleep mode directly in order to conserve energy.

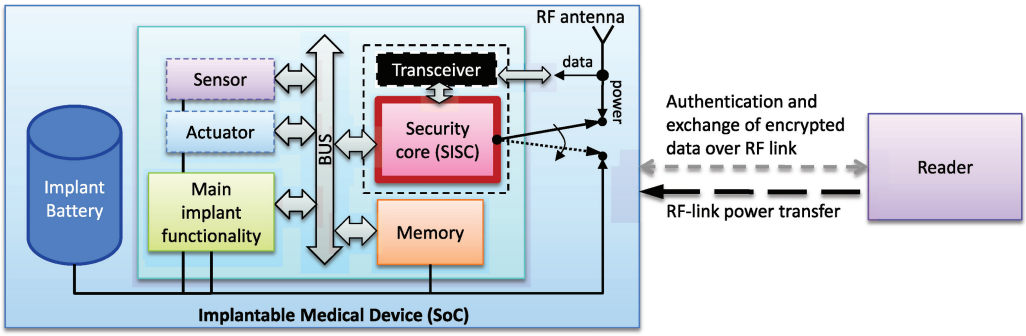


Figure 2.5: A dual core based IMD architecture [4].

2.6.4 Using a dual-core architecture

Strydis et al. [4] proposed a dual core IMD architecture, one supporting primary IMD functionality and the other exclusively handling security. A Smart-Implant Security Core (SISC), as shown in Fig. 2.5, is decoupled from the IMD main module in terms of processing and powering needs. So its security module will not interfere with functions running in the IMD primary module. It can counter battery DoS attacks since the SISC is powered by energy harvested from RF links.

Besides, some papers seek novel techniques to make a rechargeable IMD. The study in [66, 67] proposed a wirelessly charged battery circuit for biomedical implants. The energy can also be harvested from the human body, e.g. inertial kinetic energy of the body, and thermoelectric energy of the body [68, 64]. Currently, these techniques are still in their infancy and cannot be implemented in commercial IMDs. However, power constraints of the IMD will be mitigated when its battery can be recharged by the energy harvesting techniques without harming the body tissues.

2.7 Summary

This chapter has provided an investigation of security solutions proposed for the IMDs. We first analyze requirements of the security design, including threat modeling, regulation requirements, and trade-offs among different design requirements. The solutions proposed for balancing the trade-off between security and emergency accessibility have been categorized and analyzed, and their advantages and disadvantages are highlighted. We discuss the use of these security solutions in the patient's normal check-up environment. The IMD is severely restricted by its resources. We later discuss security solutions which can be resource-conserving and can counter resource depletion attacks. By conducting this survey, we find that particular requirements and constraints of the IMD create unique challenges in its security design. Designing a practical solution for an IMD is still an open issue and needs to be addressed before the wide spread deployment of the next generation IMDs.

Table 2.2: A comparative analysis of main security solutions for IMDs.

Group No.	Security Schemes	IMD Modifications	IMD Resource Consumption	Potential breach of Patient Values
(1)	IMD Shield [1] MedMon [29] User Alert	No modifications required.	No	Yes
(2)	IMD Guardian [2] Heart-to-Heart [25] ECG-KD [47, 35, 30]	1) ECG signal sampling and processing. 2) Feature extraction from ECG signals. 3) Security related communication protocols.	High	IMD Guardian: Yes Others: No
(3)	Ultrasonic-AC [3]	1) Ultrasonic based distance measurement. 2) Key storage and verification algorithm. 3) Security related communication protocols.	High	No
(4)	Two Level-AC [24]	1) Biometric features storage. 2) Biometric feature verification algorithm. 3) Security related communication protocols.	Middle	No
(5)	Direct-KD [31, 43]	1) Key storage and verification algorithm. 2) Security related communication protocols.	Low	Using bracelet: Yes Others: No

Table 2.3: Main security solutions for supporting patients regular check-up in the hospital.

Group No.	Security Schemes	Operations in Normal Check-ups
(1)	IMD Shield [1] MedMon [29] IMD Guardian [2]	The doctor makes the IMD enter an emergency operation mode by turning off the proxy. *Or:* 1) The patient brings a security token or the doctor has a pre-distributed key. 2) The doctor uses the key and gets authenticated by the proxy device.
(2)	Heart-to-Heart [25] ECG-KD [47, 35, 30]	1) The IMD and the programmer measure ECG signals synchronously. 2) Both entities extract ECG features, e.g. IPI values or binary sequences. 3) By using ECG features, the programmer obtains the security key, or gets authenticated. *Or:* The doctor uses a pre-deployed key and gets access to the IMD directly.
(3)	Ultrasonic-AC [3]	1) Ensure the programmer is close to the IMD. 2) Use a security token to gain access to the IMD.
(4)	Two Level-AC [24]	1) The doctor measures and obtains the patient's biometric features, e.g. fingerprints, iris. 2) The doctor inputs these feature values to obtain access to the IMD.
(5)	Direct-KD [31, 43]	1) The doctor has a pre-deployed key or reads the key from the patient's bracelet or tattoo. 2) The doctor uses the key to obtain access to the IMD.

Chapter 3

Fuzzy Commitment based Key Distribution for IMD Security

As discussed and highlighted in Chapter 2, a great challenge in the design of IMD security is the trade off between*Security vs. Accessibility*. The security mechanism of an IMD has to guarantee that the IMD is accessible by an emergency doctor without any pre-deployed security related information. In a conventional security system, a symmetric key is used for authentication or encryption purposes. Anyone who wants to have access to the system needs to obtain the key beforehand. However, if we use this conventional security mechanism for the IMD, how can the doctor obtain the key in an emergency situation? Patients bearing an IMD may need an emergency treatment when they travel to other cities or countries. The emergency doctors in these cities may not have a pre-deployed key. As discussed in Chapter 2, saving the key in a remote server or using a backdoor key have limitations and vulnerabilities.

In this chapter, we propose a fuzzy commitment (FuzComm) based key distribution scheme which uses ECG signals to encrypt and transmit the symmetric key from the IMD to an external trusted programmer. This scheme uses the fuzzy commitment theory to perform this key distribution process [51, 69]. It can provide data confidentiality of the

key for the IMD against eavesdropping and other active attacks from adversaries. The scheme establishes a secure channel model which encrypts the key with real-time ECG data over a wireless transmission medium. This key can only be revealed by an IMD programmer which has the ability of measuring real-time ECG signals synchronously with the IMD. This scheme is robust since it can tolerate system noise, including the noise from a mismatch between two ECG binary sequences (BSes) and the wireless channel noise. Because it does not require pre-deployment of credentials, the IMD can be accessed by doctors without any key distribution. Performance analysis based on the real ECG data (obtained from the MIT PhysioBank database [70]) shows that this scheme can transmit the key from the IMD to the programmer securely.

The FuzComm scheme implements a simple security policy for the IMD which we call *"touch-decipher"*: a programmer is authorized to have access to the IMD if and only if it has a significant physical contact with the patient's body. This access authorization will be disabled once the programmer loses physical contact with the patient. It is based on the common sense that a person who can contact the patient physically has the ability to cure or harm the patient. This touch-decipher policy balances the conflicting requirements of security and accessibility. Emergency medical responders can gain access to the IMD by making a physical contact with the patient's body while the adversary's access is to be prevented without access to real-time ECG data. This chapter is organized as follows:

- Section 3.1 describes the problem of using conventional security methods for the IMD, and appropriate assumptions in the FuzComm scheme design.
- We present the design of the FuzComm scheme in Section 3.2 & 3.3, which includes the overview as well as a detailed description of the scheme.
- In Section 3.4, we analyze characteristics of ECG signals, and present a BS generation algorithm with high matching performance.

- The performance of the FuzComm scheme is analyzed in Section 3.5. The real ECG data obtained from the MIT PhysioBank database are utilized in the experiments.

3.1 Problem Formulation

IMDs communicate with an external device called an IMD programmer infrequently. A wireless session with the IMD is initiated by the programmer during which the private data in the IMD are shared with or the parameters of the IMD are modified by the programmer. By the U.S. Federal Communications Commission (FCC) requirement, the IMD normally does not initiate a session unless it detects a life-threatening condition, e.g. life-threatening arrhythmias for pacemaker and ICD patients [1]. As discussed in Chapter 2, currently wireless communications of most IMDs are not secure [12, 13]. However, traditional security mechanisms cannot directly applied into the IMD, since the access to security keys or credentials is difficult when the patient visits a different hospital in an emergency situation. Therefore, the IMD security issues have to be addressed to allow for the deployment of a secure, robust, and efficient IMD based system.

In the FuzComm scheme, we assume that adversaries cannot measure real-time ECG signals from a patient; however, they can still use historical ECG data of the patient to attack this scheme. As measuring ECG signals requires a physical contact with the patient's body, the adversaries will be detected by the patient immediately if they try to measure the patient's ECG signals. Moreover, physical attacks could be launched on the patient's body if an adversary has the ability to make a physical contact with the patient. We also assume that medical personnel are trustworthy and hospitals provide a safe environment. This is reasonable as government regulatory agencies will oversee the conduct of doctors in hospital settings.

3.2 Scheme Overview

The FuzComm scheme includes two components: the IMD and the external programmer. The IMD, implanted in the body, assists and/or monitors the patient's health, while the programmer is an outside device which can access data in the IMD and program it wirelessly, as shown in Fig. 3.1.

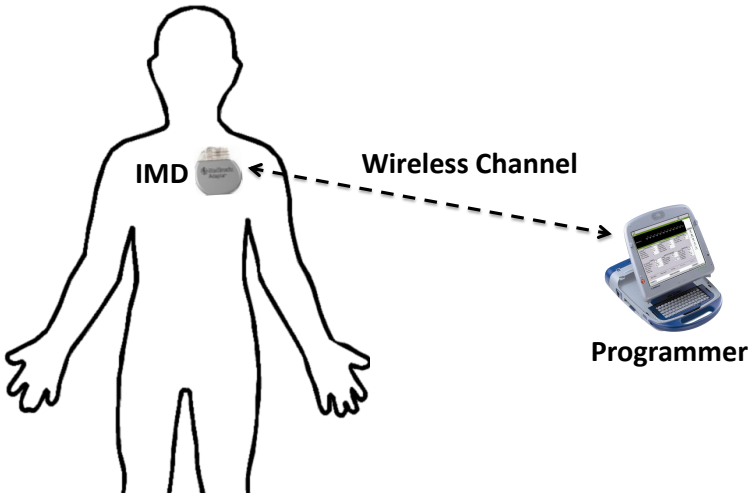


Figure 3.1: Wireless communications between an IMD and an external programmer.

Both the IMD and the programmer are currently standard medical devices and most IMDs have the capability of measuring ECG signals [1, 2]. In our scheme, an ECG sensor is connected to the programmer and is to measure ECG signals from the wrist of the patient. The IMD and the programmer will measure ECG signals synchronously and then transmit the key via the wireless channel, as shown in Fig. 3.2. In order to secure the key transmission in the channel without pre-deployment, the FuzComm scheme utilizes a secure channel model in which ECG signals measured from the IMD and the programmer

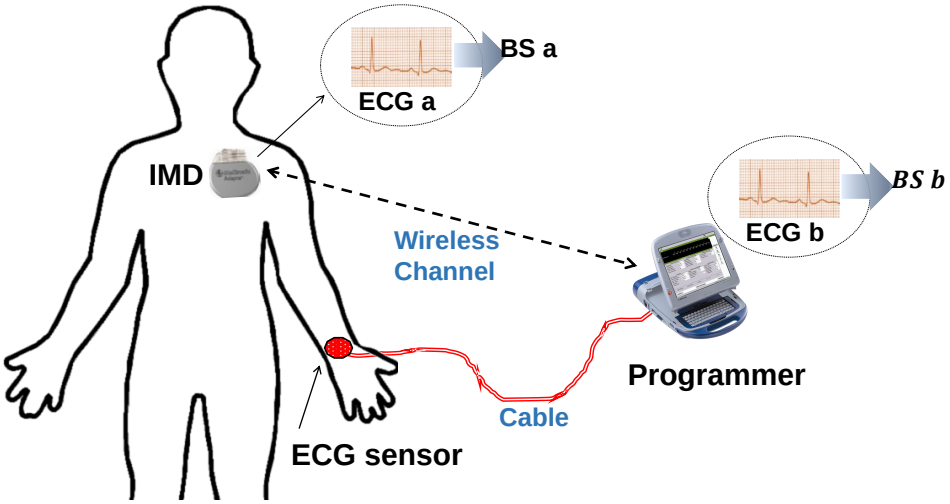


Figure 3.2: The system structure of ECG based security schemes for the IMD, which are used for schemes in Chapter 3 & 4 .

are used to encrypt and decrypt the key, respectively, as shown in Fig. 3.2.

The FuzComm scheme provides data confidentiality by securing the channel against adversaries with following steps. Firstly, the IMD and the programmer measure ECG signals from the same patient synchronously, and then generate ECG BSes with the same signal processing algorithm, respectively. Secondly, the IMD encodes the symmetric key with an ECG BS (hiding) and sends the encoded key into the wireless channel. After that, the programmer receives the encoded secret key and decodes it with the corresponding ECG BS (revealing). A hashed value of the key will be sent from the IMD to the programmer, which will be used to check the integrity of the decoded key. If the channle experiences serious channel fading, then the decoded key cannot pass this check and will be discarded. In this case, a repeat process is required to re-do the whole process in order to obtain the key.

We do not assume that the authorized programmer must have any credentials stored at the IMD. Thus the patient can be treated by doctors with a programmer in any hospital

in the emergency situation. In contrast, an unauthorized programmer has no capability to reveal the secret key as long as it cannot physically measure real-time ECG signals.

One key feature of the FuzComm scheme is secrecy which is achieved due to the inherent randomness of an ECG signal. That is, the ECG-encoded secret key can only be decoded by the ECG signals measured from the same subject synchronously. Another key feature of the FuzComm scheme is robustness due to the application of Error Correcting Codes (ECC) and the reconciliation between two ECG BSes. The ECC has the capability to correct error bits caused by system and wireless channel noise, including the mismatch between two ECG BSes generated from the IMD and the programmer. The reconciliation process can reduce the mismatch rate between two ECG BSes.

3.3 Fuzzy Commitment based Key Distribution

In this section, we describe the mathematical model of the FuzComm scheme. Following the classic Shannon communication model [71], the FuzComm communication model is illustrated schematically in Fig. 3.3. It consists of five parts:

3.3.1 Information Source and Destination

An information source is an IMD carried by a patient into a hospital where there is no pre-deployed credential of the IMD but the patient needs to be treated in an emergency situation. The secret key in the IMD, denoted by K, has to be shared with the doctors, so that the doctors can learn the patient's information and program the IMD accordingly. The IMD, after reveiving a request from the programmer, will measure ECG signals and generate ECG binary sequences for the purpose of encrypting the key K.

A destination is a programmer to whom the secret key K needs to be transmitted to. The programmer is operated by the doctors in order to obtain K. It measures real-

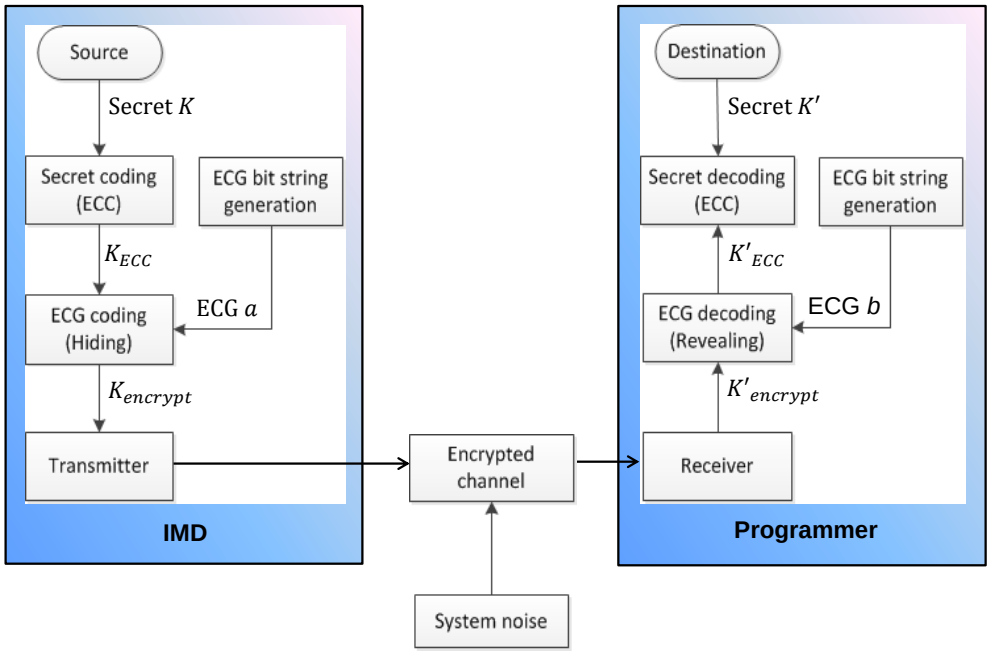


Figure 3.3: The schematic communication model of the FuzComm scheme.

time ECG signals from the patient's wrist synchronously with the IMD, as shown in Fig. 3.3. Then the programmer uses the same ECG BS generation algorithm as the IMD to generate random binary sequences from the measured ECG signal.

3.3.2 ECG Binary Sequence Generation

When the secret needs to be shared between the IMD and the programmer using the ESS scheme, they have to first extract ECG features. The IMD and the programmer achieve this goal by sampling the ECG signals simultaneously. An example of two simultaneously sampled ECG signals is shown in Fig. 3.4. In the figure, ECG1 and ECG2 are two ECG traces measured from different parts of the same body of the patient synchronously. In one ECG trace there are three major waves – namely P wave, QRS complex

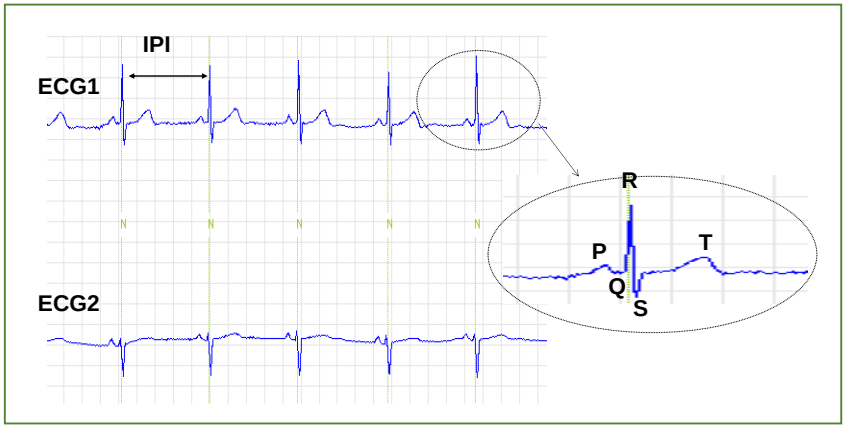


Figure 3.4: Two simultaneously sampled ECG signals from the same subject.

and T wave [72]. The P wave represents the depolarization impulse of the atria; QRS complex represents the ventricular depolarization while the T wave represents the ventricles repolarization. As the QRS complex is normally more significant than P and T waves, the R peak of the QRS complex is used here to generate ECG binary sequences. Given a continuous ECG waveform, Inter-pulse Intervals (IPIs) are computed by the interval of two consecutive R peaks in time domain, as in Fig.3.4. Suppose $t_{R(i)}$ is the timing of the i^{th} R peak, then $IPI_i = t_{R(i)} - t_{R(i-1)}$. With consecutive IPIs, ECG binary sequences of ECG_{imd} at the IMD side and ECG_{pro} at the programmer side will be then generated for the commitment process. ECG quantization algorithm is described in some detail in the Section 3.4.

3.3.3 System Noise

System noise in the FuzComm scheme includes noise from a mismatch between two ECG binary sequences, ECG_{imd} and ECG_{pro}, and noise from the wireless channel. Because of measurement inaccuracies of ECG sensors and the complexity of the ECG signals, the mismatch rate between ECG_{imd} and ECG_{pro} holds the major part of the system noise,

and would be up to 15% according to the ECG data analysis described later in this chapter under simulation section. The system noise of the FuzComm Scheme poses a major challenge in the implementation of the FuzComm scheme. In order to address this issue, we need to improve the following aspects of the scheme: use efficient Error Correcting Codes (ECC), use precise ECG measurement sensors, and design an ECG signal processing algorithm which can generate two random BSes with high matching performance.

3.3.4 Secret Key Encoding and Decoding

The technique of Error Correcting Codes (ECC) is exploited in order to correct transmission errors caused by the system noise. In the secret coding process at the IMD side, the secret K is coded by adding redundant information which is utilized to correct errors in the secret decoding process at the programmer side. As the channel noise could cause up to 15% error bits, the simple error detection method of parity checking does not work here. Some types of ECC techniques with strong error correction capability, such as BCH codes [73] and Reed-Solomon codes [74], would be applicable here.

3.3.5 A secure channel

A secure channel is one in which the transmitted secret K is encrypted with ECG binary sequences and then sent to the programmer. As shown in Fig. 3.3, it includes three blocks within the IMD: ECG binary sequence generation, ECG coding, Transmitter, and another three corresponding blocks within the programmer: ECG binary sequence generation, ECG decoding, Receiver. As the secret K is encrypted before transmission, it is resistant to adversaries who aim to harvest the patient's private data in transit.

- *ECG coding.* Within the IMD, an XOR operation is performed between the ECG_{imd} and the coded secret K_{ECC} as: $K_{encrypt} = ECG_{imd} \oplus K_{ECC}$, so the secret is hidden

in transmission. Here the K_{ECC} is the outcome of the secret coding process with the ECC and the $K_{encrypt}$ is the encrypted data or the cipher-text of the secret K.

- *Message transmitting and receiving.* The transmitter in the IMD sends to the wireless channel a message, denoted by F_{commit}, which contains the encrypted secret $K_{encrypt}$. Because of the wireless channel noise, the received message is denoted by $F^{'}_{commit}$ which contains the received encrypted secret, $K^{'}_{encrypt}$, as the input of the ECG decoding process.

- *ECG decoding.* Within the programmer, another XOR operation is performed between ECG_{pro} and the received $K^{'}_{encrypt}$ as $K^{'}_{ECC} = ECG_{pro} \oplus K^{'}_{encrypt}$. Due to system noise, $K^{'}_{ECC}$ might be slightly different from K_{ECC}.

Commitment Phase: The process at the IMD side, including the secret coding and the ECG coding, is the commitment phase, and the encrypted data (commitment), $K_{encrypt}$, binds and hides the secret K.

Decommitment Phase: The process at the programmer side is the decommitment phase, including the ECG decoding and the secret decoding. After this, the secret K is revealed. A check process is performed by the programmer in this phase in order to check the correctness of the revealed secret.

3.3.6 Example

Suppose we want to send a secret, $k = 0, 1$, to the programmer as a simple example. Considering an ECC with code set $C = 00000, 11111^2$ with a decoding function f_C and a code distance of 5, it can correct up to 2-bit errors. Two ECG Inter-Pulse-Intervals (IPIs) are measured from the IMD as $ECG_{IPI1} = 845ms, 853ms$, and from the programmer as $ECG_{IPI2} = 846ms, 855ms$. Extracting the lower five bits from each IPI binary, we can

get two ECG binary sequences as $ECG_{imd} = 01101, 10101$ and $ECG_{pro} = 01110, 10111$. Here the difference between ECG_{imd} and ECG_{pro} is the noise of the encrypted channel.

The FuzComm scheme works as follows: in the secret coding with code C, $k_{ECC} = 00000, 11111$. In ECG coding, $k_{encrypt} = ECG_{imd} \oplus k_{ECC} = 01101, 01010$. Suppose the wireless channel noise is simply omitted, then the received $k'_{encrypt} = k_{encrypt}$. In ECG decoding, $k'_{ECC} = ECG_{pro} \oplus k_{encrypt} = 00011, 11101$. As f_C can correct up to 2-bit errors, the output of the secret decoding is $f_C(k'_{ECC}) = 0, 1$. Therefore, the programmer obtains the secret k from the IMD.

3.4 ECG Binary Sequence Generation Algorithm

In this section we present an ECG binary sequence generation algorithm which is used within the FuzComm scheme. The real ECG data, obtained from the MIT PhysioBank database (*http://www.physionet.org/physiobank*), are utilized here. Analysis tools from the PhysioBank website, such as the WFDB for GNU/Linux, and the MATLAB are employed to analyze ECG signals [70].

For the design goal of security, we need to make sure that the commitment sent from the IMD cannot be decrypted by any non-real-time ECG signals from the same subject. This requires the randomness of the ECG feature extracted from the ECG signal. To achieve the design goal of robustness, it requires that the designed ECC can correct errors from the channel noise in most cases. As the mismatch between ECG_{imd} and ECG_{pro} is the major part of the noise, the mismatch rate has to be reduced; meanwhile, the quantization algorithm has to balance the binary sequence generation rate and the mismatch rate.

In light of the design goals of security and robustness, the establishment of the ECG binary sequence generation algorithm has two main aspects: *quantization and reconcilia-*

tion.

3.4.1 Quantization

This subsection discusses how to extract a random binary sequence from ECG IPIs. In order to empirically estimate how many random bits we can extract from one IPI value, the IPI values are converted to binary and then examined to determine the randomness of each bit. It is obvious that the random bits are the lower order bits. As shown in Fig. 3.5, the same bit of consecutive IPI binary values form a bit sequence, so we can get 10 bit sequences (the $1st$ bit to the $10th$ bit sequence). For a random variable $\chi = 0, 1$, we can calculate the entropy of each bit sequence using the formula:

$$H(\chi) = -p_0 log_2 p_0 - p_1 log_2 p_1 \tag{3.1}$$

where p_0 and p_1 are the probability mass function of 0s and 1s respectively. The largest entropy is 1 when it follows a uniform distribution. We calculated the entropy of each bit sequence, with results shown in Fig. 3.6 It can be seen that the 1s probability of the first eight bits is around 0.5 and their entropy is close to 1. This signifies that $1s$ and $0s$ in these bits closely follow the uniform distribution.

Nonetheless, the entropy alone cannot guarantee randomness. For instance, a binary sequence with consecutive 1s in the first half and consecutive 0s in the second half still has the largest entropy. In order to ensure randomness, we performed a two-tailed runs test with a significance level of 5% on each bit sequence. In statistics, a 'run' of a binary sequence is a maximal non-empty segment of the binary sequence consisting of the same elements (consecutive 1s or 0s). The test result shows that the lower 5 bits pass the test while the 6th to 10th bits fail the test.

Therefore from the perspective of randomness, the lower 5 bits of each IPI binary

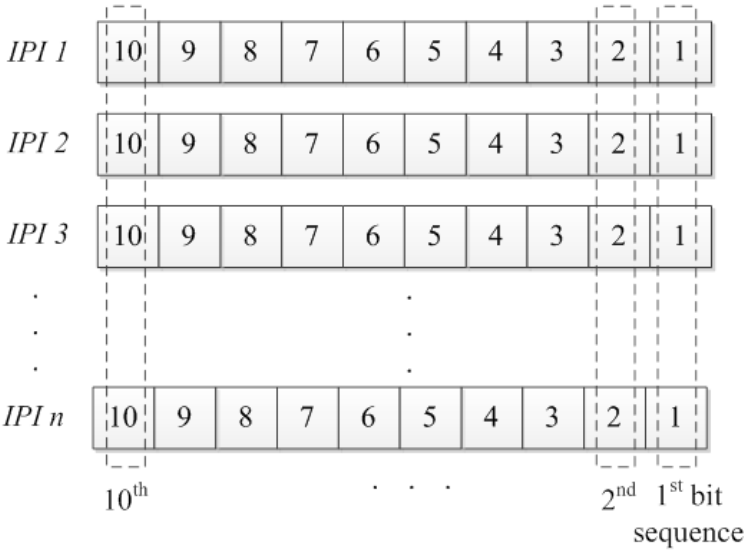


Figure 3.5: The bit sequence of one IPI binary value starting from the Least Significant Bit.

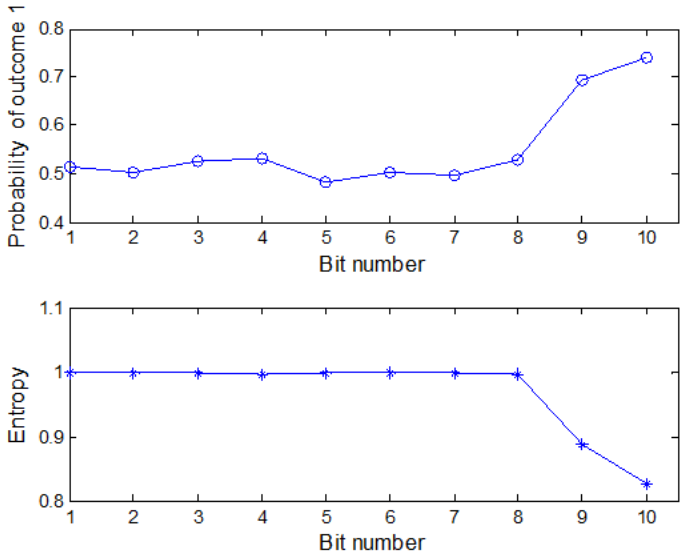


Figure 3.6: The probability of '1s' and the entropy of each bit sequence.

value can be used in quantization.

3.4.2 Reconciliation

The IPI values measured by the IMD and the programmer are mostly not equal, which causes a high mismatch rate between ECG_{imd} and ECG_{pro} (about 15% in our experimental data). For example, for the ECG signals in the experiment with 125Hz sample frequency, the measurement accuracy of each sample point is 8msec. Therefore we cannot use the bit sequence from each IPI value directly.

A classic class of cyclic ECC called BCH codes was chosen in the scheme. The principal advantage of BCH codes is that they can be decoded with a small and low-powered electronic hardware [75]. In order to prevent the brute-force attack and considering the BCH codes, we recommend that the length of the generated ECG binary sequence be at least 127 bit long. Table 3.1 shows the error correction capability of the BCH codes when the code length $n = 127$, where k is the message length and the t is the largest number of error bits it can correct. Combining the code efficiency and the error correction capability, we chose the BCH code as (127, 64, 10) whose error correction capability is up to 7.87%.

Here we set the design goal of robustness for this system: the average mismatch rate between ECG binary sequences is up to 40% of the error correction capability of the chosen ECC. In light of this design goal, the average mismatch rate between ECG_{imd} and ECG_{pro} should be around 3%. Based on these facts, we design a two-step reconciliation algorithm involving: Simple Moving Average and parity check.

(a) Simple Moving Average

As the measurement errors can be regarded as white noise, Simple Moving Average (SMA) is used here to smooth out short-term fluctuations of errors at two ends. The SMA is the unweighted mean of a series of different subsets in the whole data sequence.

Table 3.1: The error correction capability of BCH codes when n=127.

n	k	t
127	92	5
127	85	6
127	78	7
127	71	9
127	64	10
127	57	11
127	50	13
127	43	14
127	36	15

For this system, the SMA of consecutive IPIs with window size w is given by

$$\begin{cases} SMA_1 = \frac{1}{w}(\sum_{i=1}^{w} IPI_i) \\ SMA_j = SMA_{j-1} - \frac{1}{w}IPI_{j-1} + \frac{1}{w}IPI_{j+w-1} \\ j = 2, 3, ..., m - w + 1 \end{cases} \tag{3.2}$$

Fig. 3.7 shows the variation of the mismatch rate between ECG_{imd} and ECG_{pro} versus the window size when applying SMA method to the IPIs. It can be seen that the mismatch rate declines along with the window size. The decline when the window size $w < 15$ is faster than that when $w \geq 15$. In order to balance the binary sequence generation rate and the mismatch rate, we chose the window size as 14 for this system.

After the SMA process we need to ensure that the IPI data are still random. A normal probability plot of SMA processed IPIs with window size 14 is shown in Fig. 3.8 For a normal probability plot, the plot will be linear if the data are normal; otherwise, it will introduce curvature in the plot for other distribution types. As we can see in Fig. 3.8

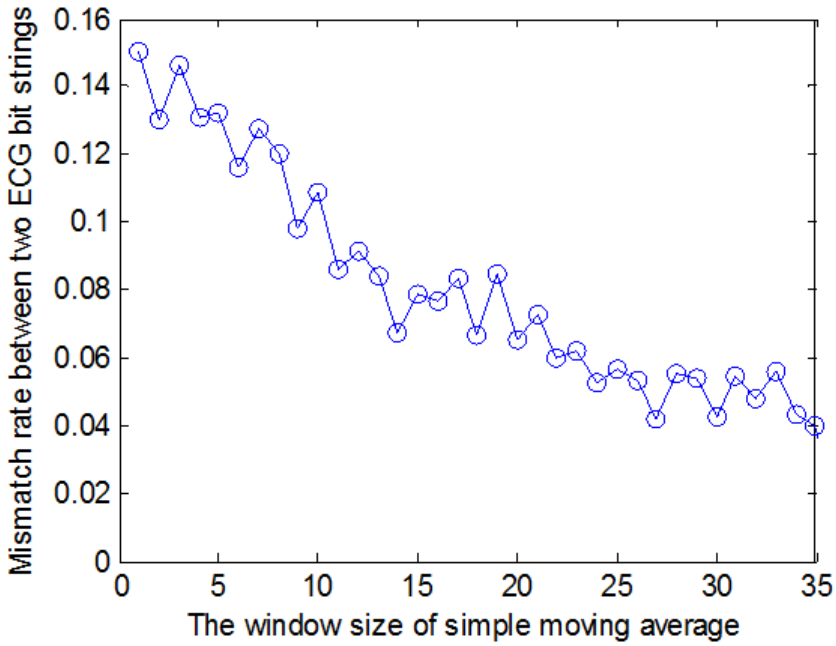


Figure 3.7: The variation of mismatch rate between two ECG strings versus the window size when applying the SMA method to the IPIs.

that the plot is nearly linear, the distribution of SMA processed IPIs is close to normal. Therefore the SMA processed IPIs are still random.

By applying the SMA with window size 14, the mismatch rate declines from 15% to 6%. However, this is still higher than our goal of 3%. So we employ step 2: Parity Check.

(b) Parity Check

We observed each pair of SMA-processed IPI binary values at both sides, and found that the Least Significant Bit (LSB) was normally different. In order to increase the success rate of parity check, the LSB is not included when extracting binary sequences from each IPI. So we can get 4 bits (the 2^{nd} to $5th$ bit in Fig. 3.5) from each IPI.

Bits from two consecutive SMA-processed IPIs at both sides are extracted to form an 8-bit block. Then both sides calculate the parity of their own block and exchange the parity information. If the parity is the same, each side extracts 7 bits of the block and

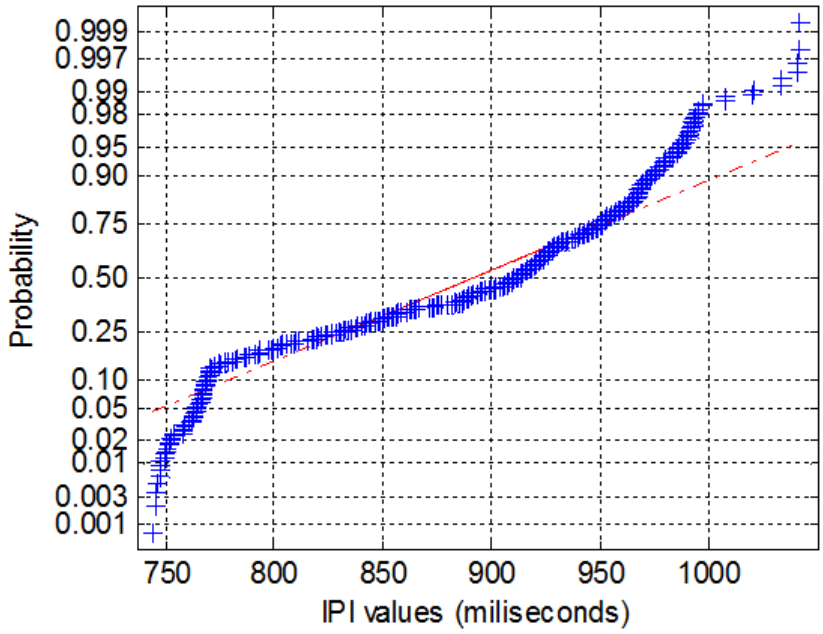


Figure 3.8: The normal probability plot of SMA IPIs when the window size is 14.

discards the last bit as the parity check leaks one bit of information. This process moves ahead until there are 127 bits on each side.

A simulation was performed to test the improvement of the parity check. The result showed that the mismatch rate was reduced to about 3%, which satisfies our design goal of robustness. As we can extract 7 bits from two consecutive IPIs, generating a 127-bit long ECG binary sequence would take about 38 seconds.

We can see from the experiment that the IPI mismatch is dominated by the difference between two ECG samples, although they are sampled synchronously from the same subject. Several approaches could be adopted to improve this mismatch performance. First, we can try to measure the two ECG signals from almost the same part of the body. So, the changes caused by the ECG signal passing through the physical channel of the body can be mitigated significantly. For example, the IMD measures the ECG from the chest where it is implanted. Then we can make the programmer to measure the ECG

from the part where it is close to the IMD, such as the skin outside the chest. Then, ECG signals received by the IMD and the programmer undergo a similar channel fading and interference. Second, we can try to improve the accuracy of ECG measurements, such as using a larger sampling rate, improving the ECG signal processing algorithm. Then, the mismatch caused by measurement errors would be reduced.

3.5 System Performance

In this section, the real-ECG data from the MIT PhysioBank are used to analyze the performance of the system in terms of security and robustness [70]. The MATLAB is used as the analysis tool. Normally, we can measure ECG signal from the wrist and the chest of the patient synchronously.

We tested the randomness of generated ECG binary sequences to ensure that the ECG binary sequence used in encryption cannot be guessed by using a brute-force attack. The EDE scheme relies upon generated ECG BSs following what Shannon defines a purely random process [76]. Our first experiment was to analyze the randomness of captured ECG IPI values. We collected $15,000$ consecutive IPI values and plotted histograms as shown in Fig.3.9. These histograms clearly show that the fluctuation of IPI values fits into a normal distribution. We also tested other IPI data sets which were collected from other patients. The test showed that all these IPIs follow a normal distribution. Thus the distribution of consecutive IPIs is almost normal, which indicates the randomness of ECG IPI values. This normal distribution is a fundamental requirement in order to generate random BSes from IPI values.

We then calculate the entropy to measure the uncertainty of generated ECG BSs. For a random variable $\chi = 0, 1$, we can calculate the entropy of each bit sequence using the Eq. 3.1. The entropy result of binary sequences generated from about 100 ECG samples

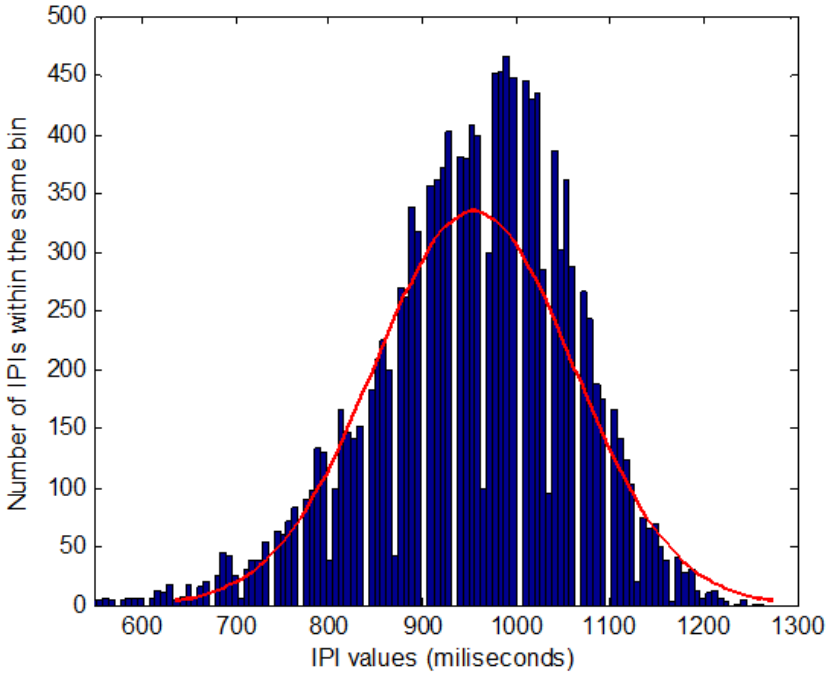


Figure 3.9: The histogram of consecutive IPI values sampled at 125Hz with a normal distribution fit ($\mu = 955ms, \sigma^2 = (106.5ms)^2$).

is shown in Fig.3.10. It can be seen that the entropy values of most ECG binary sequences were close to 1, with the mean entropy of 0.992. Furthermore, a two-tailed runs test was also performed during the experiment, which showed that more than 95% of ECG binary sequences passed the two-tail runs test with a significance level of 5%. Therefore, the generated ECG binary sequences have a good performance of randomness.

In order to comprehensively analyze the randomness of generated ECG BSs, we performed an experiment with the National Institute of Standards and Technology (NIST) randomness test suite [77]. The quality of randomness of ECG binary strings was statistically evaluated by employing the state-of-the-art NIST test suite [77] which is used for testing random and pseudo-random number generators. The outputs are p-values which indicate the probability that the generated BSs are random or not. If the p-value is less than a threshold (normally 1%), the hypothesis that a binary string is random is then

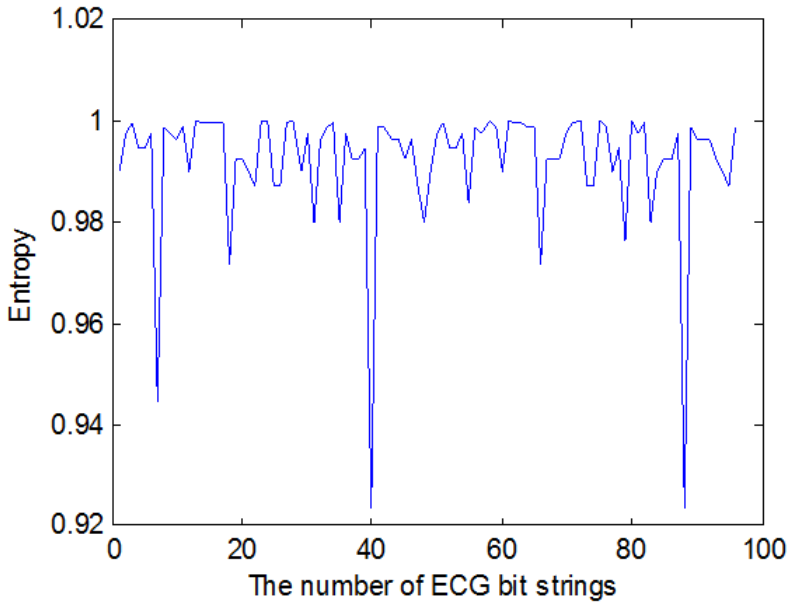


Figure 3.10: The calculated entropy of generated ECG binary strings.

rejected.

Table 3.2: NIST statistical test results for generated ECG BSes.

Statistical test	p-value	Proportion	Pass/Fail
Frequency	0.739918	1.0000	Pass
Block Frequency	0.035174	0.8000	Pass
Cumulative Sums^{+}(2)	0.862243	1.0000	Pass
Longest Run	0.122325	0.8000	Pass
FFT	0.534146	0.7000	Pass
Non-overlapping Template^{+}(148)	0.037516	0.9128	Pass
Serial^{+}(2)	0.042397	0.6000	Pass
Linear Complexity	0.066910	0.7500	Pass

We used in the test an aggregate of databases of 18 subjects from the NSRDB [70], 79 subjects from EDB [78], 47 subjects from MITDB [70] and 23 subjects from AFDB [70],

with test results shown in Table 3.2. Tests which produce multiple p-values are represented by a (+) followed by the number of different generated values in parenthesis. The table displays their mean values.

We check the p values in Table 3.2. It shows that all p values are greater than 1%. So, it passes the NIST randomness test. Therefore, these generated ECG BSs can be regarded as truly random. Then these BSs can be used for secure communications in practical OTPs.

3.6 Summary

In this chapter we presented a fuzzy commitment based key distribution scheme for the IMD security. This scheme transmits the symmetric key from the IMD to the external programmer securely, so that unauthorized doctors can obtain the key to gain access to the IMD in the emergency situation. In this scheme, the IMD and the programmer measure ECG signals synchronously from the same patient, and generate two BSes which are used for encryption and decryption of the key, respectively. A ECG BS generation algorithm has been proposed to generate two random BSes with high matching performance. Simulation analysis shows that generated ECG binary sequences are random, and thus this scheme can be used to secure key distribution between the IMD and the programmer.

Chapter 4

Key Distribution Using Fuzzy Vault Primitive for IMD Security

In Chapter 3, we have proposed a fuzzy commitment (FuzComm) based key distribution scheme for the IMD security. In the scheme, ECG binary sequences are generated by the IMD and the programmer for the key hiding and revealing purposes, respectively. However, because of variations between ECG signals from different parts of the body, it is challenging to generate the same ECG BSes from the measured ECG signals. Meanwhile, Juels and Sudan proposed another key distribution scheme for biometric applications, named a fuzzy vault scheme [50], which can avoid this BS generation process [79, 80]. In this chapter, we study the fuzzy vault based key distribution scheme (FuzVault) for the IMD security. It uses the same structure as the FuzComm scheme in which the IMD and the programmer are required to measure ECG signals synchronously, as shown in Fig. 3.2. It also implements the *"touch-decipher"* which means any doctor who has permission to touch the body of the patient will be authorized to obtain access the IMD.

In this FuzVault scheme, IPI values of the ECG signal are used directly for encryption and decryption purposes while in the FuzComm scheme they are used to generate random BSes. A set of IPIs is obtained from measured ECG signals in the IMD and

the programmer, respectively. A polynomial is to be constructed in the IMD with the symmetric key as its coefficients. The corresponding points of the IPI set in the IMD on the polynomial curve is then calculated and sent to the programmer. These points are hidden from attackers by adding noisy points into a vault, which are called *chaff points* in [50]. The programmer matches points in the vault with its own IPI set in order to determine points on the polynomial. It reconstructs the polynomial by using these points and obtains the key from the polynomial coefficients.

In this chapter, we compare the FuzComm scheme and the FuzVault scheme from different perspectives. Both schemes use ECG signals for the purpose of the IMD key distribution. They have many parts in common, such as ECG signal sampling and feature extraction, and key hiding and revealing processes. So, it is necessary to compare these two schemes when they are applied for the IMD security, and analyze advantages and disadvantages of each scheme.

The basic goal of designing this scheme is that we need to reduce overheads for communications and computations in the IMD, since the IMD, an implanted wireless device, has limited resources in terms of memory, battery and processing capability. The programmer, as an external device, has powerful computation capability and a battery which can be easily charged. So, the designed scheme should be lightweight for the IMD without compromising its security level. The organization of this chapter is as follows.

- Section 4.1 describes the mathematical model of the fuzzy vault primitive.
- We propose our FuzVault scheme for the IMD security in Section 4.2, and evaluate its performance in Section 4.3.
- In Section 4.4, we conduct a comparative analysis between the FuzComm scheme and the FuzVault scheme, since both use ECG signals for the same purpose (key distribution for the IMD security).

4.1 Fuzzy Vault Primitive and Improvement

A fuzzy vault primitive, proposed by Juels and Sudan in [50], is to lock (conceal) a secret in a construct called a vault by a set A. The secret can be unlocked (revealed) by another set B which overlaps substantially with set A. The block diagram of the scheme is shown in Fig. 4.1.

From the block diagram, we can see that the primitive is implemented in the transmitter and the receiver, respectively. Within the transmitter, the vault construction process is accomplished by the following steps:

1. Encoding the secret K with an Error Correction Code (ECC) to generate an encoded secret K_{ECC}.

2. Constructing a vth order polynomial $p(x)$ over the variable x whose coefficients $(c_v, c_{v-1}, \ldots, c_0)$ form the secret $K_{ECC} = c_v|c_{v-1}|\ldots|c_0$. Here the operator $'|'$ means concatenating these coefficients.

3. Projecting the set A onto the polynomial $p(x)$ and creating a set $R = \{a_i, p(a_i)\}$ where $a_i \in A$ and $1 \leq i \leq |A|$.

4. Adding randomly generated chaff points $C = \{c_i, d_i\}$ to R where $c_i \notin A$ and $d_i \neq p(c_i)$. The chaff points are not on the polynomial and could be generated by adding random noise to polynomial points.

The entire collection of points, $R \cup C$, constitutes a commitment of $p(x)$, that is, of the secret K. Once the vault is constructed, it can only be unlocked by the set B which has a commonality with the set A. The reveal procedure is an inverse process described as below.

1. Constructing a subset $Q \in R \cup C$ according to the element in B as $Q = \{(b_i, p(b_i))|b_i \in B\}$ by making sure that the number of elements in Q is no less than $v + 1$.

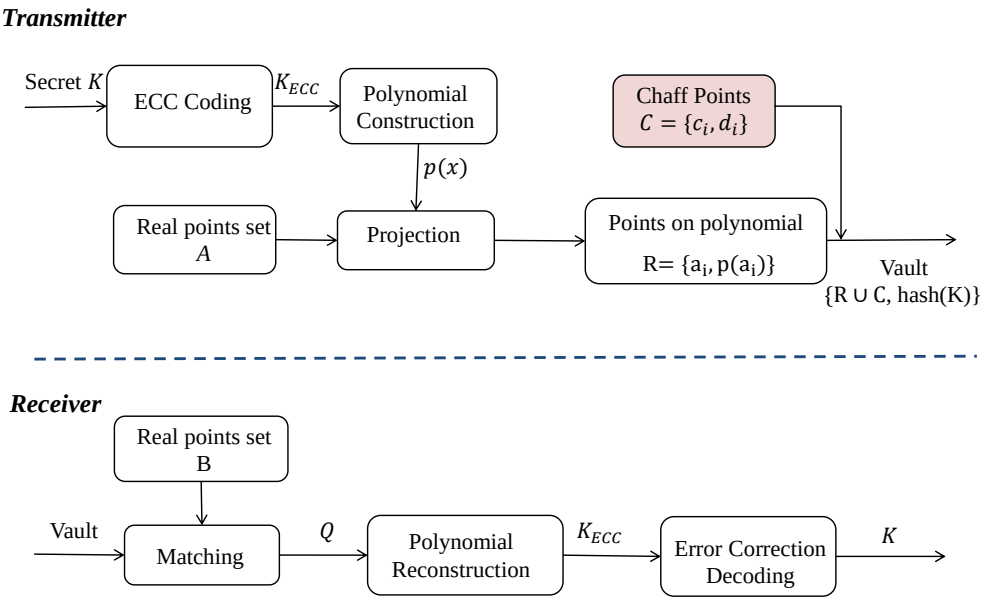


Figure 4.1: Block diagram of the conventional fuzzy vault scheme.

2. Reconstructing the vth order polynomial $p(x)$ with the set Q; Lagrangian interpolation is employed here to reconstruct $p(x)$.

3. Decoding the coefficients of $p(x)$ using the ECC decoder and extracting the secret K'. The Reed-Solomon decoder is used here to correct errors caused by the channel.

4. If the whole process is successful, then the decoded secret K' equals K; otherwise, the reveal process fails, and we need to re-do the fuzzy vault process.

The security of the fuzzy vault primitive depends on the number of chaff points added into the vault (the pink block shown in Fig.4.1) [50]. The greater the number of chaff points, the more secure the underlying scheme is. However, since the IMD has limited resources, adding a large number of chaff points increases overheads of the IMD. In order to reduce the IMD overheads, we suggest not adding chaff points into the vault but still

keep the vault secure here. As there are no chaff points, the points lying on the polynomial p would then be exposed to the attacker. In order to avoid the disclosure of the real points, we suggest sending hash values of a_i with a cryptographic hash function instead of adding chaff points. So the set R is reconstructed as $R = \{hash(a_i), p(a_i)\}$. In the unlocking process, we compare the hash values of the elements in B, $hash(b_i)$, with the x-axis values of points in R, and construct a subset $Q = \{b_i, p(a_i)|b_i \in B, hash(b_i) = hash(a_i)\}$. Then the subset Q is used to reconstruct the polynomial p in order to reveal the secret K.

Here we calculate hash values of x-axis with a cryptographic hash function. So, it is infeasible for the attacker to recover the x-axis values of points with given hash values when the search space is large enough. If an attacker launches a dictionary attack to it, the search space is the size of a_i, that is, the bit length of all values in the set. So, it is quite large for the attacker to break it. Therefore, the vault sent out would not disclose points on the polynomial line, which achieves our stated design goal of security. Meanwhile, as a large amount of chaff points are not added into the system, this improvement reduces overheads in the IMD in terms of memory, computation and communications, which achieves our design goal of being lightweight for the IMD.

4.2 Proposed FuzVault Scheme for IMD Security

Based on the improved fuzzy vault primitive described earlier, we propose a Fuzzy Vault based key distribution (FuzVault) scheme to transmit the secret key securely from the IMD to the programmer. We assume that both the IMD and the programmer have the capability to sample ECG signals in a loosely simultaneous manner. Here there is no strict requirement of synchronization as long as the set A generated from ECG signals within the IMD has a significant common part with the set B generated within the programmer. The Reed-Solomon (R-S) code [74] is used here although any type of a linear ECC is also

possible. The whole FuzVault scheme is described as follows.

4.2.1 Feature Generation

The IMD and the programmer sample ECG signals simultaneously at a specific sampling rate for a fixed time period. As shown in Fig.3.4, the QRS complex is normally more significant than P and T waves. So, the R peak of the QRS complex is used here to generate ECG features. Given a continuous ECG waveform, IPIs are computed by the interval of two consecutive R peaks in time domain. Suppose $t_R(i)$ is the timing of the i^{th} R peak, then $IPI(i) = t_R(i) - t_R(i-1)$. With consecutive IPIs measured in the IMD, the set A is established as $A = \{a_i\} = \{IPI_{imd}(i)|i = 1, 2, \cdots, n\}$. In the same feature generation algorithm, the set B in the programmer is established as $B = \{b_i\} = \{IPI_{pro}(j)|j = 1, 2, \cdots, n\}$.

4.2.2 Hiding the Secret

Once ECG features are generated, the IMD establishes a v_{th} order polynomial as,

$$p(x) = c_v x_v + c_{v-1} x_{v-1} + \cdots + c_0 \tag{4.1}$$

where the polynomial $p(x)$ encodes the secret in its coefficients. The secret used here is the output of the R-S coding process, K_{RS}, with the input K. R-S coding adds redundant information into the secret K so that it can resist noise in the wireless channel. The order of the polynomial is not a secret and known to others including attackers. With the polynomial $p(x)$ and the feature set A, the IMD constructs a fuzzy vault by computing the set $R = \{hash(a_i), p(a_i)\}$, where $a_i \in A$ and $i = 1, 2, \cdots, n$. As an attacker cannot generate a message with the hash function according to the value $hash(a_i)$, the points lying on the polynomial are hidden from the attacker.

4.2.3 Vault Exchange

The IMD sends the vault R to the programmer by using a message as: IMD $\longrightarrow$ programmer:

$$msg1 = \{ID_{imd}, ID_{pro}, Nonce, R, hash(K), MAC(.)\}. \tag{4.2}$$

Where ID_{imd} and ID_{pro} are the IDs of the IMD and the programmer, respectively. $Nonce$ (a unique random number) is used to resist against the replay attack, $hash(K)$ is the hash value of the secret K, and $MAC(.) = MAC(K, ID_{imd}|R|Nonce|hash(K))$ is a Message Authentication Code (MAC) to check the integrity of the message. The cryptographic hash function, SHA-2, is selected in the scheme to compute $MAC(.)$ and other hash values. The SHA-2, designed by the U.S. National Security Agency (NSA), is one set of hash functions widely used in security applications and protocols.

4.2.4 Revealing the Secret

Considering the noise in the wireless channel, the received vault at the programmer is denoted as R'. The set Q is created as,

$$Q = \{(b_i, p'(b_i))|b_i \in B, (b_i, p'(b_i)) \in R'\} \tag{4.3}$$

The programmer then reconstructs the polynomial $p(x)$ with points in Q. By using the Lagrangian interpolation, if we know $v+1$ points, $\{(x_0, y_0), (x_1, y_1), \cdots, (x_v, y_v)\}$, on the polynomial, then the polynomial can be reconstructed as,

$$\begin{cases} p'(x) = \sum_{j=0}^{v} y_j l_j \\ l_j(x) = \prod_{i=0, i\neq j}^{v} \frac{x - x_i}{x_j - x_i} \end{cases} \tag{4.4}$$

In order to successfully reconstruct the polynomial, the number of elements in the set Q should be no less than $v+1$. The concatenation of coefficients in the $p'(x)$ forms the R-S encoded secret K'_{RS}. After the R-S decoding process on the K'_{RS}, the secret K' is generated.

4.2.5 Acknowledgment

If the hash value of the generated secret K', $hash(K')$, equals the received $hash(K)$ in the message, then the FuzVault process succeeds. The programmer sends a reply message back to the IMD in order to inform the IMD of its successful secret key transmission with a message: programmer $\longrightarrow$ IMD:

$$msg2 = \{ID_{imd}, ID_{pro}, Nonce, MAC(K', ID_{imd}|ID_{pro}|Nonce)\} \tag{4.5}$$

where the symbols have the same meanings as before.

4.3 Performance Evaluation

In this section we analyze the security performance of the proposed FuzVault scheme. The real ECG data, obtained from the MIT PhysioBank database [70], are utilized here. Data were collected from 10 subjects, and 10 ECG data sets were generated from each subject. The ECG data is sampled at 125Hz, and there is a time-stamp associated with each value in data files. Each data set has about 5 minutes ECG data. IPIs from these ECG data are calculated and form an aggregate for the test. Larger sampling rate (e.g., 1000Hz) is better for the accuracy of the IPI calculation. However, ECG data with 1000Hz sampling rate are currently not available in the MIT PhysioBank. Analysis tools from the PhysioBank website, such as the WFDB for GNU/Linux, and the MATLAB are employed to analyze ECG data.

4.3.1 ECG IPI Randomness

An important prerequisite of applying the FuzVault scheme is that IPI values in each set should be random. Otherwise, the adversary can easily guess which part of points in the vault are on the polynomial line. We analyzed the characteristics of IPIs in Chapter 3, as shown in Fig. 3.9. It shows that the fluctuation of IPI values fits into a normal distribution curve. This normal distribution also pertains to IPIs calculated from other subjects' ECG data. So, ECG IPIs of a subject has the characteristic of randomness.

Therefore, in the vault sent into the wireless channel, the adversary cannot guess which point is on the polynomial line excepts through launching a brute-force attacking.

4.3.2 FRR/FAR Performance

In this part, parameters of FRR (False Rejection Rate) and FAR (False Acceptance Rate) are used to evaluate the performance of the proposed FuzVault scheme. FRR measures the likelihood that the constructed vault is incorrectly rejected by the synchronously recorded ECG data from the same subject, while FAR measures the likelihood that the constructed vault is incorrectly revealed by the subject's historic ECG data or ECG data from another subject. The Half Total Error Rate (HTER), calculated by $HTER = (FAR + FRR)/2$, is employed here to aggregate measures of FRR and FAR.

Ten subjects' ECG data were randomly selected from the MIT-BIH Database. The test was run on each subject ten times at ten random start-times. The FAR, FRR and HTER performance versus the polynomial degree v (the error correction capability is 2) are shown in Table 4.1.

It is shown in Table 4.1 that the FAR decreases when the polynomial degree v increases. This is because a bigger polynomial degree requires more features in common to successfully reveal the secret, then the probability of mismatching two ECG data sets decreases. This indicates that the system is more secure as the polynomial degree be-

Table 4.1: FAR, FRR and HTER performance versus the polynomial degree.

Polynomial degree v	FAR	FRR	HTER=(FAR+FRR)/2
1	0.2811	0.0040	0.1426
2	0.1243	0.0144	0.0693
3	0.0432	0.0431	0.0432
4	0.0162	0.1244	0.0703
5	0.0108	0.1962	0.1035
6	0.0054	0.3062	0.1558

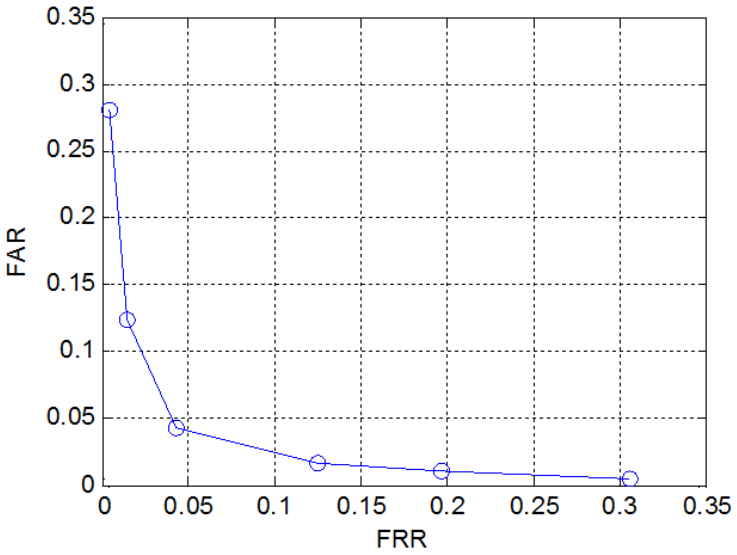


Figure 4.2: The relationship between the FAR and the FRR.

comes bigger. The table also shows that the FRR increases when the polynomial degree v increases. This is due to the fact that, when more common features in two ECG data sets are required, it is more likely for the FuzVault scheme to incorrectly reject revealing the vault, although ECG data are measured from the same subject. The relationship between the FAR and the FRR, as shown in Fig. 4.2, is that the FAR decreases as the FRR increases. In order to balance these two measures, the HTER is calculated in Table

4.1 . It shows that the degree of $v = 3$ is optimal for this system.

We also find from the simulation that the computational complexity increases exponentially when the polynomial degree increases, which will increase the overhead of the whole system. This is obvious as the system has to calculate values of the polynomial.

4.4 Comparative Analysis between Fuzzy Commitment and Fuzzy Vault

Both the FuzComm scheme and the FuzVault scheme use the same physiological signal (ECG) for the same purpose (key distribution for the IMD system). To provide guidance to researchers, this section conducts a comparative analysis of these two schemes to identify their similarities and differences, and contrast their relative merits and demerits.

In this section, we compare their models step by step. According to our analysis, we find that both techniques follow common workflows to conceal a symmetric key in a transmitter and reveal the key in a receiver, respectively. On the other hand, the FuzComm scheme has a more complicated process in obtaining ECG measurements than the FuzVault scheme. By contrast, its key concealing and revealing process is much simpler when compared to the FuzComm scheme. Besides a superior FAR performance of the FuzComm scheme, their FRR performance is comparable. Since the polynomial calculation and reconstruction are utilized in the fuzzy vault scheme, from the perspective of computational complexity, the FuzComm scheme is recommended for lightweight WBAN sensors.

4.4.1 Fuzzy Commitment Modeling

The fuzzy commitment primitive, proposed by Juels and Wattenberg [51], has two phases: the commitment phase at the transmitter side and the decommitment phase at

the receiver side, as shown in Fig. 4.3. The secret key, K, is bound and hidden in the commitment, $K_{encrypt}$, in the commitment phase and revealed in the decommitment phase. A hash value of the key, K, is calculated in the commitment phase and sent to the receiver, so that the receiver can check the correctness of the decommitted key. The mathematical model of the primitive is described in Chapter 3 Section 3.3.

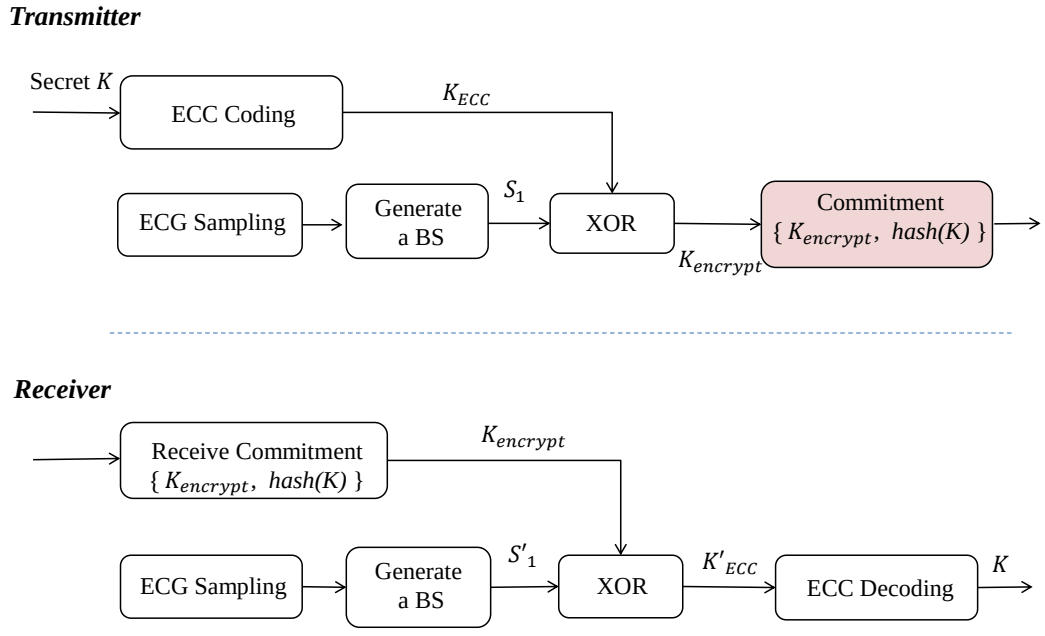


Figure 4.3: Block diagram of the fuzzy commitment scheme.

4.4.2 Common Workflows

After analyzing schemes of the fuzzy commitment and the fuzzy vault, their common workflows are identified and shown in Fig. 4.4. Both the techniques extract measurement values from ECG signals and then use these measurements to conceal/reveal the key. Their general workflows can be described as follows.

(a) Transmitter

The transmitter sends an SYN message to the receiver in order to synchronize the ECG sampling; proper measurements are then extracted from sampled ECG signals for concealing the key. Before the key concealment process, the key is encoded by the ECC in order to correct potential bit errors from the wireless channel and the ECG measurements. The hash value of the key, $hash(K)$, is also computed and included in the commitment message. Finally, the commitment message with the key information is sent into the public channel.

(b) Receiver

The receiver, after receiving the ECG SYN message from the transmitter, samples the ECG signal synchronously, and extracts proper ECG measurement values. When receiving the commitment message from the transmitter, it reveals the concealed key from the commitment using its own ECG measurements, resulting in K'_{ECC}. It then goes through the ECC decoding process to correct bit errors, obtaining K'. At this point, the receiver checks whether $hash(K')$ equals the received $hash(K)$. If they are equal, the revealed key, k', is correct; otherwise, the receiver requires a re-transmission from the transmitter.

4.4.3 Comparative Analysis

Although both the techniques follow common workflows, as shown in Fig. 4.4, there are several differences between them. They both could use the same technique to do ECG sampling, but the algorithms used to process sampled ECG signals are different; likewise, they could use the same kind of codes of ECC, e.g. BCH codes, but the error correcting capabilities required by the two fuzzy schemes are different. Their differences are highlighted as below.

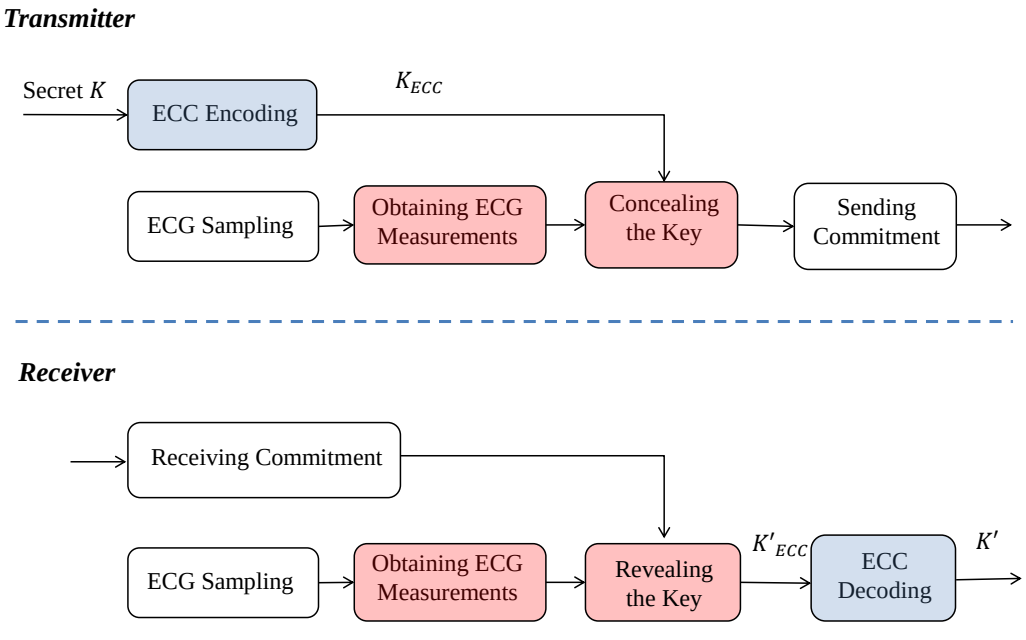


Figure 4.4: Common workflows of the fuzzy commitment scheme and the fuzzy vault scheme.

(a) Obtaining ECG Measurements

As analyzed in the ECG modelling, IPI values, IPI_i, are detected from ECG signals and then processed by different algorithms for these two fuzzy schemes. In the fuzzy vault scheme, with detected consecutive IPIs, the transmitter establishes a set $A = \{a_i\} = \{IPI_i | i = 1, 2, \cdots\}$ while the receiver establishes a set $B = \{b_i\} = \{IPI_j | j = 1, 2, \cdots\}$ with its own synchronously measured ECG signal. Meanwhile, in the fuzzy commitment scheme, random binary bits from each IPI are extracted and concatenated in order to form BSs using the algorithms proposed in [35, 25]. Both the transmitter and the receiver generate a BS, S_1 and S'_1, respectively using their own detected ECG signals.

(b) ECC Encoding Process

The requirement of the error correcting capability is an important aspect when choosing one kind of ECC. According to their mathematical models, wireless channel noise affects both of them; besides, the major part of bit errors in the fuzzy commitment scheme are introduced by the error bits between S_1 and S_1'. In the fuzzy vault scheme, some error bits in the key may also be caused by errors of polynomial reconstruction. However, according to the analysis in [35, 36], the error bits in the fuzzy commitment scheme are much larger than those in the fuzzy vault scheme, and have to be corrected using one kind of ECC in each communication session. On the other hand, when designing the fuzzy vault scheme, since bit errors of the key from the wireless channel noise and the polynomial reconstruction may not happen in each session, we can choose not to include the ECC encoding and decoding processes in the scheme but request re-transmission of the commitment message upon detecting errors in the key.

(c) Concealing the Key

The key concealment is critical in both the schemes. In the fuzzy vault scheme, as shown in Fig. 4.1, this block includes processes as polynomial construction, projection, adding chaff points; meanwhile, in the fuzzy commitment scheme, as shown in Fig. 4.3, it is an XOR operation between K_{ECC} and S_1. At this step, the fuzzy vault scheme constructs a vth order polynomial where the K_{ECC} is embedded into its coefficients; then it projects points within the set A onto the polynomial by calculating their polynomial values; reasonable chaff points are calculated and added into the vault in order to conceal genuine points. Clearly at this step, the complexity of the process within the fuzzy commitment scheme is much simpler than those within the fuzzy vault.

(d) Revealing the Key

The key revealment is another critical step in both the schemes within the receiver. In the fuzzy vault scheme, this step includes matching and polynomial reconstruction, as shown in Fig. 4.1. In the matching process, it compares points in its own generated set B of IPIs and x-values of received points in the vault and obtains the genuine points on the polynomial; it then reconstructs the polynomial using these genuine points. After that, it concatenates coefficients of the polynomial and reveals the key. By contrast, the fuzzy commitment scheme reveals the key by a really simple XOR operation between the received $K_{encrypt}$ and its own generated BS S_1'; however, the revealed key, K'_{ECC}, has bit errors and has to go through the ECC decoding process.

(e) Computation Overhead

The computation workload in each step of these two schemes is different from each other. The main computation in the fuzzy vault scheme is in processes of the key concealment in the transmitter and the key revealment in the receiver where the polynomial is calculated and then re-constructed. By contrast, these two steps in the fuzzy commitment scheme are as simple as performing an XOR operation; however, the fuzzy commitment scheme has to generate two random BSs with high matching performance at two ends respectively. The algorithm proposed in [35] includes processes such as accumulation and modulo, contraction mapping and Gray coding.

We also find from the simulation study that the polynomial calculation and reconstruction in the fuzzy vault scheme consumes resources. As discussed in [47,81], the order of the polynomial can vary from 6 to 12. We take the lowest order polynomial to distribute a 56-bit key for example. The equation of the polynomial, $p(x)$, is denoted by $p(x) = c_6x^6 + c_5x^5 + \cdots + c_1x + c_0$ in which the coefficients are concatenated to form the 56-bit key $K = c_6|c_5|\cdots|c_0$. So, each coefficient, c_i, is 8-bit long. For a normal sinus

rhythm with a heart rate of 60-100 beats per minute, IPI values vary from 600ms to 1000ms. So, for the term with the highest degree, c^6x^6, its value varies from 2^{62} to 2^{68}. Even if we use its variation part, from 0 to 400ms, the calculation result of c^6x^6 may be still up to 2^{62}. Therefore, calculating the polynomial consumes resources and there may be a risk of a buffer overflow to the lightweight wireless sensors.

(f) Communication Overhead

In the fuzzy commitment scheme, the main data sent into the public channel contains a key hash value $hash(K)$ and an encrypted key, $K_{encrypt}$, while that in the fuzzy vault has $hash(K)$ and a vault with a large number of chaff points. For example, the PSKA algorithm [47] implements experiments with vault sizes varying from 300 to 5000. Therefore, the communication overhead in the fuzzy vault scheme is larger than that in the fuzzy commitment scheme.

Finally both the schemes check the correctness of the revealed key, K', by comparing its hash value, $hash(K')$ with the received $hash(K)$. If they are equal, the key is accepted as legitimate; otherwise, a request to re-do the whole key distribution process is to be sent to the transmitter.

4.4.4 Results of Comparison

This section has compared two kinds of widely investigated ECG-based key distribution schemes within a WBAN, namely the fuzzy vault scheme and the fuzzy commitment scheme. After the comparative analysis, we extract their common workflows and find the difference of each step within the flows. We summarize our findings below:

1. These two fuzzy schemes use the same physiological signal (ECG) for the same purpose (symmetric key distribution) within the WBAN; they follow common workflows to conceal a key in the transmitter and reveal the key in the receiver, respectively.

2. At the step of obtaining ECG measurements, the fuzzy commitment scheme has to generate two random binary sequences with high matching performance from ECG IPI values while the fuzzy vault scheme uses ECG IPIs directly to project a set of points on the polynomial. Therefore, the fuzzy commitment scheme has a more complicated process than the fuzzy vault scheme when obtaining the ECG measurements.

3. In the fuzzy commitment scheme, the ECC process is essential in order to correct error bits between two generated binary sequences, while in the fuzzy vault scheme, since errors may not always happen in the wireless channel transmission and the polynomial reconstruction process, this process may be omitted but requires a re-do of the whole commitment process when errors are detected in the revealed key.

4. The key concealment in the fuzzy vault scheme includes polynomial construction, point projection and adding a large number of chaff points into the vault, which is obviously more complicated than a simple XOR operation within the fuzzy commitment scheme.

5. Similar to the key concealment, its inverse, the key revealment, in the fuzzy vault scheme is more complicated than that in the fuzzy commitment scheme.

4.5 Summary

In this chapter, we have presented a fuzzy vault based key distribution scheme for the IMD security. It utilizes ECG IPI values to hide/reveal the key in wireless transmission. It solves the unique challenge of *Security vs. Accessibility* in the IMD security design. In addition, an improvement of using hashing functions instead of chaff points is proposed for the FuzVault scheme. The performance analysis shows that this scheme meets our

design goals of security.

This chapter also conducts a comparative analysis between key distribution schemes using the fuzzy commitment primitive and the fuzzy vault primitive. These two schemes follow common workflows to conceal a key in the transmitter and reveal the key in the receiver, respectively. However, the key concealing and revealing processes (XOR operations) in the fuzzy commitment scheme are much simpler than those in the fuzzy vault scheme (polynomial calculation and reconstruction). From the perspective of the computational complexity, the fuzzy commitment scheme is recommended for lightweight IMDs.

Chapter 5

Encryption for IMDs Using Modified One-Time Pads

In this chapter, we present an ECG based Data Encryption (EDE) scheme for IMDs. The EDE is designed with the ability to provide information-theoretically unbreakable encryption where two well-known techniques of classic One-Time Pads (OTPs) and Error Correcting Codes are combined to achieve a cryptographic primitive for IMDs.

The EDE scheme is an extension of our previous work in Chapters 3 & 4 which have focused on the ECG-based key distribution between the IMD and the programmer [48,82]. IMDs normally perform therapeutic or even life-saving functions for patients; attacks to IMDs could cause fatal consequences. That is why IMDs have to be rigorously protected from adversaries. Considering that OTPs, as proven by Shannon [76], are information-theoretically secure, the EDE scheme using modified OTPs can rigorously protect IMDs from adversaries whose aim is to decipher messages to compromise secrecy.

Unlike other ECG-based key agreement schemes where ECG features are used to facilitate key distribution, in the EDE scheme, random binary strings generated from ECG signals are directly used as keys for encryption. OTP keys are generated by the IMD and the programmer, respectively, before each encryption attempt; thus the EDE scheme does

not require a cryptographic infrastructure to support key distribution, storage, revocation and refreshment. The IMD is protected by the EDE scheme and therefore it cannot be accessed by the adversaries. However, medical personnel can gain access to the IMD by measuring real-time ECG data in emergencies. Therefore, the EDE design achieves a balance of high security and high accessibility for the IMD. Our data and security analysis shows that the EDE is a viable scheme for protecting IMDs.

The EDE scheme is based on *physiological signal-based OTPs* which uses binary strings generated from ECG as keys for direct encryption. OTPs were widely used for covert communications by intelligence agencies during the World War II and the Cold War [83]. Recently the OTP concept was applied in quantum [84] and optical scattering [85] based cryptography. Recent research on ECG-based key agreement, proposed in [81, 86, 47, 87], establishes a symmetric random key between two sensors where ECG signals are used to conceal the key in distribution. Unlike these schemes, security keys in our scheme are generated from ECG signals and are used to encrypt secret data directly.

The organization of this chapter is as follows.

- In Section 5.1, an architecture of the EDE scheme is presented where the IMD and the programmer are required to measure ECG signals in order to generate keys for encryption.

- The EDE scheme is designed in detail in Sections 5.2 & 5.3, including linear ECC, modification to the OTPs, and the EDE communication protocols.

- We evaluate the scheme in Section 5.4 in terms of OTP key randomness, temporal variance and distinctive, FAR/FRR performance and overhead.

- The security analysis of the proposed EDE scheme is analyzed from two perspectives in Section 5.5: requirements of OTP keys and security of the modified OTP algorithm.

5.1 EDE Scheme Architecture

The EDE scheme includes two components: an IMD and an external programmer. The IMD is an electronic device which is implanted in the body to assist and/or monitor a patient's health, while the programmer is an outside device which has the ability to access data in the IMD and program it wirelessly. Currently most IMDs have the capability of measuring ECG signals [2]. In our scheme, an ECG sensor is connected to the programmer and measures ECG signals from, for example, the wrist of the patient, as shown in Fig.5.1. It is convenient to add an ECG measuring function into the programmer since it is an outside device and is normally kept in a hospital.

The scheme is depicted in Fig.5.1. It can be seen that the IMD and the programmer measure ECG signals synchronously and random binary key sets, $K_A = \{k_{ai}\}$ and $K_B = \{k_{bi}\}$, are then generated by each device, respectively. K_A is used to encrypt secret data with modified OTPs in the IMD while K_B is used to decrypt the ciphertext. Since there

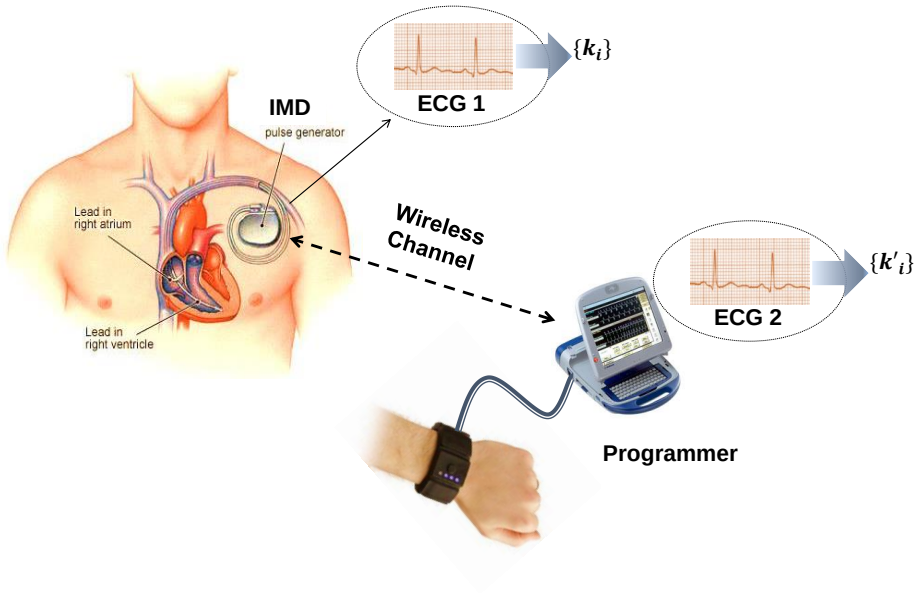


Figure 5.1: Secure communications using the EDE scheme.

can be a mismatch between K_A and K_B due to bit errors of the keys, error correcting codes (ECC) are employed to correct these errors within the decrypted message.

One key feature of the EDE is that the keys are independently generated by each device. The EDE does not require key distribution or transmission from one sensor to another. Key refreshment can be easily achieved by generating keys at two sensors directly. Also there is no need of key storage and revocation, since a fresh pair of keys, k_{ai} and k_{bi}, will be generated before each new encryption cycle and will not be re-used according to OTP rules.

Another key feature is that the EDE scheme inherits the property of perfect secrecy from OTPs, and can provide information-theoretically secure encryption for IMDs. As IMDs normally perform therapeutic or life-saving functions, this feature is critical to IMD security. Inherent characteristics of ECG bit strings of randomness, temporal variance and distinctiveness ensure that OPT keys cannot be probed, duplicated or speculated without a physical contact with the patient's body.

5.2 ECG-based Data Encryption Scheme

One-Time Pads (OTPs), although acknowledged as mathematically unbreakable, have limited applications in the modern computing era [84, 85]. This is because the OTPs require the storage of a large number of random keys and guarantee that no keys are re-used. The EDE scheme applies a practical and secure approximation of OTPs for the IMD system where the OTP keys are generated by the sender and the receiver respectively and synchronously. This section presents the EDE scheme in detail. We first design a modified OTP algorithm for the purpose of IMD encryption and then propose a protocol which executes the EDE scheme using this algorithm.

5.2.1 Linear Error-Correcting Codes

We design a system with a secret s in the secret space S, an encryption algorithm F_{Enc} and its corresponding decryption pair F_{Dec}. Considering that there can be mismatch between K_A and K_B, our designed EDE algorithm has to satisfy the following:

Definition 1. *An encryption/decryption pair* (F_{Enc}, F_{Dec}) *with parameters* (S, K_A, K_B) *is complete with* $\epsilon-$*error tolerance when the following condition holds: for each* $s_i \in S$ *and each key pair* (k_{ai}, k_{bi}) *of* (K_A, K_B) *where* $|k_{ai} - k_{bi}| \leq \epsilon$*, the decryption process* $F_{Dec}(k_{bi}, F_{Enc}(s_i, k_{ai})) = s_i$ *is a success with an overwhelming probability.*

This requires that the EDE scheme has the capability to correct errors caused by the key pair mismatch between (k_{ai}, k_{bi}). Here the overwhelming probability is that it is larger than $1 - \varepsilon$ for certain negligible value ε. Borrowing design ideas from the area of fuzzy vault [50], fuzzy commitment [51] and fuzzy extractor [88], an Error Correcting Code (ECC) is introduced into the scheme which can help to implement the protocol security.

We denote a binary linear ECC (n_e, k_e, t_e) with error correcting capability t_e, where n_e is the length of the codeword C_e and k_e the message length. The ECC encoding function, $encoderECC(.)$, maps the message $s \in \{0,1\}^{k_e}$ into its codeword $s_e \in \{0,1\}^{n_e}$. According to the ECC linear property an XOR operation of any two codewords leads to another codeword within the same codeword set. Here we use the Hamming distance to measure the distance of codewords, denoted by $d_H\{.,.\}$ and the Hamming weight is denoted by $\|.\|$. For an ECC with error correcting capability t_e, the minimum distance of codewords is $d_H = 2t_e + 1$.

Given an ECC encoded message $s_e \in \{0,1\}^{n_e}$, the minimum distance to any codeword $c_e \in C_e$ is defined as $d_{min}(s_e, c_e) = min_{c_e \in C_e} d_H(s_e, c_e)$. If $d_{min}(s_e, c_e) \leq t_e$ the ECC decoding function, $decoderECC(.)$, returns the message corresponding to the closest codeword within C_e. Otherwise, this encoded message is not decodable. A classic class of cyclic ECC called BCH codes is chosen in our design. The principal advantage of BCH codes is

that they can be decoded with a small and low-powered electronic hardware [75], such as the wireless sensor nodes.

5.2.2 Modified One-Time Pad Algorithm

For classical OTPs working over a secret s_i in the secret space S, a corresponding key k_i in the key space K, the resulted cryptogram c_i in the cryptogram space C is denoted by $c_i = f(s_i, k_i) = s_i \oplus k_i$, where f is a function with a unique inverse f^{-1} and $\oplus$ is an XOR operation which mixes each bit of s_i with each bit of k_i. Thereafter, c_i is to be sent through a public channel. At the receiving end, the same OTP key k_i is applied to decrypt the secret s_i by $s_i = f^{-1}(c_i, k_i) = c_i \oplus k_i$. For a series of secret messages $S = \{s_1, s_2, \cdots\}$, the corresponding cryptogram is denoted by $M_c = F(S) = \{f(s_1, k_1), f(s_2, k_2), \cdots\}$ while its decryption process is denoted by $S = F^{-1}(M_c) = \{f^{-1}(c_1, k_1), f^{-1}(c_2, k_2), \cdots\}$. Here F denotes the implementation of f on each element of S while F^{-1} is its inverse f^{-1} on each element of M_c. OTPs become unbreakable only when the keys are kept secret, truly random, never re-used in whole or part and have the same length as the message.

Because of the uncertainty of physiological (ECG) signals, the requirement in Definition 1 has to be satisfied. Thus, the classical OTPs are modified as follows: (a) *Encryption process* F_{Enc}. A series of secrets S are mapped to ECC codewords S_e by $S_e = encoderECC(S)$ at the beginning where redundant information is added to correct errors caused by key bit mismatches. Then OTP operations are performed on S_e to encrypt the secrets by the function $M_c = F_{Enc}(S, K_A) = F(S_e, K_A)$. (b) *Decryption process* F_{Dec}. The cryptogram M_c is decrypted by K_B by $S'_e = F^{-1}(M_c, K_B)$. S'_e is slightly different from S_e due to bit errors between K_A and K_B. In order to correct these error bits, ECC decoding process is performed by $S' = decoderECC(S'_e)$ where S' is the output of the modified OTPs. In order to ensure that S' equals S, the hash value of S, $hash(S)$, is sent to the receiver along with the cryptogram M_c. Hence the receiver computes the

hash value $hash(S')$ with the same hash function as the sender and compares it with the received $hash(S)$. The modified OTP scheme succeeds if they are equal; otherwise it fails and the output S' is discarded. Within this modified OTP scheme, the length of the key has to be as long as the ECC codeword, not the secret, since the ECC codeword is XORed with the key.

Lemma 1. *For $\forall k_{bi} \in K_B$, the decryption process F_{Dec} succeeds when the number of key bit errors is less than or equal to ECC error correction capability, denoted by $|k_{ai} - k_{bi}| \leq t_e$. The largest error tolerance ϵ of the scheme equals the ECC error correction capability t_e.*

Proof. In the i_{th} OTP encryption, the cryptogram $m_{ci} = F_{Enc}(s_i, k_{ai}) = s_{ei} \oplus k_{ai}$, where $s_{ei} = encoderECC(s_i)$ is the ECC encoding output. In the decryption process, $s'_{ei} = f^{-1}(m_{ci}, k_{bi}) = m_{ci} \oplus k_{bi} = (s_{ei} \oplus k_{ai}) \oplus k_{bi}$. The output s'_{ei} is different from s_{ei} due the mismatch between k_{ai} and k'_{bi}. According to the condition of the Lemma 1 $|k_{ai} - k_{bi}| \leq t_e$, we can obtain $|s_{ei} - s'_{ei}| \leq t_e$. Hence the ECC has the capability to correct bit errors within s'_e and generate the secret s_i by the ECC decoding function $decoderECC(.)$. The ECC can only correct up to t_e error bits, thus the largest error tolerance ϵ is decided by the ECC error correction capability t_e. □

Fig. 5.2 depicts a comparison of the classical OTPs and the EDE scheme within which the technique of OTPs is combined with the ECC to fulfill the transmission of the secret s_i. The three subfigures are discribed as below: (a) The theoretically perfect OTP mixes a piece of secret data s_i with a random key k_i to generate a ciphertext c_i; (b) In the EDE, the OTP keys are generated from ECG signals; (c) Error Correcting Codes (ECC) are employed in order to correct errors caused by key error bits. In classical OTPs, key pre-distribution is critical but involves high risk. The same copy of a key set has to be distributed securely to the sender as well as the receiver for successful decryption. However, the EDE scheme generates keys by extracting binary strings from ECG signals directly, as shown in Fig. 5.2 part (c). The sender (IMD) and the receiver (programmer)

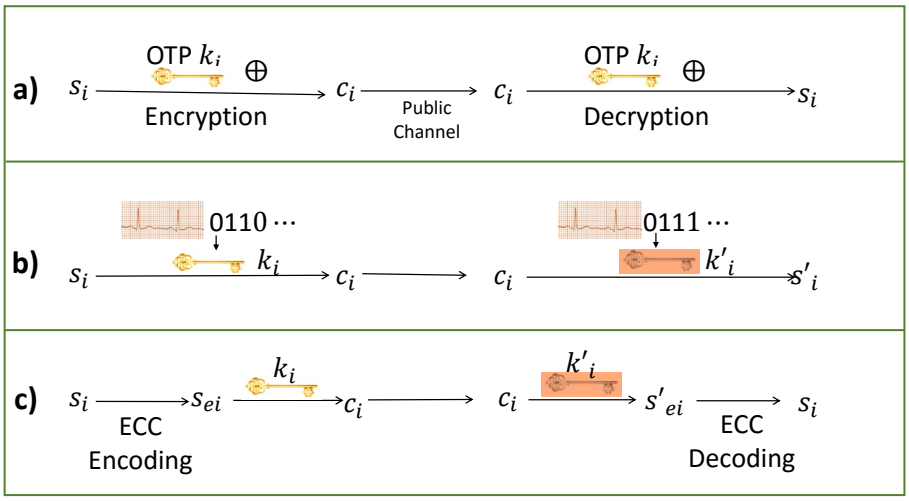


Figure 5.2: The modified One-Time Pad (OTP) protocol using ECG signal.

generate binary strings, k_{ai} and k_{bi}, from simultaneously measured ECG signal; thus the EDE scheme does not require key pre-deployment or transmission.

5.3 Communication Protocol Design

After a programmer initiates a communication session with an IMD, a protocol implementing the EDE scheme is described as follows.

5.3.1 ECG Binary String Generation

The programmer sends a synchronization request to the IMD for sampling ECG which indicates the sampling start-time T_{start} with a timestamp in the frame. Since there would be a timing difference between two clocks residing in the IMD and the programmer, these two clocks is synchronized by following the IEEE 1588 standard [89]. In the EDE scheme, the programmer is selected as a master while the IMD is a slave. In the synchronization frame, the programmer indicates its current time $T_{current}$ and sends $T_{current}$ to the IMD.

Since the IMD is very close to the programmer (less than $1m$), the transmission time of this frame is negligible. Therefore, the IMD uses $T_{current}$ to correct its clock. After the clock synchronization, the IMD and the programmer sample ECG signals synchronously at the time T_{start}. Two highly matched and random ECG binary strings, k_{ai} and k_{bi}, are then generated by the IMD and the programmer respectively, using the aforementioned algorithm. We do not require key pre-distribution or key transmission here as keys will be generated by each device independently.

5.3.2 Process in the IMD

After generating k_{ai}, the process executed in the IMD is shown in Algorithm 1. Firstly the secret s_i is encoded by an ECC encoding process, $encoderECC(.)$, to create s_{ei} in which redundant information is added for error correction purposes. Then the cryptogram c_i is created by an XOR operation. A $hash$ value is computed by a one way hash function $h(.)$ in order to check message integrity and the correctness of decoded s'_i at the programmer. The input of this hash function is combined by four parameters, which obviously has a long bit length. So, it has a large search space for the potential attacker who may launch a dictionary attack to it. A fresh random number generated by a counter, $nonce$, is used as a session identifier to prevent potential replay attacks. A message, msg, includes id_{imd} and id_{pro}, identity numbers of the IMD and the programmer respectively. The message

along with the hash value is then sent to the programmer through a public channel.

Algorithm 1: Process in the IMD

Input : the secret s_i and the key k_{ai}

Output: a message, msg, sent to the public channel

$s_{ei} = encoderECC(s_i)$;

$c_i = s_{ei} \oplus k_{ai}$;

$hash = h(id_{imd}|nonce|c_i|s_i)$;

$msg = (id_{imd}, id_{pro}, nonce, c_i, hash)$;

$startSession(msg)$;

5.3.3 Process in the programmer

After receiving the message, msg, the process in the programmer is shown in Algorithm 2. The received message msg' might include noise from potential channel interference that can be introduced accidentally due to overlapping channels or deliberately by an adversary. With generated key k_{bi}, the process is as follows. (a) The programmer decrypts c'_i by an XOR operation with k_{bi}, resulting s'_{ei} which could be different from s_{ei} due to the mismatch between k_{ai} and k_{bi} and/or wireless channel noise. (b) An ECC decoding process, $decoderECC(.)$, is then performed to correct error bits between s'_{ei} and s_{ei}, resulting in s'_i. (c) $hash_{pro}$ is computed with the same hash function as that in the IMD and compared with the received $hash'$ so as to check both the integrity of received msg' and the correctness of decoded s'_i. If $hash_{pro}$ equals $hash'$, the received msg is not modified in transmission and the obtained s'_i is the same as the secret s_i; a 'success' code is then assigned to the acknowledgement ack; otherwise ack is assigned a 'failure' code. The programmer finally sends ack to the IMD to confirm the decryption results.

Similarly, the encryption protocol for messages from the programmer to the IMD can be constructed as follows: the programmer initiates and synchronizes the communication session with the IMD at the beginning, which generates two random ECG BSs respectively.

Thereafter, the programmer follows a process similar to that in Algorithm 1 to encrypt the message while the IMD performs decryption using a process similar to that in Algorithm 2. The *ack* message is sent back in each communication session to inform the programmer of the decryption result of each message.

Algorithm 2: Process in the programmer

Input : the key k_{bi} and msg' received from the public channel

Output: the decrypted s'_i and acknowledgement *ack*

$msg' = (id'_{imd}, id_{pro}, nonce', c'_i, hash')$;

$s'_{ei} = c'_i \oplus k_{bi}$;

$s'_i = decoderECC(s'_{ei})$;

$hash_{pro} = h(id'_{imd}|nonce'|c'_i|s'_i)$;

if $hash_{pro} = hash'$ **then**

    $s'_i = s_i$;

    $ack \leftarrow success$;

else

    $ack \leftarrow failure$;

end

Send *ack* to the IMD ;

5.4 Scheme Evaluation

In this section, we provide an evaluation of the EDE scheme by performing a series of experiments. Lacking the ability to obtain IPI measurements from IMDs in the lab, we follow a similar analysis as in [81, 86, 47, 87] and generate OTP keys by using the ECG data from the MIT PhysioBank database (http://www.physionet.org/physiobank). Experiments were carried out on the ECG data from 167 subjects: 18 subjects (128Hz, 5 men and 13 women) from the MIT-BIH Normal Sinus Rhythm (NSRDB) [70], 79 subjects

(250Hz, 466Mbit) from European ST-T (EDB) [78], 47 subjects (360Hz, 107Mbit) from MIT-BIH Arrhythmia (MITDB) [70] and 23 subjects (250Hz, 607Mbit) from MIT-BIH Atrial Fibrillation (AFDB) [70]. Considering potential applications to pacemakers or ICDs, the last two databases (MITDB and AFDB) contain arrhythmia ECG signals.

In the experiment, we measure ECG signals from two different part of the same subject. One ECG signal is measured by the IMD from where the device is implanted, such as the chest for the pacemaker and the ICD. Another ECG signal is measured by the programmer. In our design, an ECG sensor is designed for the programmer and can be used to measure ECG signals from the wrist of the patient. Here, we suggest the programmer to measure the ECG from the wrist, as it is convenient and common in clinics.

5.4.1 OTP Key Randomness and Temporal Variance

A basic requirement of using the OTPs is that keys used for encryption should be random. The ECG BS generation algorithm in Section 3.4 is used here. Randomness analysis of these ECG BSes is performed in Section 3.5, including randomness of ECG IPIs, entropy of ECG BSes and a statistic NIST randomness test. Thus, we assure that these generated ECG BSes incorporate the characteristic of randomness.

We evaluated the generated ECG binary strings for temporal variance to ensure that the encrypted secret cannot be decrypted by the same subject's historical or future ECG signals. In the experiment, we sampled ECG signals on each subject from the MIT-BIH NSRDB over 300 random start-times and computed the average Hamming distance between k_{ai} and k_{bi}. Fig.5.3 shows an experiment result from one subject. The x-axis represents k_{ai} of all samples while y-axis represents k_{bi}. Colors represent the range within which the actual Hamming distance falls. The higher values are in red while the lower values are in blue.

We can see from Fig.5.3 that the Hamming distance values between k_i and k_i' generated

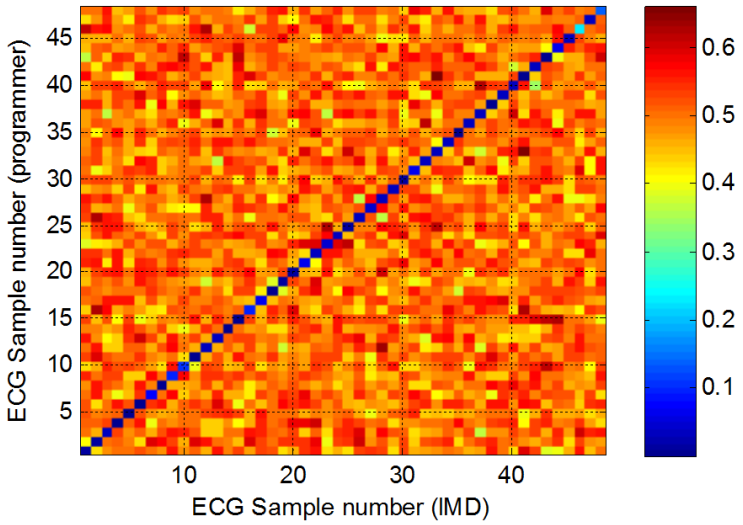


Figure 5.3: Hamming distance between two ECG binary strings generated from two different body parts of the same subject.

at two different start-times are quite high, with the average distance of about 49.72% (about 63 bits). This bit error rate was much higher than the error correcting capability of BCH codes; thus the encrypted secret could not be decrypted by using ECG signals from the same subject measured at a different start-time. Meanwhile, obtaining correct ECG binary strings via brute-force attack was also impossible in a realistic setting. This is because the adversary has no knowledge as to which of the 127 bits are different; obtaining the correct ECG bit string via brute-force would require $\binom{127}{63}$ attempts which is equivalent to launching a brute force attack on a 127-bit long secret.

We can also see from Fig.5.3 that all the diagonal values are very small (less than 10%), which means that the Hamming distance between k_{ai} and k_{bi} generated at the same start-time are quite low; therefore, the error bits in the decrypted secret could be corrected by proper ECC.

5.4.2 OTP Key Distinctiveness

The property of distinctiveness is to ensure that the secret encrypted by an IMD implanted in one subject cannot be decrypted by another programmer using ECG signals from another subject (either accidentally or maliciously). This property enables us to distinguish IMD systems on different subjects. In the experiment, we sampled ECG signals on each subject from the MIT-BIH NSRDB over 300 random start-times and computed the average Hamming distance between two ECG binary strings from different subjects. The average distance was 49.99% (about 63 bits) which is similar to that for temporal variance shown above. This result shows that the secret encrypted by an IMD using ECG signals from one subject cannot be decrypted by another programmer using another subject's ECG signals. This can prevent attackers from decrypting secrets using a different subject's ECG data.

5.4.3 FAR/FRR Analysis

FRR and FAR are two critical parameters to be taken into consideration when evaluating any biometric-based security schemes. In our experiment, FRR is the measure of the likelihood that a programmer fails to decrypt a secret from an IMD by using simultaneously measured ECG signals from the same subject, while FAR is the measure of the likelihood that a programmer could decrypt a secret from an IMD by using the same subject's historical or future ECG data or data from another subject. Considering that error correction codes are employed, FRR and FAR will vary according to BCH codes' error correction capability. Fig.5.4 shows experiment results of FRR and FAR on each ECG database with BCH code length $n = 127$.

We observe from Fig.5.4 that results of FAR tests on all ECG databases are zero, which means the encrypted secret could not be decrypted by either the same subject's historical or future ECG data or ECG measured from other subjects. These results are

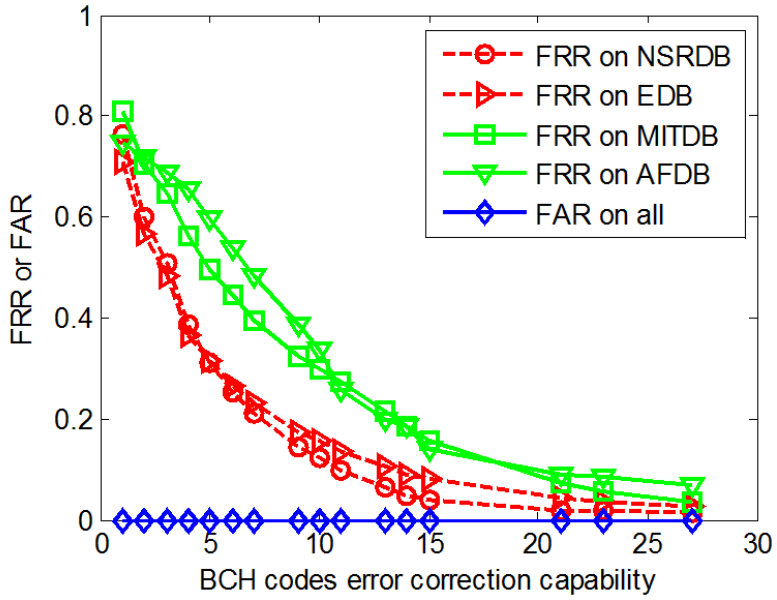


Figure 5.4: FRR and FAR vary versus BCH codes error correction capability.

consistent with the analysis of temporal variance and distinctiveness. We also see from Fig.5.4 that FRR declines dramatically when the error correction capability, t, increases, ending at around 5%. We note that the performance on databases of sinus rhythm ECG (NSRDB and EDB) is better than that on arrhythmia databases (MITDB and AFDB). However, this scheme could be applied for arrhythmia patients as long as QRS peaks of ECG signals can be measured correctly. Generally, from our observation of experimental results, the more accurate the measured value of the ECG IPI is, the better the FRR performance.

5.4.4 Overhead Analysis

Communication overhead is negligible in the EDE as the ciphertext sent into the channel is of the same length as the codeword of BCH codes. Adding a large number of chaff points to hide the data [47, 81] is not needed here as the secret data is already encrypted. The main concern of computation overhead can be due to the security related

processes supported within the IMD since this device is battery powered and implanted in the body. The programmer, as an external device in the hospital or clinics, could be easily designed with hardware capable of supporting intensive computational overheads. So, we focus on overhead analysis on the IMD.

In the IMD, processes include ECG binary string generation, BCH encoding, hash function and encryption. The encoding process of cyclic BCH codes could be quickly processed by linear feedback shift registers (LFSR). As presented in [2], SHA-1 hash process is around 4ms on the platform of TelosB with TinyOS 2.1. The encryption process is a simple XOR operation. As discussed in [2, 35] ECG detection and IPI calculations could be assumed as basic functions for IMDs (e.g., pacemakers).

In the ECG BS generation algorithm, SMA values of m consecutive IPIs are calculated first; since the window size w is a constant, its order of the computational complexity is $O(m)$. For Gray coding, if a binary SMA_i is $(b_{q-1}, b_{q-2}, \cdots, b_0)_2$, the corresponding Gray code $(g_{q-1}, g_{q-2}, \cdots, g_0)_2$ is calculated by,

$$\begin{cases} g_{q-1} = b_{q-1} \\ g_i = b_{i+1} \oplus b_i, i = 0, 1, \cdots, q-2. \end{cases} \tag{5.1}$$

It could be seen that Gray coding needs $(q-1)$ bitwise addition operations for each SMA_i. Compared to an addition operation, the bitwise operation is slightly faster; thus its complexity can be assumed as a fraction of an addition, denoted by $\gamma(0 < \gamma < 1)$. Thus the total number of bitwise addition operations for Gray coding is $O(\gamma(q-1)(m-w+1)) = O(m)$. The complexity of LSB removal could be easily completed by reading bits except the LSB. Therefore, the order of the computational complexity of the BS generation algorithm is $O(m)$, which shows that it only adds a little complexity to the IMD.

The energy overhead due to communications is another important requirement when designing the EDE scheme. As discussed in [90], for a CrossBow MICA2DOT mote using a Chipcon CC1000 radio, receiving and transmitting one byte message consume $28.6\mu J$

and $59.2\mu J$, respectively. For the IMD transmitting and receiving an N-byte message, the energy consumption is $59.2N\mu J$ and $28.6N\mu J$, respectively. Compared with the fuzzy vault-based scheme [47,81], our EDE scheme does not require any chaff points to be added into the message; therefore, the energy consumption is much lower than the fuzzy vault-based scheme.

5.5 Security Analysis

The security of OTPs relies predominantly on their key management capabilities; the EDE communication protocols supporting OTPs also play a key role in security. This section discusses how the EDE scheme obeys rules of key management in OTPs, the property of perfect secrecy and protocol security.

5.5.1 Requirements of OTP Keys

According to Shannon's analysis [76], the protocol of OTPs is information-theoretically unbreakable only if properly applied. The rules for OTP keys are that: (a) the key is as long as the secret message; (b) the key is truly random; (c) each key is used only once and (d) the key is destroyed immediately after use. Requirements of (a) and (b) have been achieved based on previous analysis; the EDE scheme will dispose of each key after use which is in line with the requirement (d).

Requirement (c) is to make sure that each key will not be re-used. Traditional OTPs use a small note pad to print a large set of random keys, and use a new key in each operation. In the computing era, this rule requires the storage of a large number of random keys, checking a key has been used before or not, e.g. OTPs applied in quantum [84] or optical scattering [85] based cryptography. However, this is not practical in the IMD settings since the IMD is a tiny device implanted in the human body having limited

resources such as memory, battery power, etc. Thus we do not require the IMD to save all keys for verification purposes.

The randomness of generated ECG binary strings guarantees that the use of same keys would hardly happen. Considering the error correction capability of the ECC, e.g. $BCHcode(n, m, t)$, two BSs with Hamming distance t could be regarded as the same key as they have the ability to decrypt each other's ciphertext. The OTP key length is equal to the code length n. As bits in the key are purely random according to NIST tests, the probability of a similar key being generated as before can be modeled as an $(n-t)$-fold Bernoulli trial with probability $p = 0.5$, denoted by $B(n, p)$. So, the success probability of generating the same or similar key can be calculated by,

$$\begin{cases} f(k; n, p) = \binom{n}{k} p^k (1-p)^{n-k} \\ k = n - t \end{cases} \tag{5.2}$$

For a BCH code(127,64,10) the success probability is 1.23×10^{-24}, which is negligible and could be assumed as zero in a practical test. Therefore we can assure that keys in the EDE scheme will not be re-used. So, adversaries cannot obtain any sensitive information to attack the scheme using statistical analysis or pattern matching. Consequently, all the requirements of OTP keys are fulfilled in the EDE.

5.5.2 Scheme Security

(a) Perfect secrecy

The EDE scheme inherits the property of perfect secrecy from OTPs. For a secret message s with a priori probability $P(s)$, a posteriori probability, $P(s|c)$, of the secret if

cryptogram c is intercepted can be denoted by Bayes's theorem as,

$$P(s|c) = \frac{P(s)P(c|s)}{P(c)} \tag{5.3}$$

in which $P(c)$ is the probability of obtaining cryptogram c, and $P(c|s)$ is the conditional probability of c when the secret s is chosen. According to definition by Shannon [76], perfect secrecy is that a posteriori probability is equal to a priori probability independently of all values. So, intercepting the cryptogram gives no information to adversaries. In this scheme, it is required that $P(s|c) = P(s)$. As $P(s) \neq 0$, it is also required that $P(c|s) = P(c)$. In the EDE scheme, one OTP key is used for one encryption only and the keys are purely random by the NIST test. Furthermore, although the ECC is employed in the scheme, the analysis using Eq. 5.3 shows the probability of generating two closely related keys with the number of error bits within the ECC error correcting capability t being negligible; thus the probability of c given the secret s is equal to the probability of obtaining c in any case, which means $P(c|s) = P(c)$. In the EDE protocol design, a one way hash function is used to check message integrity and correctness of decrypted message, and the digests (hash values) could be intercepted by adversaries in the public channel. However, since the adversary could barely manage to invert the digest to obtain the message, this intercepted digest would not compromise the message secrecy as long as the length of the digest is no less than the message length, such as choosing SHA-1 for a 128-bit message. The preimage attack is hard to launch here considering the length of the digest and the length of inputs of the hash function, including $id_{imd}, nonce, c_i, s_i$. Therefore the EDE scheme satisfies the property of having perfect secrecy.

In contrast to conventional symmetric encryption, perfect secrecy makes the scheme immune even to brute-force attacks. For secret messages $\{s_1, s_2, s_3, \cdots\}$, the encrypted messages are $M_c = \{f(s_1, k_{a1}), f(s_2, k_{a2}), f(s_3, k_{a3}), \cdots\}$. As keys are purely random, guessing a secret in M_c (e.g. s_1) requires trying all possible keys. Even if the adversary obtains s_1 and k_{a1}, it cannot gain any information about the key needed to decrypt other

secrets in M_c due to the property of key randomness.

(b) Protocol security

The EDE scheme has the ability to protect IMDs from eavesdropping. The active adversaries which aim to obtain data from the IMDs can also be thwarted due to the presence of encryption. Compared to traditional OTPs, our proposed scheme has additional features as highlighted below. (a) This scheme provides *hash* values to verify message integrity and the correctness of decrypted secrets. Any modification of the message would be detected by the programmer as the modified message cannot verify the *hash* value. (b) The device ID, ID_{pro}, indicates which device the IMD intends to communicate with; the *nonce* maintains freshness of each session, and prevents potential replay attacks.

5.6 Summary

In this chapter, we have presented an information-theoretically secure encryption method for IMDs, namely the ECG-based Data Encryption (EDE). The EDE scheme uses physiological (ECG) signal-based OTPs to encrypt secret data from IMDs before transmission. OTP keys are to be generated by each device from synchronously measured ECG signals, respectively. As ECG signals are used as natural random input into the encryption algorithm, there is no need of a cryptographic infrastructure to support key distribution, storage, revocation and refreshment. We analyzed the performance of the scheme by using MIT PhysioBank ECG data, which showed that the EDE is a viable approach to secure IMDs from eavesdropper and active adversaries. The security analysis showed that the EDE scheme fulfills the requirements of OTP key management, and thus inherits the property of perfect secrecy from OTPs.

The EDE scheme combines two well-known techniques of *One-Time Pads* and *Error*

Correcting Codes to achieve a cryptographic primitive for IMDs. Emergency medical personnel can gain access to patients' IMDs by measuring the patients' real-time ECG signals while adversaries are blocked since they lack the capability to measure ECG data in real time. Thus the designed EDE scheme achieves the balance of high security and high accessibility (in an emergency situation).

Chapter 6

Multiple ECG Fiducial-points based Binary Sequence Generation

As discussed in Chapters 3 & 5, generating random binary sequences is a fundamental issue in the ECG-based cryptography. This ECG-based security approach benefits IMDs. For instance, the H2H algorithm [25] uses random BSes generated from real-time ECG signals to perform authentication between an IMD and a device programmer; therefore, patients bearing IMDs, e.g., pacemakers and implantable defibrillators, can be cared for by any qualified medical personnel in emergencies while preventing attackers from accessing IMDs. These ECG BSes can also be used for the fuzzy commitment based key distribution (in Chapter 3) or the EDE scheme which uses modified one-time pads (in Chapter 5).

Currently, random ECG BSes are typically generated by processing inter-pulse intervals of ECG signals [35, 91, 2, 25, 30, 48]. IPIs are defined as time intervals between two consecutive heartbeats and are proven to be as a random source [2]. As proposed in [25,2], only the last 4-bits of each binary IPI can be regarded as random bits and are extracted to form random BSes. Therefore generating a 128-bit BS requires at least 32 IPIs, which means the sensor node has to successfully detect at least 33 consecutive heartbeats. Considering that the heart rate of a normal sinus rhythm is 60-100 beats per minute for an

adult, generating a 128-bit BS would require 20-30 seconds which is considerably time consuming for real-time requirements of a WBAN system. This problem is also pointed out by the study in [87].

In order to reduce the latency of such methods, this chapter presents an ECG Multiple Fiducial-points based Binary Sequence Generation (MFBSG) algorithm. Using this algorithm, multiple ECG fiducial points are used to obtain feature values, including RR, RQ, RS, RP and RT intervals. Later these ECG feature values are utilized to generate random BSes with low latency. Discrete wavelet transforms are employed to decompose the ECG signal at multiple resolutions and detect these fiducial points on the signal. According to our analysis on real ECG data, these ECG feature values have characteristics of randomness, and thus can be utilized to generate random BSes. Compared with the schemes that solely rely on IPIs to generate BSes, this MFBSG algorithm uses five feature values from one heartbeat cycle. Therefore the time required to generate a BS is significantly reduced, achieving the design goal of low-latency for a WBAN system. According to our analysis, the complexity of the algorithm is comparable to that of fast Fourier transforms. These randomly generated ECG BSes can be used as security keys for encryption or authentication, or be used to facilitate symmetric key distribution within a WBAN system. The advantages of this MFBSG algorithm are summarized as follows:

- The proposed MFBSG algorithm uses multiple fiducial points to obtain five feature values from one heartbeat cycle. Compared to the BS generation algorithms relying solely on IPIs, it improves the time efficiency of BS generation, and thus achieves the design goal of low-latency.

- Since the MFBSG algorithm uses more fiducial points when compared to the algorithms exclusively based on IPIs, discrete wavelet transforms are proposed to precisely detect ECG fiducial points and obtain time intervals between them. We have analyzed the complexity of the algorithm, and found that it is comparable to

that of fast Fourier Transforms.

- Unlike schemes based on the pseudo-random number generator (PRNG), the MFBSG uses ECG signals as natural random sources. Therefore, it does not need random seeds and complex computations which are essential in the PRNG.

The organization of this chapter is as follows.

- We propose and describe the MFBSG algorithm in Section 6.2, and its wavelet process and BS gneration process in Sections 6.3 and 6.4, respectively. This algorithm uses wavelet transforms to extract ECG fiducial points which are later used to produce random BSes.

- Section 6.5 presents a comprehensive evaluation of the algorithm on aspects of ECG wavelet transforms, randomness of ECG features, BS randomness and distinctiveness, and complexity analysis.

- We discuss a potential application of the MFBSG algorithm in Section 6.6. The generated BSes can be used as secret keys for encryption and authentication purposes or be used to facilitate the key distribution.

6.1 ECG Modelling

The blood circulation of the human body can be regarded as an out-of-band channel to share security information for WBANs. ECG signals measured synchronously by two sensors within the same WBAN have a major part in common because the heartbeats are from the same source of the signal. Therefore, the ECG IPI values in the time domain have been investigated to secure WBANs [35, 91]. However, according to the analysis in [2,25], only the last 4-bits of each IPI value can be extracted to form random BSes. So generating a 128-bit BS in the time domain would require 20-30 seconds for the normal

sinus rhythm, which is inefficient for wireless communication. In order to improve the BS generation efficiency, we review the characteristics of ECG waveform first.

Normally one ECG trace includes three major waves: P wave, QRS complex and T wave. The P wave represents the depolarization impulse of the atria; QRS complex represents the ventricular depolarization while the T wave represents the ventricles repolarization [92, 93]. Fig. 6.1 depicts fiducial points of the ECG waveform. The fiducial points are landmarks that could locate the three major waves in an ECG trace [93], with each point explained in Table 6.1.

Table 6.1: Symbols of fiducial points

Fiducial point	Explanation
Ps	The starting point of P wave
P	The peak value of P wave
Pe	The end point of P wave
Q	The peak value of Q wave
R	The peak value of QRS complex
S	The peak value of S wave
Ts	The start point of T wave
T	The peak value of T wave
Te	The end point of T wave

The work done in the previous research [35,91,2,25] relies solely on the feature of time distance between two consecutive R peaks, namely the Inter-Pulse-Interval (IPI), is used to generate random BSes. In the scenario of normal sinus rhythms, the peak values of P, Q, S and T waves are also observable. Hence we can conclude from the observation that their peak values can be potentially used in WBAN security.

We use the fiducial points on the ECG trace to extract feature values for random BS generation. Unlike the ECG-based authentication where ECG amplitude values are used

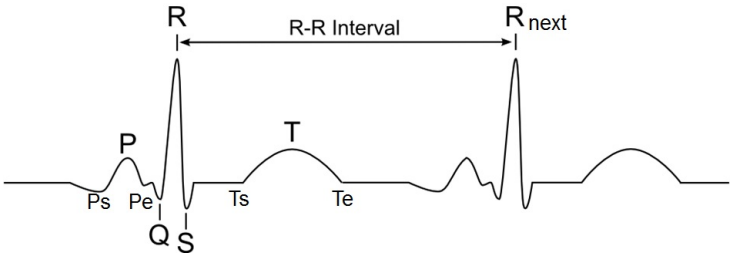


Figure 6.1: Fiducial points on ECG wave form.

as features for security purposes [94, 95], we choose not use this feature (P amplitude, QRS amplitude, T amplitude) due to the fact that amplitude values of two synchronously measured ECG traces from different parts of the same body vary significantly, normally depending on the distance from the heart. From the signal processing point of view, it is hard to detect the start and the end points of P and T waves accurately; thus we do not use points of Ps, Pe and Ts, Te in our scheme either. Therefore, we decide to use fiducial points of wave peaks, such as P, Q, R, S, T. Time intervals between them, such as RQ, RS, RP, RT as well as RR intervals, are used as feature values in the MFBSG algorithm in order to generate BSes. Each interval is the time distance between two wave peaks, as defined in Table 6.2.

Table 6.2: ECG feature values used in the MFBSG algorithm.

Feature values	Explanation
R-R	The time interval between R peak values of the i^{th} and the $(i-1)^{th}$ heartbeats
R-Q	The interval between R, Q wave peaks in the i^{th} heartbeat
R-S	The interval between R, S wave peaks in the i^{th} heartbeat
R-P	The interval between R, P wave peaks in the i^{th} heartbeat
R-T	The interval between R, T wave peaks in the i^{th} heartbeat

6.2 Overview of the MFBSG Algorithm

Several methods were proposed to generate BSes by solely using ECG IPIs [2,35,91,96, 97]. Based on the analysis of these methods, the proposed MFBSG algorithm is proposed, as shown in Fig.6.2. The algorithm samples ECG signals and removes sampling noise at the beginning. Then, it runs through two major process stages: the ECG wavelet process which uses wavelet transform to process raw ECG data and extracts timing informaiton of fiducial points (P, Q, R, S & T), and the BS generation process which generates random BSes by using the five types of ECG featurs (RR, RQ, RS, RP & RT). Binary Features(BFs) are binary digits extracted from each ECG feature.

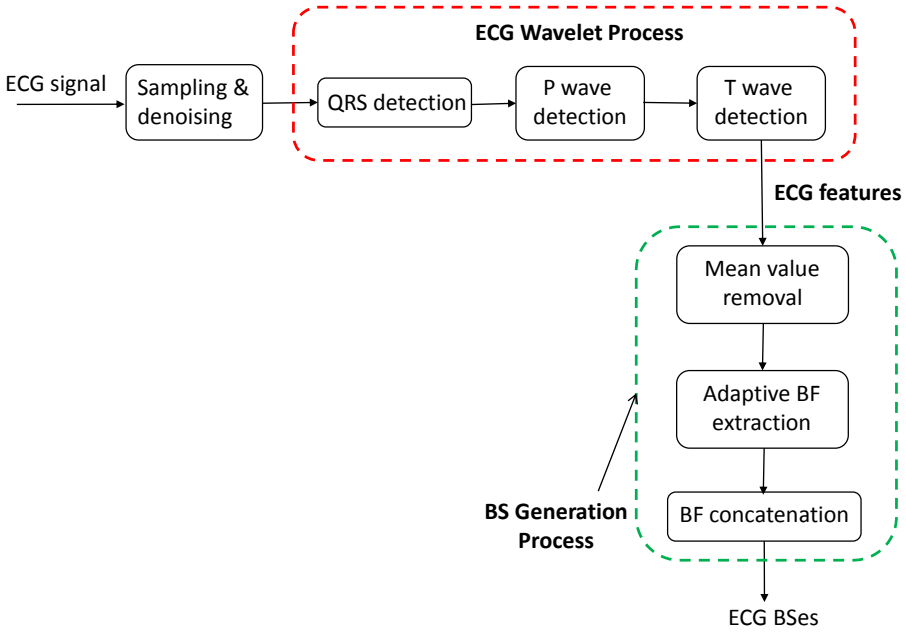


Figure 6.2: A block diagram of the wavelet-based ECG BS generation algorithm.

6.2.1 ECG Wavelet Process

The purpose of this process is to use the technique of wavelet transforms to process sampled ECG signals, involving the following steps: QRS detection, P wave detection and T wave detection. Wavelet transforms are based on a set of wavelets with limited duration, and allows the representation of temporal features of a signal at different resolutions. Since the ECG signal is characterized by a cyclic occurrence of patterns with different frequency content (P waves, QRS complexes and T waves), it is suitable to use the wavelet transform to analyze the ECG signal [98, 99]. After applying the wavelet transforms, ECG fiducial points (peak values of P, Q, R, S & T) can be extracted. Then five feature values from one heart beat cycle (RR, RQ, RS, RP & RT intervals) are calculated and are used as inputs to the next stage.

6.2.2 BS Generation Process

The purpose of this process is to generate random BSes by processing five feature values from one heartbeat cycle. After receiving ECG feature values, binary digits are extracted from each feature, named binary features (BFs). These BFs are then concatenated to obtain an x-bit long BS. Randomness is a vital requirement if BSes are used for WBAN security purposes. Meanwhile, generating BSes with high timing efficiency is another important requirement for a communication system. In order to balance the requirements of randomness and timing efficiency, this process is broken down into three discrete steps: mean value removal, adaptive BF extraction and BF concatenation.

6.3 Stage 1: ECG Wavelet Process

6.3.1 Wavelet Transforms

Wavelet transforms (WT) use a series of small wavelets with limited duration to decompose a signal. It can give good estimation of the signal in both the time and the frequency domains [100]. For a wavelet function $\psi(t)$, the wavelet transform of a signal $f(t) \in L^2(R)$ is given by

$$W_f(a,b) = \frac{1}{\sqrt{|a|}} \int_{-\infty}^{\infty} f(t)\psi^*(\frac{t-b}{a})dt \tag{6.1}$$

where a is the scale factor and b is the translation of $\psi(t)$. $\psi^*(t)$ denotes the complex conjugation of $\psi(t)$. A function $\psi(t)$ could be used as a wavelet only when its Fourier transform $\psi(\omega)$ satisfies:

$$\int_{-\infty}^{\infty} \frac{|\psi(\omega)|^2}{|\omega|} < \infty \tag{6.2}$$

The parameters, a and b, vary continuously over the real numbers. For smaller values of scale a, the temporal support for the wavelet decreases and the transform coefficients give more information about the higher frequency components of the analyzed signal, and vice versa. If $a = 2^j$ and $b = 2^j l (j, l \in Z)$ where Z is the set of integer, it becomes a dyadic WT and is denoted by,

$$W_f(2^j,b) = \frac{1}{\sqrt{2^j}} \int_{-\infty}^{\infty} f(t)\psi^*(\frac{t}{2^j} - l)dt \tag{6.3}$$

In our application, a quadratic spline wavelet is applied to detect ECG features. This wavelet was already used for ECG delineation analysis in [101, 98]. Its Fourier transform is

$$\psi(\omega) = j\omega(\frac{sin\frac{\omega}{4}}{\frac{\omega}{4}})^4 \tag{6.4}$$

For the dyadic WT, Mallat produced a classical fast wavelet decomposition and reconstruction algorithm, called Mallat algorithm [102]. Then the dyadic WT can be calculated by,

$$S_{2^j}f(n) = \sum_{k\in Z} h_k S_{2^{j-1}}f(n-2^{j-1}k) \tag{6.5}$$

$$W_{2^j}f(n) = \sum_{k\in Z} g_k S_{2^{j-1}}f(n-2^{j-1}k) \tag{6.6}$$

where $h_k, k \in Z$ and $g_k, k \in Z$ are coefficients of a low-pass filter and a high-pass filter. $S_{2^j}f(n)$ are approximation coefficients while $W_{2^j}f(n)$ are detail coefficients. $S_{2^0}f(n)$ is the analyzed signal $f(n)$, and is referred to the sampled ECG signal in our application.

Singularities normally carry critical information pertaining to the analyzed signals. In our applications, we are interested in singularities on ECG waves (peak values of P, Q, R, S and T waves). A quadratic spline wavelet with compact support, defined by Mallat et al. [103], is used in our algorithm. It is the derivative of a smoothing function. Fore every uni-phase wave, the negative minimum at each scale corresponds to the rising edge of the wave while the positive maxima corresponds to its falling edge. According to the analysis in [102], for the quadratic spline wavelet, the peaks of each wave correspond to zero-crossing points of a positive maximum-negative minimum pair of $W_{2^j}f(n)$ at different scales. As the selected $\psi(t)$ is a derivative of a smoothing function [101], we could obtain peak values of P, Q, R, S and T waves after the dyadic WT process on ECG signals. This wavelet has been well studied for ECG signal processing and singularity detection by Li et al. [101], Martínez et al. [98], Ghaffari et al. [104], etc.

6.3.2 ECG Fiducial Point Detection

Discrete Wavelet Transform (DWT) is processed on sampled ECG signals for feature extraction purpose. Within the DWT calculation, it implicitly performs filtering in the frequency domain. However, the ECG signal could still be filtered as usual in order to

reduce noise before the DWT process commences. According to the analysis of equivalent responses of $W_{2^j} f(n)$ at different scales [98] and the spectrum analysis of ECG, noise and artifact [105], it is obvious that the decomposed ECG signals at scales 2^1 and 2^4 contain most of the energy of the QRS complex. Meanwhile, the main energy of P and T waves lies at the scales 2^4 and 2^5. The ECG signal at scale larger than 2^5 is influenced by the baseline drifting, because the larger scale reflects the low-frequency part of the signal. So, the energy of QRS complex, P wave and T wave at larger scales decreases further while the noise and artifact part increases [98, 100, 106]. Therefore, we use scales from 2^1 to 2^4 for QRS complex detection and the scales 2^4 and 2^5 for P and T wave analysis.

After the DWT process, the local positive maximum-negative minimum pairs and its zero-crossing points at each scale correspond to singularities on the ECG signal. Within each heartbeat cycle, the QRS complex is the predominant part; so we detect the QRS complex and locate the R peak first, and then detect other significant points. The algorithm follows these steps: a) QRS complex delineation and R peak value detection; b) Q and S waves and their peak values detection; c) P wave delineation and its peak value detection; d) T wave delineation and its peak value detection.

(a) Detection of QRS complex and R peaks

QRS complexes in each cycle are detected by analyzing the local positive maximum-negative minimum pairs from large to small scale [105, 101, 100, 98], described as below. Firstly, modulus maxima pairs with opposite signs of $W_{2^4} f(n)$ at the scale 2^4, which is larger than a threshold ϵ^4_{qrs}, are determined and their positions $\{n^4_k, k = 1, \cdots, N\}$ are recorded. Secondly, at the scale 2^3, find modulus maxima pairs larger than a threshold ϵ^3_{qrs} at the neighbourhood $\{n^4_k\}$ and record their positions as $\{n^3_k\}$. The largest modulus maxima pair is selected if there are more than one modulus maxima pair. The record $\{n^3_k\}$, $\{n^2_k\}$ and $\{n^1_k\}$ are set to zero if no maxima pair exists. Following this process, the

location sets $\{n_k^2\}$ and $\{n_k^1\}$ of modulus maxima pair at scale 2^2 and 2^1 are determined. The algorithm here searches maxima pairs from the large to the small scale in order to reduce the affect of high frequency noise, since the DWT at a large scale gives more information about the low-frequency of the analyzed signal. By using an appropriate modulus threshold, the faulty detected maxima pairs at a large scale can be reduced greatly. According to the relationship between signal singularities and its DWT, the zero-crossing points at scale 2^1 are the R peaks on ECG signals. The distance between the positive-maxima and the negative-minima pair is normally slightly smaller than the QRS width. If this distance is larger than a certain time period, then this detected modulus maxima pair is isolated and has to be removed. The thresholds here, $\epsilon_{qrs}^4, \epsilon_{qrs}^3, \epsilon_{qrs}^2, \epsilon_{qrs}^1$, for each scale have to adapt to the ECG signal variations.

(b) Detection of Q, S waves and peaks

Q and S waves in QRS complexes contain characteristic information and their peak values are used in our BS generation algorithm. When detecting Q and S peaks, the algorithm starts from the R peak location in that QRS complex. For the decomposed signal at scale 2^2, there is a modulus maxima pair with opposite signs of $W_{2^2}f(n)$ before and after the R peak location, corresponding to the Q peak and the S peak, respectively. The local modulus maxima must be larger than a threshold ϵ_Q^2 or ϵ_S^2 respectively for the Q wave or the S wave. After locating zero-crossing points of the modulus maxima pairs at scale 2^2, a process which is similar to the R peak detection is conducted at scale 2^1. The zero-crossing points before and after R peaks at the scale 2^1 are assigned to Q and S peaks, respectively.

(c) Detection of P waves and peaks

The P wave and its peak value detection algorithm is as follows. Firstly, a search window relative to the QRS complex location is defined, with its width dependent on the calculated RR interval. On the decomposed signal at scale 2^4, we search for the local modulus maxima pair of $W_{2^4}f(n)$ with opposite signs. A threshold, ϵ_P^2, is set for the maxima searching. If the local modulus maxima pair exists, the zero-crossings between them are regarded as P wave peaks; otherwise, a similar process is to be done at scale 2^5.

(d) Detection of T waves and peaks

The T wave and its peak value detection follows a similar process as the P wave algorithm. The detection is done at scale 2^4 first. If the local modulus maxima pair does not exist, the algorithm then search the peak value at scale 2^5.

6.4 Stage 2: BS Generation Process

The BS generation process uses five types of features from each ECG heartbeat cycle. In the i_{th} heartbeat, measured ECG feature values are denoted by RR_i, RQ_i, RS_i, RP_i&RT_i. In order to simplify the algorithm for a resource-restricted wireless sensor node, each feature follows the same process to generate random BSes. For a series of ECG features, the BS generation process is described as below:

6.4.1 Mean Value Removal

As analyzed in Section 6.5, the distribution of these ECG features, although virtually following a normal distribution, has an offset from the y-axis. Therefore, this offset has to be removed from ECG features. According to the properties of a normal distribution,

the mean value of features can be used to remove this offset as,

$$\begin{cases} RR'_i = RR_i - mean(RR) \\ RQ'_i = RQ_i - mean(RQ) \\ RS'_i = RS_i - mean(RS) \\ RP'_i = RP_i - mean(RP) \\ RT'_i = RT_i - mean(RT) \end{cases} \tag{6.7}$$

The resulting set, $(RR'_i, RQ'_i, RS'_i, RP'_i \& RT'_i)$, are used in the next step to generate BSes.

6.4.2 Adaptive BF Extraction

Methods in [2,35,91] propose to extract a fixed number of bits from each IPI. However, the variation range of ECG features changes in each dataset. In order to reflect this variation, we propose an adaptive BF extraction method which determines the number of bits extracted from each feature according to the Standard Deviation (SD) of measured features, denoted by,

$$m = \lceil log2(\sigma(f_s)) \rceil \tag{6.8}$$

where $\sigma(.)$ represents the SD of a dataset and f_s represents dataset of any type of ECG features. $\lceil . \rceil$ rounds the decimal value of $log2(\sigma(f_s))$ to its nearest integer towards infinity. SDs of each type of ECG features are different, and therefore the number of extracted bits, m, change accordingly. For the i_{th} heartbeat, BFs extracted from ECG features are

denoted by,

$$\begin{cases} BF_i^1 = BFextract(RR'_i, m_1) \\ BF_i^2 = BFextract(RQ'_i, m_2) \\ BF_i^3 = BFextract(RS'_i, m_3) \\ BF_i^4 = BFextract(RP'_i, m_4) \\ BF_i^5 = BFextract(RT'_i, m_5) \end{cases} \tag{6.9}$$

where $BFextract(.)$ represents a function to extract $m_j(j = 1, 2, \cdots, 5)$ bits from its input $(RR'_i, RQ'_i, RS'_i, RP'_i \& RT'_i)$, respectively.

There are many ways to extract bits from each feature [35, 91]. In this algorithm, we use the method proposed in our work in [30]. It has four steps: Simple Moving Average (SMA) process, Gray coding, removal of the Least Significant Bit(LSB) and parity check. This method is to generate BSes for purposes of authentication and key distribution (as described in Section 6.6). If the BS is to be used as a secret key and distributed to other sensors by a secure channel, then we can just convert the feature value to binary and extract its last m bits directly. So, five BFs can be extracted from five features in the i_{th} heartbeat, denoted by $BF_i^1, BF_i^2, BF_i^3, BF_i^4$ & BF_i^5. This adaptive BF extraction method ensures the randomness of BSes based on the variations in ECG data measured real-time.

6.4.3 BF Concatenation

The extracted BFs from the i_{th} heartbeat, $BF_i^j(j = 1, 2, \cdots, 5)$, are then concatenated by,

$$BS_i = BF_i^1 \parallel BF_i^2 \parallel BF_i^3 \parallel BF_i^4 \parallel BF_i^5 \tag{6.10}$$

where $\parallel$ is a concatenation operation. BS_i represents a BS generated from the i_{th} heartbeat. In order to form an x-bit BS, BS_i generated from l consecutive heartbeats are then

concatenated by,

$$BS_x = BS_1 \parallel BS_2 \cdots \parallel BS_l \tag{6.11}$$

The number of ECG heartbeats, l, varies according to the subject's ECG signals and ECG measuring equipment. Besides, we need to have redundant ECG heartbeats, so that the algorithm can generate intended length of a BS from one sampled ECG trace in most cases. In order to ensure that generated BSes satisfy the requirement of randomness, we design a series of randomness tests, including calculating entropy, using NIST randomness test suite to test BSes, and distinctiveness analysis. The evaluation of randomness is presented in Section 6.5.

6.5 Experiments and Results

In this section, we evaluate the MFBSG algorithm by conducting a series of experiments. Lacking the ability to obtain ECG measurements in a laboratory setting, our analysis approach is similar to those in [81, 86, 47, 87] which uses the MIT PhysioBank database (http://www.physio net.org/physiobank). Experiments were carried out on the ECG data from 97 subjects: 79 subjects from the European ST-T database [78] and 18 subjects from the MIT-BIH Normal Sinus Rhythm database [70].

6.5.1 Wavelet-based ECG Signal Process

The MFBSG algorithm uses discrete wavelet transforms to process ECG signals and detect fiducial points. In order to detect fiducial points, ECG signals are decomposed by the DWT at four levels. An example of an ECG signal and corresponding wavelet transforms is shown in Fig. 6.3. The first sub-figure is the original ECG signal while the following four sub-figures (d1, d2, d3, d4) show detail coefficients of the ECG signal at scale $2^1, 2^2, 2^3, 2^4$. In order to explain wave peak detection process by using different level

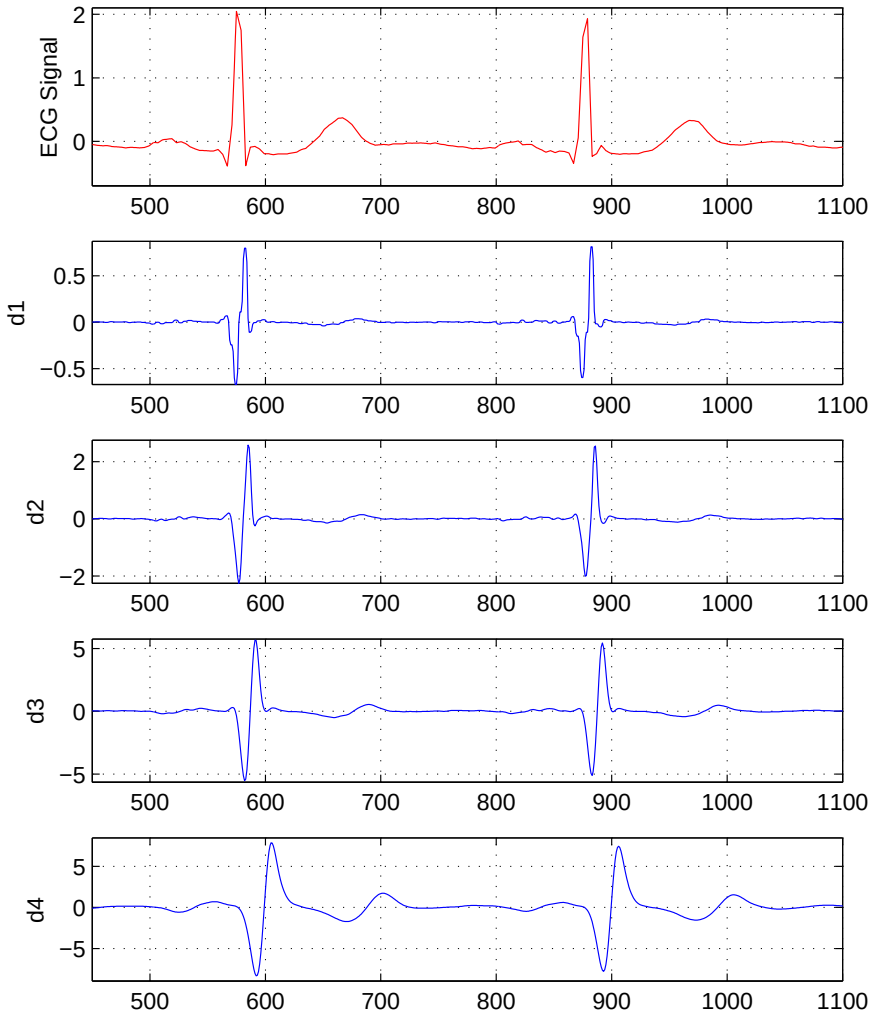


Figure 6.3: An ECG signal and its detail coefficients of wavelet transforms.

information, we enlarge related portion of the figure, as shown in Fig. 6.4, Fig. 6.5 and Fig. 6.6, where $f(n)$ is the sampled ECG signal.

Fig. 6.4 shows examples of QRS complex and individual wave peak detection. Q, R, and S peaks are marked out with lines on $f(n)$. In principle, the QRS complex is used as references for the detection of other waves and complexes. The detection algorithm follows steps as below: (i) it determines the QRS complex; (ii) it finds significant modulus

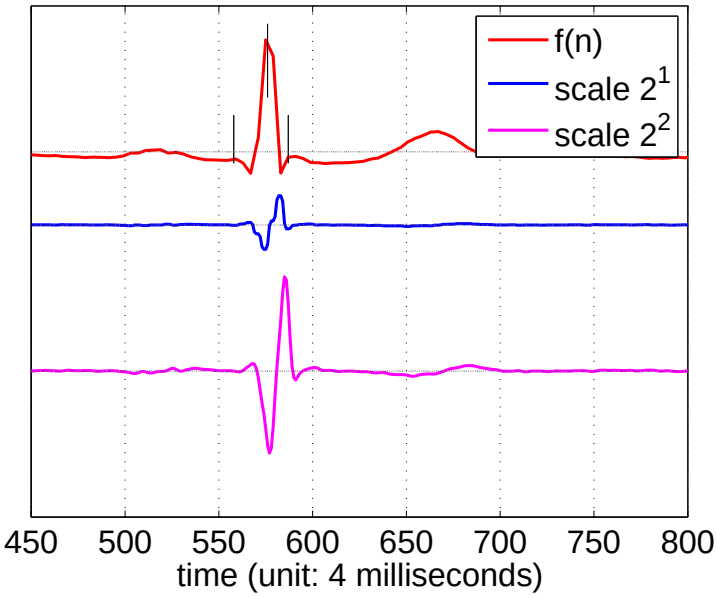


Figure 6.4: An example of QRS complex and its individual wave peak detection.

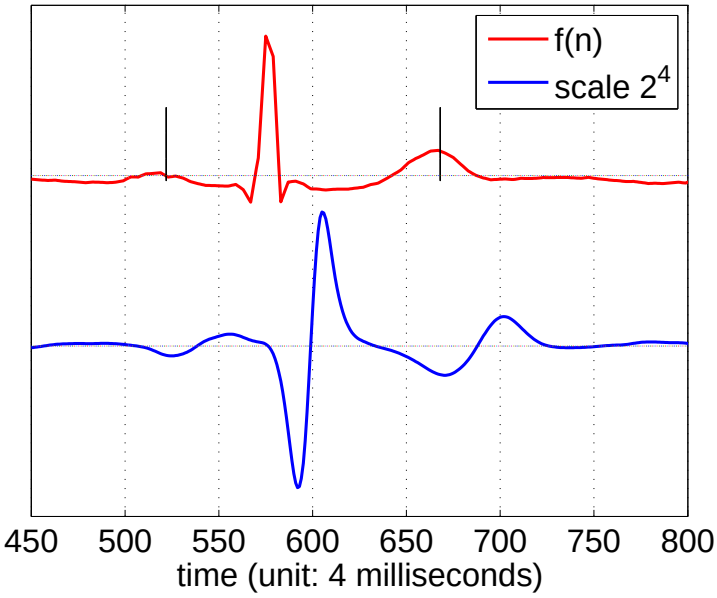


Figure 6.5: An example of P and T wave peak detection at scale 2^4.

maxima at scale 2^2, and locates the R peak at scale 2^2 and 2^1; (iii) the modulus maxima pairs before and after the R peak are located and their zero-crossings at scale 2^1 are

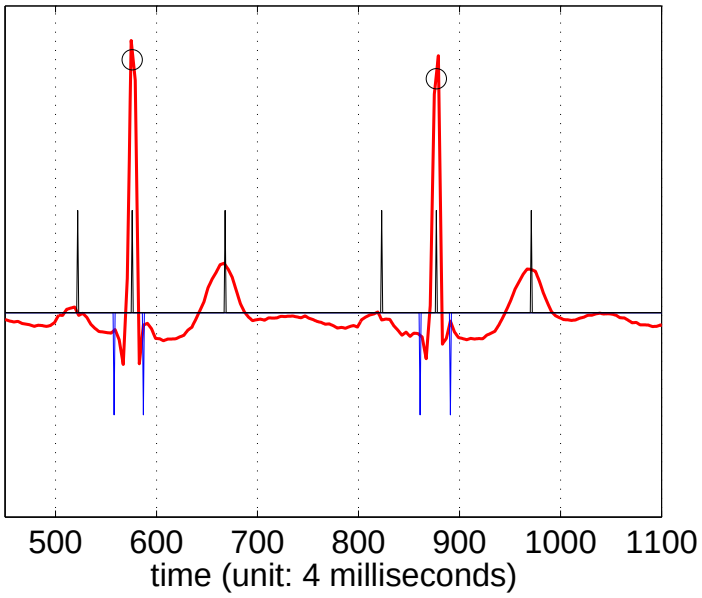
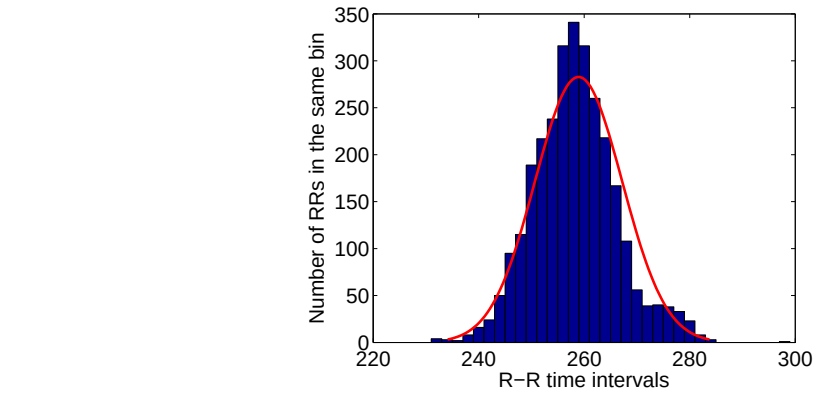


Figure 6.6: An example of ECG fiducial point detection results.

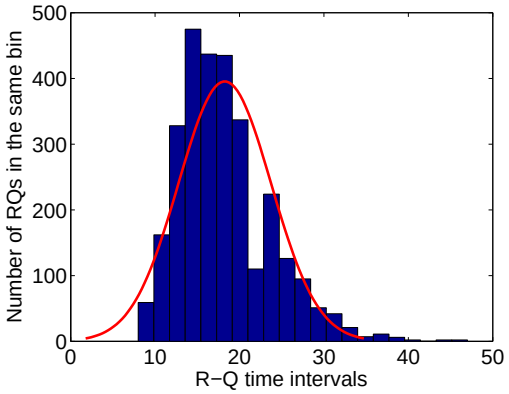
assigned to Q and S peak values.

Fig. 6.5 shows examples of detection of P and T wave peaks. Detection processes of P peaks and T peaks are carried out at scale 2^4 and use a similar algorithm. P and T peaks are marked out with lines on $f(n)$. The P peak detection follows steps as below: (i) it defines a P wave search window which is dependent on the calculated RR interval; (ii) within the search window, it looks for the modulus maxima pair at scale 2^4, and its zero-crossing point is considered as the P peak value. The T peak detection is similar to this process.

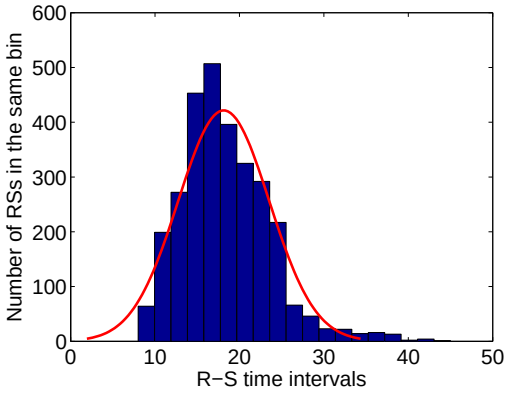
Fig. 6.6 shows examples of ECG fiducial point detection results. P, Q, R, S and T peaks are marked out with lines on the sampled ECG signal. It clearly shows that P, Q, R, S and T peaks are detected correctly marked out on the sampled ECG signal. After locating these peaks, their intervals, including RR, RQ, RS, RP and RT, are calculated and then used in the BS generation process.



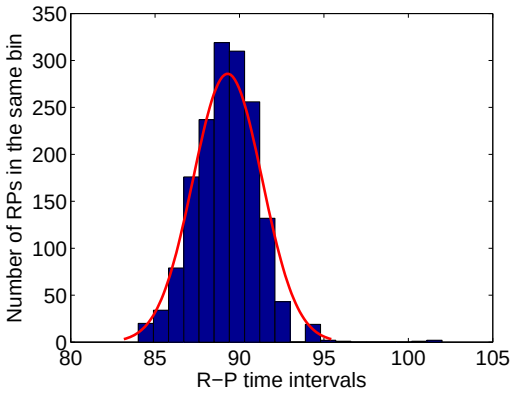
(a) RR intervals, $\mu = 259$, and $\sigma = 9$.



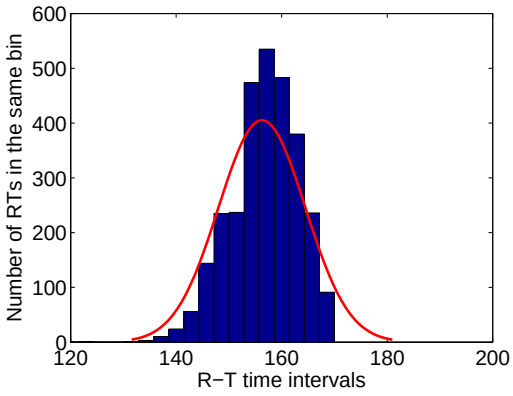
(b) RQ intervals, $\mu = 18, \sigma = 6$.



(c) RS intervals, $\mu = 18, \sigma = 5$.



(d) RP intervals, $\mu = 89, \sigma = 3$.



(e) RT intervals, $\mu = 156, \sigma = 9$.

Figure 6.7: The normal distribution fitting to the fluctuation of ECG feature values, with unit of $4ms$ (sample rate $250Hz$).

6.5.2 Randomness of ECG Features

The feasibility of using selected ECG features (RR, RQ, RS, RP & RT) to generate random BSes is based on the fact that all these ECG features possess the nature of randomness. In this experiment, we evaluated this randomness by collecting consecutive ECG feature values and plotting their histograms, with a result shown in Fig.6.7. By examining these figures, we can conclude the following:

1. The fluctuation of consecutive RR intervals, as shown in Fig.6.7 (a), fits into a normal distribution. Thus its distribution is almost normal, which indicates the randomness of RR intervals.

2. Likewise, the distributions of RQ, RS, RP & RT intervals, as shown in Fig.6.7 (b),(c),(d) & (e), are close to normal. Thus these features also possess the nature of randomness.

3. We can see from these figures that their standard deviations, σ, vary for each feature type. Therefore, we need to use an adaptive method which can determine the number of bits extracted from each feature according to their standard deviations, as described in Eq.6.8.

Therefore, these selected ECG features possess the nature of randomness. This is a fundamental property to ensure that BSes generated from these features are random.

6.5.3 Randomness of Generated BSes

Randomness of generated BSes is a vital requirement when applying BS generation techniques for WBAN security purpose. Therefore, in this experiment, we generated x-bit BSes from captured ECG features (RR, RQ, RS, RP & RT), and then analyze the randomness by calculating their entropy and running an NIST randomness test suite [77]. In our experiments, we generated 128-bit BSes for evaluation.

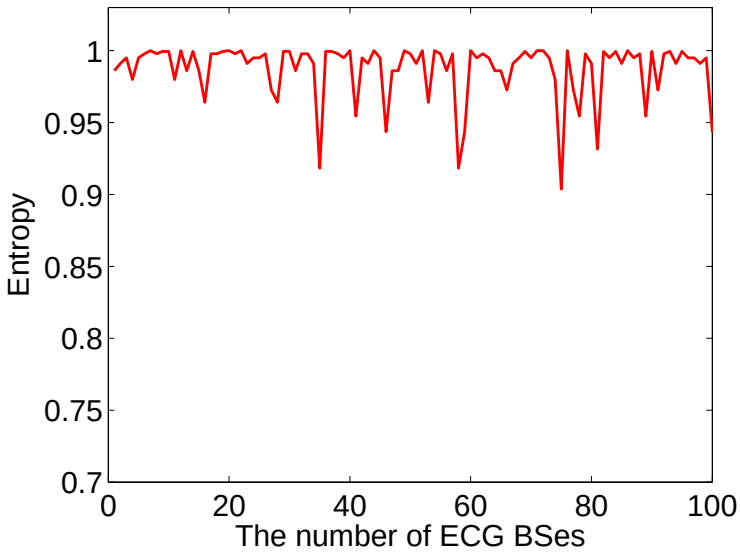


Figure 6.8: The entropy of generated ECG BSes, with the mean entropy of 0.9874.

(a) Entropy Analysis

The entropy is calculated to measure the uncertainty of generated ECG BSes. For a random variable $\chi = 0, 1$, we can calculate the entropy of each binary sequence using Eq. 3.1. The entropy results of BSes generated from about 100 ECG samples are shown in Fig.6.8. It can be seen that the entropy values of most ECG BSes were close to 1, with the mean value of 0.9874. So, the distribution of 0s and 1s in BSes is close to a uniform distribution. Furthermore, we conducted a two-tailed runs test in the experiment, which showed that more than 95% of ECG BSes passed the test with a significance level of 5%. Thus, the generated ECG BSes truly exhibit the property of randomness.

(b) NIST Randomness Test

In order to comprehensively analyze the randomness of generated ECG BSes, we performed a statistical experiment with the National Institute of Standards and Technology (NIST) randomness test suite [77]. The state-of-the-art NIST test suite is used for test-

ing random and pseudo-random number generators for cryptography. The outputs are P-values which indicate the probability that the generated BSes are random or not. If the P-value is less than a threshold (normally 0.01), the hypothesis that a BS is random is then rejected. The aggregate of BSes generated from ECG recordings from the European ST-T Database is used for the test, with results shown in TABLE 6.3. Tests that produce multiple P-values are represented by a (+) and followed by the number of different generated values in parenthesis. The table displays their mean values. The table shows that all P-values are greater than 0.01. So, it passes the NIST randomness test.

We compare the MFBSG algorithm with the ECG IPI-based BS generation method proposed by Zhang et al. [91] and the analog-to-digital (ADC) based method proposed by Callegari et al. [107] in terms of the NIST test performance. There are common executed NIST tests among them, including Frequency test, Block Frequence test, Runs test and Longest Run test. According to TABLE IV in [91] and TABLE III in [107], P-values of tests of BSes generated from these two methods are around 0.99. As explained in [77], a P-value ≥ 0.01 would mean that the sequence would be considered random with a confidence level of 99%. Therefore, BSes generated from this algorithm and the previous two methods can be considered as random with a confidence level of 99%. Their performance of randomness is comparable.

6.5.4 Distinctiveness of Generated BSes

The purpose of distinctiveness analysis is to ensure that ECG BSes generated from different subjects are significantly different from each other. Thereby, adversaries cannot obtain any information of a BS by measuring another subject's ECG signal. The Hamming distance ($D_{hamming}$) is used as a metric to assess the difference between any two BSes of equal length. $D_{hamming}$ is measured as the number of positions at which the corresponding bit values are different. The larger the $D_{hamming}$, the better the performance of the

Table 6.3: NIST statistical test results for ECG BSes generated by the MFBSG algorithm.

Statistical test	p-value	Proportion	Pass/Fail
Frequency	0.534146	0.9000	Pass
Block Frequency	0.122325	0.9000	Pass
Cumulative Sums$^+$(2)	0.167800	0.9000	Pass
Runs	0.911413	1.0000	Pass
Longest Run	0.534146	0.9000	Pass
Rank	0.066882	1.0000	Pass
FFT	0.739918	0.8000	Pass
Non-overlapping Template$^+$(148)	0.1548	0.9500	Pass
Serial$^+$(2)	0.281897	1.0000	Pass
Linear Complexity	0.350485	0.9000	Pass

generated BSes. For two bits at the same position of two BSes, u_a and u_b, the probability, $P(u_a, u_b)$, can be denoted by,

$$P(u_a, u_b) = \begin{cases} 0.25, & u_a = 0, u_b = 0 \\ 0.25, & u_a = 1, u_b = 0 \\ 0.25, & u_a = 0, u_b = 1 \\ 0.25, & u_a = 1, u_b = 1 \end{cases} \tag{6.12}$$

The reason is that it has the same probability of being 0 or 1 for any bit within a random or pseudo-random BS. Thus, the average of $D_{hamming}$ of a sufficiently large set of x-bit BSes is expected to be around $x/2$, provided that BSes are random or pseudo-random. In the experiment, we sampled ECG signals on each subject from the European ST-T Database with over 300 random start-times and computed the average Hamming distance between any two BSes, as shown in Fig.6.9. We can see that the Hamming distance values between any two BSes fit into a normal distribution, with the average distance of about

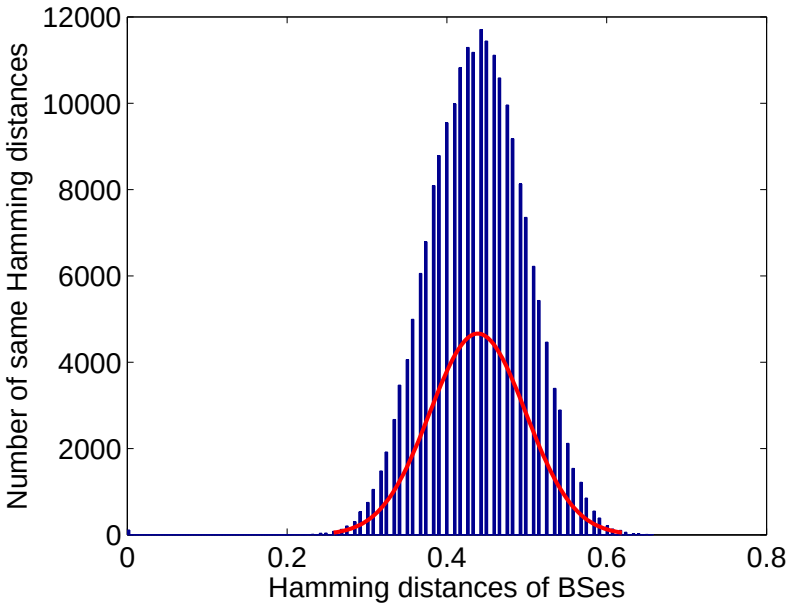


Figure 6.9: The distribution of Hamming distances between any two 128-bit BSes, with the mean distance of 45.3%.

45.3%, close to 50%. Therefore, BSes generated from a different subject's ECG signal are distinctive. This can prevent adversaries from attacking the secured WBAN by using a different subject's ECG data.

6.5.5 Algorithm Analysis

(a) Low Latency

The time required to generate a 128-bit BS in our algorithm is significantly reduced. We take experiments on ECG recordings (sample rate 250Hz) shown in Figure 6.7 as an example. According to rules described in Equation (6.8), we can extract 4 bits from each RR interval, 3 bits from each RQ and RS interval, 2 bits from each RP interval and 4 bits from each RT interval. So we can extract 16 bits in total from one heartbeat cycle. Then generating a 128-bit BS requires at least 8 heartbeat cycles. In order to have redundant information, we sampled 20% more than the basic sampling requirement, that is, totally

10 heartbeat cycles in one ECG trace. For a normal sinus rhythm with a heart rate of 60-100bpm, it takes around 6-10 seconds, much less than the time (20-30s) that is required for the solely IPI based methods. Hence this MFBSG algorithm improves the latency of the ECG-base BS generation for sensors within a WBAN system.

(b) Complexity Analysis

Our complexity analysis is performed on two stages of the MFBSG algorithm, respectively: the ECG wavelet process and the BS generation process. As analyzed in [108], the complexity of fast wavelet transforms is $O(n)$ while that of Fast Fourier Transforms (FFT) is $O(n \log n)$, where n is the sampled data size. Therefore, the computational complexity of DWT is comparable to that of FFT. Because of its low-complexity energy-efficient performance, DWT has been widely used to process ECG signals in wireless body sensor nodes [109]. The operation count of the BS generation process depends on the number of heartbeats, l, so, its complexity is $O(l)$. Obviously $O(l)$ is smaller than $O(n)$. Therefore, the complexity of the whole MFBSG algorithm is $O(n)$. This complexity is comparable to that of FFT, and thus can be implemented on wireless sensor nodes.

(c) Analysis of ECG Abnormalities

The normal sinus rhythm usually has P waves, QRS complex and T waves in one heart beat cycle. However, sometimes abnormal ECG signals may present. This may happen on patients with cardiac arrhythmia. Even ECG signals from healthy people may have normal variation in heart rate sometime. In this scenario, some kind of waves may be missing within one heartbeat cycle or its peak value may be not detected by the DWT. Then we cannot obtain all intended intervals for the BS generation algorithm. In this case, the MFBSG algorithm will use as many intervals as it can obtain from the sampled ECG signal. For instance, it may use four intervals (RQ, RS, RT, and RR) from the

current heart beat, and use all five intervals from the next heart beat if the ECG signal returns to a normal sinus rhythm. The efficiency of the algorithm will be affected, but will still be better than the BS generation methods that are soly IPI-based. If all minor peaks (P, Q, S, & T) cannot be detected, then the MFBSG algorithm will only use the RR intervals to generate random BSes, which means that the MFBSG algorithm is degraded to be the same as the solely IPI-based methods. In an extreme case, the ECG waveform may become flat when an acute heart attack occurs, such as a myocardial infarction or a a dangerous arrhythmia. As discussed by Rostami et al. [25] and Zheng et al. [30], the security mechanism of the IMD is designed to be open to any access from an external programmer when this kind of ECG waveform is detected. This is because the patient's safety is much more important than the device security.

6.6 Discussion

Random BSes generated from ECG signals have several modes of applications. For instance, BSes can be used as secret keys for encryption or authentication, or be used to facilitate symmetric key distribution. Obviously the ECG BS generation is a fundamental technique that underpins all these applications which are summarized as below.

6.6.1 Secret key

A generated random BS can be used as a secret key for encryption purposes. It is distributed to other sensors within a WBAN and even other electronic devices that lie outside of a WBAN but require secure communications with it. The ECG is regarded as a natural random signal to generate random BSes. Normally in the computing domain, a PRNG is used to generate pseudo-random BSes for cryptographic applications. The PRNG-generated sequence is completely determined by a PRNG seed which has to be

carefully chosen and protected. However, using our proposed ECG BS generation algorithm, no PRNG algorithm is embedded in a wireless sensor and therefore no seeds are required and need protection.

6.6.2 Authentication

These generated random BSes can be used for authentication within a WBAN, which can determine whether a target sensor belongs to the same or a different WBAN [25]. Since the blood circulation system provides an inborn secure communication channel, ECG signals measured simultaneously by sensors within the same WBAN have a major part in common. Thus the BSes generated synchronously from these sensors are close to each other, and can be used to identify whether they represent the same human subject or not, that is, whether sensors are within the same WBAN or not. The study in [25] proposed to set up a secure channel using lightweight public-key cryptography in which the generated BS is transmitted from one sensor to another for authentication purpose.

6.6.3 Key distribution

The ECG BS-based symmetric key distribution is supported by a theory of the fuzzy commitment [51]. Two bio-sensors, a sender and a receiver, are deployed on the same human body, and generate random BSes, bs_1 and bs'_1, by measuring ECG signals synchronously. Since these two BSes, although generated synchronously from the same body, have slight variations due to the uncertainty of physiological signals and errors from the measuring equipment, a technique of Error Correcting Codes (ECC) is employed within the fuzzy commitment to tolerate the bit errors between bs_1 and bs'_1. So, the fuzzy commitment-based key distribution process is as below.

Firstly, a symmetric key, k, in the sender is mapped to an ECC codeword as $\hat{k} = f_{ecc}(k)$, where f_{ecc} is an ECC mapping function with its inverse function as f_{ecc}^{-1}. The

mapped key, $\hat{k}$, has redundant information to correct error bits. The commitment is defined as:

$$F(k, bs_1) = (hash(k), \hat{k} \oplus bs_1) \tag{6.13}$$

where $hash(.)$ is a one-way hash function and $\oplus$ is the bitwise XOR operation. This commitment F is sent to the receiver for the decommitment process as:

$$k' = f_{ecc}^{-1}(bs_1' \oplus (\hat{k} \oplus bs_1)) \tag{6.14}$$

where k' is the decoded key, and the inverse function f_{ecc}^{-1} is used to correct error bits. If $hash(k') = hash(\hat{k})$, then $k' = k$. The decommitment process is successful. A detailed analysis of the fuzzy commitment is described by Juels et al. [51].

6.7 Summary

In this chapter, we have presented an ECG Multiple Fiducial-points based Binary Sequence Generation (MFBSG) algorithm which exploits multiple ECG feature values, including RR, RQ, RS, RP and RT intervals, to generate random BSes with low latency. In previous papers, BSes generation methods use solely IPI features from each heartbeat cycle. Since normally the last 4-bits of each binary IPI can be regarded as random bits and are extracted to form BSes, generating a 128-bit BS normally takes around half a minute, which is considerably time consuming for a real-time wireless communication system. In order to improve the timing efficiency of BS generation, this chapter investigates multiple fiducial points and obtains five feature values from one heartbeat cycle, finding that these feature values have the characteristics of randomness. Hence the MFBSG algorithm utilizes these feature values to generate BSes. It can be up to five times faster than the solely IPI-based BS generation methods.

Since this MFBSG algorithm is efficient, it achieves the design goal of low-latency for a communication system. Compared with the PRNG, it uses ECG signals as a natural

source of random bits, and hence does not require random seeds and complex computations. Meanwhile, it avoids the protection of random seeds, a requirement for most security systems. As discussed in this chapter, generated BSes can be used as security keys for encryption or authentication, or can be used to facilitate key distribution by combining the primitive of the fuzzy commitment.

Chapter 7

Conclusions and Future Work

Implantable medical devices improve the quality of patients' life, and some of them play a critical role in patients health. The new generation IMDs incorporate more computations and communications capabilities, which raise security concerns in these IMD devices. There are unique challenges in the IMD security design because of its strictly limited resources and due to its critical medical application purposes. In this thesis, we have studied the use of ECG signals for the IMD security design. The blood circulation system in the patient's body is regarded as an inborn secure communication channel to transmit the ECG signal to the IMD and its external programmer. So, the IMD and the programmer can extract information from this real-time ECG signals for security purposes. The adversaries cannot detect this real-time ECG signal to obtain the information as long as they do not have physical contact with the patient.

We conclude our research on the use of ECG signals for the IMD security in Section 7.1. In Section 7.2, we discuss general considerations in the IMD security design, including proper assumptions, decoupled design, and the issue of safety of the IMD. Section 7.3 analyzes potential future work on this research topic.

7.1 Conclusions of the Thesis

In this thesis, we have studied the use of ECG signals for the IMD security. In order to design schemes properly, we have analyzed the issues of the IMD security in Chapter 2, including threat modeling, trade-offs in the design and solutions proposed in the existing literature. Based on this analysis, we have found that security solutions for the IMDs have to meet medical requirements, especially the guarantee of access to the IMDs in an emergency situation. Emergency doctors are normally not pre-authorized to have access to the IMDs. The access to security keys by Internet is not reliable in some areas, such as rural areas and developing countries. So, the designed security solutions should address this challenge.

We have designed the ECG-signal based key distribution schemes in Chapters 3 & 4. With these two schemes, the security key of the IMD can be distributed to the programmer for supporting the emergency treatment, so that doctors can gain access to the IMD by using this key. Both schemes require the IMD and the programmer to obtain real-time ECG signals from the same patient synchronously. The key in the IMD is hidden by using its measured ECG signal, and sent to the wireless channel. The programmer then reveals the key by using its own measured ECG signal.

In Chapter 3, a fuzzy commitment primitive is employed to perform this hiding/revealing process. Two random BSes are generated by the IMD and the programmer from their measured ECG signals, respectively. Since the measured ECG signals in the IMD and the programmer are from the same source (the heartbeat) and are synchronous, these two generated BSes match each other. So, these two BSes are used for hiding the key at one node and revealing the key at another node, respectively.

Chapter 4 uses a fuzzy vault primitive to hide/reveal the key for the IMD security. In this scheme, a polynomial is constructed in the IMD. The key is embedded in its coefficients. ECG IPI values are used to calculate points on the polynomial. These points,

mixed with chaff points, are sent to the programmer. The chaff points here are used to hide the polynomial points from adversaries. The programmer then extracts points on the polynomial and re-constructs the polynomial. So, the key is revealed by concatenating its coefficients.

We have designed an ECG based Data Encryption (EDE) scheme in Chapter 5 to provide information-theoretically unbreakable encryption for the IMDs. This EDE scheme combines two well-known techniques of classic one-time pads and error correcting codes. OTP keys are generated by each device from synchronously measured ECG signals, respectively. So, this scheme does not require a cryptographic infrastructure to support key distribution, storage, revocation and refreshment. This scheme can be used to transmit critical data from the IMD to the programmer, such as the IMD device number, the patient's name and date of birth, the doctor's information and even the symmetric key.

We need to generate random binary sequences from ECG signals for the schemes designed in Chapters 3 & 5. As found from our study, most schemes in the literature use IPIs exclusively to generate random binary sequences. Since normally the last 4-bits of each binary IPI can be regarded as random bits and are extracted to form BSes, generating a 128-bit BS will take around half a minute. This is considerably time consuming for IMDs in a real-time communications system. In order to improve its time efficiency, Chapter 6 investigates multiple ECG fiducial points and obtains five feature values from one heartbeat cycle, including RR, RQ, RS, RP and RT intervals, which are later utilized to generate random BSes. Compared with schemes that solely rely on IPIs, the time required to generate a BS is reduced significantly, thus achieving the design goal of low-latency for a communications system. According to our analysis, the complexity of the algorithm is comparable to that of fast Fourier transforms. Therefore, it can be used in a wireless IMD node.

In conclusion, this thesis has studied the possible use of ECG signals for the IMD se-

curity, including key distribution, encryption and binary sequence generation. With these schemes, unauthorized doctors can gain access to the IMD by measuring the patient's ECG signal in an emergency situation. Therefore, the ECG-based security solutions can address the unique challenge in the IMD security.

7.2 Discussions of IMD Security Design

IMDs have specific characteristics, e.g., implanted in the body, powered by non-chargeable batteries, required to have a long lifetime, playing life-saving functions for the patient. All of these create unique challenges in its security scheme design. Although a few solutions have been proposed for addressing the trade-offs among different IMD security goals, it is not clear which one is optimal in terms of implementation and commercialization. Some proposed solutions can provide reasonably high security, but consume a considerable amount of resources of the IMD. Thus, the IMD security design is still an open question.

In order to design a suitable security scheme for the IMD, we believe the following suggestions deserve some consideration.

7.2.1 Proper Assumptions

The IMD operation involves multiple parties, including patients, doctors and hospitals, emergency medical personnel, and IMD manufacturers. If we assume these parties are trustworthy, the IMD security design can be simplified. The IMD may record all accesses and active commands in the past few months into its log for the purpose of analysis and detection. For instance, if a depressed patient commits suicide by taking advantage of his/her IMD, this action will be recorded in the log for investigation. We can assume licensed doctors are trustworthy and hospitals are a safe working environment. These

are reasonable assumptions to make since government regulatory agencies will oversee hospitals and behaviours of doctors in the hospital.

We may design the IMD security in an environment of a Wearable and Implantable Body Sensor Network (WIBSN), and trust the gateway of the network. The IMD can offload its security related work to the gateway. It is similar to the external base station based security solutions [2, 1, 23, 29]. In this way, a secure pairing protocol is required between the IMD and the gateway. Some physiological(ECG) signal based security schemes assume that the patient can detect physical contact on the body from adversaries [25, 24, 30].

7.2.2 Decoupled Design

We propose to use the concept of a decoupled design as the IMD is a complex medical system. With a decoupled design, each component of a system works independently and any changes to one component will have minimal effect on the others. Current IMD products have been approved by government agencies(e.g., FDA), and are used by patients. So, we can assume the entire IMD primary module as a subsystem and design its security module as another subsystem. Merging both of the subsystems can obtain the next generation IMD products. Our goal is to design a decoupling or loose coupling IMD security system in terms of hardware (e.g., power system, processor, memory) and software. Ideally, the security subsystem does not require any modifications to the current version of the IMD, leading to a fully decoupled design. Adopting the decoupled design for IMD security schemes has several advantages as highlighted below:

- Using decoupled design will reduce the complexity of the next generation IMD product. If the added security module is decoupled from the current IMD system, we can just focus on designing and manufacturing the security part. If it is loosely coupled, we need to be careful with the interfaces between the security module and

the current IMD system. In this way, it will save time and rework of the whole product design.

- This design reduces risks of IMD recalls. The root cause of many quality, manufacturing, and performance problems can be traced to undesirable interactions between various components or systems of a product design [110]. If the IMD has problems related to safety, quality, efficacy or presentation, it has to be recalled. Using a decoupled security module, it requires minimal changes to the current approved IMD system. So, it will introduce fewer problems to the IMD primary module. Most of issues would be from the security module and its interface with the IMD primary module.

- It will help to get the new generation IMD product approved by government agencies. The review of a new IMD by the agencies may focus on the security part and its changes to the current IMD, not on the whole IMD product rigorously again. So manufacturers will prefer this approach when designing their next generation IMDs.

Some proposed IMD security solutions are in line with this decoupling concept. The external proxy based solutions run security protocols in an external device, e.g., the IMD Shield [1], MedMon [29]. The study in [4] proposed a security core (SISC) which is decoupled from the primary IMD core. It has its own instruction and data memory blocks and an independent power source from the RF energy harvesting. Thus this dual-core design can be used to counter battery depletion attacks. Therefore, the decoupled design benefits the whole IMD product and would be adopted.

7.2.3 Safety Overrides Security

The main function of an IMD is to cure ailments and maintain patient's health. Hence, the safety and utility of an IMD has a higher priority than its privacy and se-

curity requirements. As advised by the FDA, manufacturers should carefully consider the balance between security safeguards and the usability of the IMD for medical purposes [20]. One example is to guarantee access to the IMD by unauthorized doctors during an emergency situation [5]. This is a key challenge and has stimulated a few proposals [12, 3, 1, 2, 29, 24, 25, 30]. Nonetheless, some drawbacks of these schemes, hinder their application to current IMD products. Thus, we still need researchers to explore innovative solutions from different perspectives.

Another main design concern related to safety is that the major part of IMD resources should be allocated for supporting IMD medical functionality. So, the designers must carefully weigh costs arising from security algorithms against the safety and utility capabilities of the IMD. Some strong security protocols could be prohibitively expensive in terms of computation and communications. This may drain significant amounts of energy of the IMD and cause frequent surgical operations for device replacement [111]. Adversaries may even easily launch power DoS attacks by just constantly waking up the IMDs. Current proposals try to address this issue by adopting methodologies of using lightweight security algorithms, harvesting RF energy or using a separate security unit. Our goal is to design a security solution optimized to use as fewer resources of the IMD as possible. To design such an elegant security solution requires a comprehensive threat modeling framework with sound underpinning assumptions.

7.2.4 IMD Security Framework

The IMD security design has complicated requirements which can not be solved by relying on one single security solution. So, we need to design a systematic security framework to secure IMDs. A viable security scheme should consider applications in both the emergency situation and the normal circumstance as well as the constraints of IMD resources. A cryptographic audit log can be used to record all critical parameter changes

and data accesses in the IMD, so doctors can review the log when any anomaly is detected [112].

On the other hand, the IMD security issue involves a wide range of stakeholders consisting of patients and their family, hospitals and doctors, manufacturers and their engineers and government regulation agencies (e.g., FDA). We can establish and offer an information security awareness program in order to educate all stakeholders on the use of IMDs, such as patients and their family members, doctors and other related employees in a hospital [113]. Such a program will endeavor to impart skills that are necessary to operate the IMD in a correct fashion and save/transmit sensitive medical data in a secure way. We can also adopt a patient awareness mechanism into the system, such as the user alert scheme. So, the IMD could issue a notification (visual, audible or tactile) whenever it establishes a wireless connection with an external programmer or whenever a critical setting changes [5]. Furthermore, a practical solution should be designed specifically for each type of IMDs. This is because each type of IMD is designed and fabricated for a specific purpose, e.g., pacemakers for treating cardiac diseases, insulin pumps for treating diabetes and deep brain neurostimulators for treating Parkinson's diseases. Besides, stringent requirements from government agencies can be a power to drive manufacturers developing innovative and practical IMD cybersecurity controls [20].

7.3 Future Work

Currently several solutions have been proposed for the the trade-off between security and accessibility in the emergency situation. With these solutions, doctors who are not pre-authorized can have access to the IMDs for emergency treatment. Nonetheless, it is not clear which one would be optimal and practical from the engineering point of view. A proper solution should consider strict resource limitations of the IMD and its

medical application purposes. Besides, it should also be easy to operate by doctors and patients. There are standards and regulations designed for wireless medical communications, e.g. IEEE 802.15.6-2012: Wireless Body Area Networks [114]. These standards should be followed in order for any proposed solutions to be compatible with other medical devices. Considering that there are many different types of IMDs in the market, the designed security module should bring about minimal changes to the current IMD system specifications.

In addition, as analyzed in Chapter 2, there are two more trade-offs that we need to consider in the IMD security design: emergency access vs. normal access, and strong security vs. limited resources. Current solutions are primarily designed for the trade-off of security vs. accessibility, which normally consume extra resources for the emergency access. However, the patient's normal visit to the hospital happens regularly and should avoid this extra resource consumption of the IMD, especially its battery. So, we suggest the design of an appropriate key distribution scheme for supporting this kind of access. Likewise, the research on energy supply of the IMD, such as wireless charging and RF energy harvesting, can help to mitigate its battery limitations.

In summary, the future research should focus on the design of practical security solutions for the IMDs. The solutions need to balance the privacy and security of the IMD with its safety and utility. In the design, we need to consider the limitations of the IMD resources and its specific medical applications. The designed system should also be easily operated by doctors and patients, especially in the time-critical emergency treatment.

Abbreviations

BS Binary Sequence

BF Binary Feature

DoS Denial of Service

DWT Discrete Wavelet Transforms

ECG Electrocardiogram

ECC Error Correcting Code

EDE ECG-based Data Encryption

FAR False Acceptance Rate

FDA Food and Drug Administration

FFT Fast Fourier Transform

FRR False Rejection Rate

HTER Half Total Error Rate

IMD Implantable Medical Device

ICD Implantable Cardiac Defibrillator

IPI Inter-pulse-Interval

KD Key Distribution

MFBSG Multiple Fiducial-points based Binary Sequence Generation

OTP One-Time Pad

PRNG Pseudo-Random Number Generator

RSSI Received Signal Strength Indicator

SISC Smart-Implant Security Core

TOA Time of Arrival

WBAN Wireless Body Area Network

WIBSN Wearable and Implantable Body Sensor Network

WT Wavelet Transforms

WSN Wireless Sensor Network

Bibliography

[1] S. Gollakota, H. Hassanieh, B. Ransford, D. Katabi, and K. Fu, "They can hear your heartbeats: Non-invasive security for implantable medical devices," in *Proceedings of the ACM SIGCOMM 2011 Conference*, ser. SIGCOMM '11. New York, USA: ACM, 2011, pp. 2–13.

[2] F. Xu, Z. Qin, C. Tan, B. Wang, and Q. Li, "IMDGuard: Securing implantable medical devices with the external wearable guardian," in *INFOCOM, 2011 Proceedings IEEE*, April 2011, pp. 1862–1870.

[3] K. B. Rasmussen, C. Castelluccia, T. S. Heydt-Benjamin, and S. Capkun, "Proximity-based access control for implantable medical devices," in *Proceedings of the 16th ACM conference on Computer and Communications Security*. ACM, 2009, pp. 410–419.

[4] C. Strydis, R. M. Seepers, P. Peris-Lopez, D. Siskos, and I. Sourdis, "A system architecture, processor, and communication protocol for secure implants," *ACM Transactions on Architecture and Code Optimization (TACO)*, vol. 10, no. 4, p. 57, 2013.

[5] D. Halperin, T. Kohno, T. S. Heydt-Benjamin, K. Fu, and W. H. Maisel, "Security and privacy for implantable medical devices," *Pervasive Computing, IEEE*, vol. 7, no. 1, pp. 30–39, 2008.

[6] Intensive Care Hotline, "What is a pacemaker?" accessed: 2016-01-20. [Online]. Available: http://intensivecarehotline.com/pacemaker/

[7] Therapeutic Goods Administration, Australia, "InterStim and InterStim II neurostimulation devices used for sacral nerve stimulation," January 2015, accessed: 2016-01-20. [Online]. Available: https://www.tga.gov.au/alert/interstim-and-interstim-ii-neurostimulation-devices-used-sacral-nerve-stimulation

[8] Medtronic Australasia Pty Ltd, "What is insulin pump therapy and how does it work?" accessed: 2016-01-20. [Online]. Available: https://www.medtronic-diabetes.com.au/pump-therapy/what-is-insulin-pump-therapy

[9] Cochlear Ltd, "Cochlear implants & cochlear implant technology," accessed: 2016-01-20. [Online]. Available: http://www.cochlear.com/wps/wcm/connect/au/home/understand/hearing-and-hl/hl-treatments/cochlear-implant

[10] G. Fang, M. A. Orgun, R. Shankaran, E. Dutkiewicz, and G. Zheng, "Truthful channel sharing for self coexistence of overlapping medical body area networks," *PLoS One*, vol. 11, no. 2, p. e0148376, 02 2016.

[11] Medtronic, "Carelink remote monitoring network," accessed: 2016-01-01. [Online]. Available: http://www.medtronic.com/patients/bradycardia/pacemaker/our-pacemakers/carelink/index.htm

[12] D. Halperin, T. S. Heydt-Benjamin, B. Ransford, S. S. Clark, B. Defend, W. Morgan, K. Fu, T. Kohno, and W. H. Maisel, "Pacemakers and implantable cardiac defibrillators: Software radio attacks and zero-power defenses," in *Proceedings of the 2008 IEEE Symposium on Security and Privacy*, ser. SP '08. Washington, DC, USA: IEEE Computer Society, 2008, pp. 129–142.

[13] C. Li, A. Raghunathan, and N. K. Jha, "Hijacking an insulin pump: Security attacks and defenses for a diabetes therapy system," in *e-Health Networking Applications and Services (Healthcom), 2011 13th IEEE International Conference on.* IEEE, 2011, pp. 150–156.

[14] J. Radcliffe, "Hacking medical devices for fun and insulin: Breaking the human scada system," in *Black Hat Conference presentation slides*, 2011.

[15] G. Zheng, G. Fang, R. Shankaran, M. Orgun, J. Zhou, L. Qiao, and K. Saleem, "Multiple ECG fiducial points based random binary sequence generation for securing wireless body area networks," *IEEE Journal of Biomedical and Health Informatics*, vol. PP, no. 99, pp. 1–1, 2016.

[16] T. Denning, K. Fu, and T. Kohno, "Absence makes the heart grow fonder: New directions for implantable medical device security," in *Proceedings of the 3rd Conference on Hot Topics in Security*, ser. HOTSEC'08. Berkeley, CA, USA: USENIX Association, 2008, pp. 5:1–5:7.

[17] X. Hei, X. Du, J. Wu, and F. Hu, "Defending resource depletion attacks on implantable medical devices," in *Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE*, Dec 2010, pp. 1–5.

[18] A. Ibaida and I. Khalil, "Wavelet-based ECG steganography for protecting patient confidential information in point-of-care systems," *Biomedical Engineering, IEEE Transactions on*, vol. 60, no. 12, pp. 3322–3330, 2013.

[19] Y. Poullet, "EU data protection policy. the directive 95/46/EC: Ten years after," *Computer Law & Security Review*, vol. 22, no. 3, pp. 206–217, 2006.

[20] US Food and Drug Administration (FDA), "Content of premarket submissions for management of cybersecurity in medical devices: draft

guidance for industry and food and drug administration staff," October 2014. [Online]. Available: http://www.fda.gov/downloads/medicaldevices/deviceregulationandguidance/guidancedocuments/ucm356190.pdf

[21] U.S. FDA, "Vulnerabilities of hospira lifecare PCA3 and PCA5 infusion pump systems: FDA safety communication," May 2015, accessed: 2016-01-01. [Online]. Available: http://www.fda.gov/medicaldevices/safety/alertsandnotices/ucm446809.htm

[22] G. Kolata, "Of fact, fiction and Cheneys defibrillator," October 2013, accessed: 2016-01-01. [Online]. Available: http://www.nytimes.com/2013/10/29/science/of-fact-fiction-and-defibrillators.html

[23] G. Zheng, G. Fang, M. Orgun, and R. Shankaran, "A non-key based security scheme supporting emergency treatment of wireless implants," in *Communications (ICC), 2014 IEEE International Conference on*. IEEE, 2014, pp. 647–652.

[24] X. Hei and X. Du, "Biometric-based two-level secure access control for implantable medical devices during emergencies," in *INFOCOM, 2011 Proceedings IEEE*. IEEE, 2011, pp. 346–350.

[25] M. Rostami, A. Juels, and F. Koushanfar, "Heart-to-heart (H2H): authentication for implanted medical devices," in *Proceedings of the 2013 ACM SIGSAC conference on Computer and Communications Security*, ser. CCS '13, 2013, pp. 1099–1112.

[26] M. B. Alam, M. B. Munir, R. Rattan, S. Flanigan, E. Adelstein, S. Jain, and S. Saba, "Battery longevity in cardiac resynchronization therapy implantable cardioverter defibrillators," *EP Europace*, vol. 16, no. 2, pp. 246–251, 2014.

[27] K. M. Silay, C. Dehollain, and M. Declercq, "A closed-loop remote powering link for wireless cortical implants," *Sensors Journal, IEEE*, vol. 13, no. 9, pp. 3226–3235, 2013.

[28] R.-F. Xue, K.-W. Cheng, and M. Je, "High-efficiency wireless power transfer for biomedical implants by optimal resonant load transformation," *Circuits and Systems I: Regular Papers, IEEE Transactions on*, vol. 60, no. 4, pp. 867–874, 2013.

[29] M. Zhang, A. Raghunathan, and N. Jha, "MedMon: Securing medical devices through wireless monitoring and anomaly detection," *Biomedical Circuits and Systems, IEEE Transactions on*, vol. 7, no. 6, pp. 871–881, Dec 2013.

[30] G. Zheng, G. Fang, R. Shankaran, and M. A. Orgun, "Encryption for implantable medical devices using modified one-time pads," *Access, IEEE*, vol. 3, pp. 825–836, 2015.

[31] T. Denning, A. Borning, B. Friedman, B. T. Gill, T. Kohno, and W. H. Maisel, "Patients, pacemakers, and implantable defibrillators: Human values and security for wireless implantable medical devices," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2010, pp. 917–926.

[32] S. Schechter, "Security that is meant to be skin deep: Using ultraviolet micropigmentation to store emergency-access keys for implantable medical devices," Microsoft Research, Tech. Rep. MSR-TR-2010-33, April 2010. [Online]. Available: http://research.microsoft.com/apps/pubs/default.aspx?id=122137

[33] S. Mathur, W. Trappe, N. Mandayam, C. Ye, and A. Reznik, "Radio-telepathy: Extracting a secret key from an unauthenticated wireless channel," in *Proceedings of the 14th ACM International Conference on Mobile Computing and Networking*, ser. MobiCom '08. New York, NY, USA: ACM, 2008, pp. 128–139.

[34] H. Hwang, G. Jung, K. Sohn, and S. Park, "A study on MITM (man in the middle) vulnerability in wireless network using 802.1 x and eap," in *Information Science and Security, 2008. ICISS. International Conference on.* IEEE, 2008, pp. 164–170.

[35] S.-D. Bao, C. Poon, Y.-T. Zhang, and L. feng Shen, "Using the timing information of heartbeats as an entity identifier to secure body sensor network," *Information Technology in Biomedicine, IEEE Transactions on*, vol. 12, no. 6, pp. 772–779, Nov 2008.

[36] S.-D. Bao, "A matching performance study on IPI-based entity identifiers for body sensor network security," in *Biomedical Engineering and Informatics (BMEI), 2012 5th International Conference on*, Oct 2012, pp. 808–811.

[37] A. Jain, A. Ross, and S. Pankanti, "Biometrics: a tool for information security," *Information Forensics and Security, IEEE Transactions on*, vol. 1, no. 2, pp. 125–143, June 2006.

[38] S. Rane, Y. Wang, S. Draper, and P. Ishwar, "Secure biometrics: Concepts, authentication architectures, and challenges," *Signal Processing Magazine, IEEE*, vol. 30, no. 5, pp. 51–64, Sept 2013.

[39] J. L. Berger, "Medical implant device with RFID tag and method of identification of device," Feb. 19 2008, US Patent 7,333,013.

[40] P. D. Bradley, "An ultra low power, high performance medical implant communication system (MICS) transceiver for implantable devices," in *Biomedical Circuits and Systems Conference, BioCAS IEEE.* IEEE, 2006, pp. 158–161.

[41] L. OTT, "The evolution of bluetooth® in wireless medical devices," *Socket Mobile, Inc. White Papers*, 2010, accessed: 2016-02-01. [Online].

Available: https://www.socketmobile.com/docs/default-source/white-papers/socket_bluetooth-medical_white-paper.pdf?sfvrsn=2

[42] J. McCarthy, "Headphones may hinder pacemakers," November 2008, accessed: 2016-01-01. [Online]. Available: http://www.irishhealth.com/article.html?id=14596

[43] E. Corndorf, "Secure telemetric link," U.S.A. Patent US 7 930 543 B2, April, 2011. [Online]. Available: https://www.google.com/patents/US7930543

[44] M. Zhang, A. Raghunathan, and N. K. Jha, "Trustworthiness of medical devices and body area networks," *Proceedings of the IEEE*, vol. 102, no. 8, pp. 1174–1188, 2014.

[45] M. McLoone and M. J. Robshaw, "Public key cryptography and RFID tags," in *Topics in Cryptology–CT-RSA 2007*. Springer, 2006, pp. 372–384.

[46] A. Perrig, J. Stankovic, and D. Wagner, "Security in wireless sensor networks," *Commun. ACM*, vol. 47, no. 6, pp. 53–57, Jun. 2004.

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

[48] G. Zheng, G. Fang, R. Shankaran, M. Orgun, and E. Dutkiewicz, "An ECG-based secret data sharing scheme supporting emergency treatment of implantable medical devices," in *Wireless Personal Multimedia Communications (WPMC), 2014 International Symposium on*. IEEE, 2014, pp. 624–628.

[49] G. Zheng, G. Fang, M. A. Orgun, and R. Shankaran, "A comparison of key distribution schemes using fuzzy commitment and fuzzy vault within wireless body area

networks," in *Personal, Indoor, and Mobile Radio Communications (PIMRC), 2015 IEEE 26th Annual International Symposium on.* IEEE, 2015, pp. 2120–2125.

[50] A. Juels and M. Sudan, "A fuzzy vault scheme," *Designs, Codes and Cryptography*, vol. 38, no. 2, pp. 237–257, 2006.

[51] A. Juels and M. Wattenberg, "A fuzzy commitment scheme," in *Proceedings of the 6th ACM conference on Computer and communications security.* ACM, 1999, pp. 28–36.

[52] S. Jana, S. N. Premnath, M. Clark, S. K. Kasera, N. Patwari, and S. V. Krishnamurthy, "On the effectiveness of secret key extraction from wireless signal strength in real environments," in *Proceedings of the 15th annual international conference on Mobile computing and networking.* ACM, 2009, pp. 321–332.

[53] S. T. Ali, V. Sivaraman, and D. Ostry, "Eliminating reconciliation cost in secret key generation for body-worn health monitoring devices," *IEEE Transactions on Mobile Computing*, vol. 13, no. 12, pp. 2763–2776, 2014.

[54] L. F. Cranor and S. Garfinkel, *Security and usability: designing secure systems that people can use.* " O'Reilly Media, Inc.", 2005.

[55] S. Hosseini-Khayat, "A lightweight security protocol for ultra-low power asic implementation for wireless implantable medical devices," in *Medical Information & Communication Technology (ISMICT), 2011 5th International Symposium on.* IEEE, 2011, pp. 6–9.

[56] C. Strydis, D. Zhu, and G. N. Gaydadjiev, "Profiling of symmetric-encryption algorithms for a novel biomedical-implant architecture," in *Proceedings of the 5th Conference on Computing Frontiers*, ser. CF '08. New York, NY, USA: ACM, 2008, pp. 231–240.

[57] G. Contreras, M. Martonosi, J. Peng, G.-Y. Lueh, and R. Ju, "The xtrem power and performance simulator for the intel xscale core: Design and experiences," *ACM Trans. Embed. Comput. Syst.*, vol. 6, no. 1, Feb. 2007.

[58] H. Ohta and M. Matsui, "A description of the MISTY1 encryption algorithm," United States, Tech. Rep. RFC2994, 2000.

[59] R. Zimmermann, A. Curiger, H. Bonnenberg, H. Kaeslin, N. Felber, and W. Fichtner, "A 177 mb/s vlsi implementation of the international data encryption algorithm," *Solid-State Circuits, IEEE Journal of*, vol. 29, no. 3, pp. 303–307, 1994.

[60] R. L. Rivest, M. Robshaw, R. Sidney, and Y. L. Yin, "The RC6^{TM} block cipher," in *First Advanced Encryption Standard (AES) Conference*, 1998.

[61] C. Beck, D. Masny, W. Geiselmann, and G. Bretthauer, "Block cipher based security for severely resource-constrained implantable medical devices," in *Proceedings of the 4th International Symposium on Applied Sciences in Biomedical and Communication Technologies*, ser. ISABEL '11. New York, NY, USA: ACM, 2011, pp. 62:1–62:5.

[62] N. Ellouze, M. Allouche, H. Ben Ahmed, S. Rekhis, and N. Boudriga, "Securing implantable cardiac medical devices: use of radio frequency energy harvesting," in *Proceedings of the 3rd international workshop on Trustworthy embedded devices*. ACM, 2013, pp. 35–42.

[63] P. D. Mitcheson, "Energy harvesting for human wearable and implantable biosensors," in *Engineering in Medicine and Biology Society (EMBC), 2010 Annual International Conference of the IEEE*. IEEE, 2010, pp. 3432–3436.

[64] J. Olivo, S. Carrara, and G. De Micheli, "Energy harvesting and remote powering for implantable biosensors," *IEEE Sensors Journal*, vol. 11, no. EPFL-ARTICLE-152140, pp. 1573–1586, 2011.

[65] "Activa RC neurostimulator for deep brain stimulation," accessed: 2016-03-01. [Online]. Available: https://professional.medtronic.com/pt/neuro/dbs-md/prod/activa-rc/index.htm

[66] P. Li and R. Bashirullah, "A wireless power interface for rechargeable battery operated medical implants," *Circuits and Systems II: Express Briefs, IEEE Transactions on*, vol. 54, no. 10, pp. 912–916, Oct 2007.

[67] A. K. RamRakhyani, S. Mirabbasi, and M. Chiao, "Design and optimization of resonance-based efficient wireless power delivery systems for biomedical implants," *Biomedical Circuits and Systems, IEEE Transactions on*, vol. 5, no. 1, pp. 48–63, 2011.

[68] P. Mitcheson, E. Yeatman, G. Rao, A. Holmes, and T. Green, "Energy harvesting from human and machine motion for wireless electronic devices," *Proceedings of the IEEE*, vol. 96, no. 9, pp. 1457–1486, Sept 2008.

[69] P. Failla, Y. Sutcu, and M. Barni, "Esketch: a privacy-preserving fuzzy commitment scheme for authentication using encrypted biometrics," in *Proceedings of the 12th ACM Workshop on Multimedia and Security*. ACM, 2010, pp. 241–246.

[70] A. L. Goldberger, L. A. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C.-K. Peng, and H. E. Stanley, "Physiobank, physiotoolkit, and physionet components of a new research resource for complex physiologic signals," *Circulation*, vol. 101, no. 23, pp. e215–e220, 2000.

[71] C. E. Shannon, "A mathematical theory of communication," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 5, no. 1, pp. 3–55, 2001.

[72] S. Safie, J. J. Soraghan, L. Petropoulakis *et al.*, "Electrocardiogram (ECG) biometric authentication using pulse active ratio (PAR)," *Information Forensics and Security, IEEE Transactions on*, vol. 6, no. 4, pp. 1315–1322, 2011.

[73] V. D. Goppa, "A new class of linear correcting codes," *Problemy Peredachi Informatsii*, vol. 6, no. 3, pp. 24–30, 1970.

[74] R. J. McEliece and D. V. Sarwate, "On sharing secrets and reed-solomon codes," *Communications of the ACM*, vol. 24, no. 9, pp. 583–584, 1981.

[75] R. C. Bose and D. K. Ray-Chaudhuri, "On a class of error correcting binary group codes," *Information and control*, vol. 3, no. 1, pp. 68–79, 1960.

[76] C. E. Shannon, "Communication theory of secrecy systems," *Bell system technical journal*, vol. 28, no. 4, pp. 656–715, 1949.

[77] A. Rukhin, J. Soto, J. Nechvatal, M. Smid, and E. Barker, "A statistical test suite for random and pseudorandom number generators for cryptographic applications," Nat. Inst. Standards Technol., Gaithersburg, MD, USA, Tech. Rep., 2010.

[78] A. Taddei, G. Distante, M. Emdin, P. Pisani, G. Moody, C. Zeelenberg, and C. Marchesi, "The european ST-T database: standard for evaluating systems for the analysis of st-t changes in ambulatory electrocardiography," *European heart journal*, vol. 13, no. 9, pp. 1164–1172, 1992.

[79] F. Miao, S.-D. Bao, and Y. Li, "A modified fuzzy vault scheme for biometrics-based body sensor networks security," in *Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE*, Dec 2010, pp. 1–5.

[80] L. Yao, B. Liu, K. Yao, G. Wu, and J. Wang, "An ECG-based signal key establishment protocol in body area networks," in *Ubiquitous Intelligence Computing and 7th International Conference on Autonomic Trusted Computing (UIC/ATC), 2010 7th International Conference on*, Oct 2010, pp. 233–238.

[81] C. Hu, X. Cheng, F. Zhang, D. Wu, X. Liao, and D. Chen, "OPFKA: Secure and efficient ordered-physiological-feature-based key agreement for wireless body area networks," in *INFOCOM, 2013 Proceedings IEEE*, April 2013, pp. 2274–2282.

[82] G. Zheng, G. Fang, M. Orgun, R. Shankaran, and E. Dutkiewicz, "Securing wireless medical implants using an ECG-based secret data sharing scheme," in *Communications and Information Technologies (ISCIT), 2014 14th International Symposium on*. IEEE, 2014, pp. 373–377.

[83] M. Borowski and M. Lesniewicz, "Modern usage of "old" one-time pad," in *Communications and Information Systems Conference (MCC), 2012 Military*, Oct 2012, pp. 1–5.

[84] F.-G. Deng and G. L. Long, "Secure direct communication with a quantum one-time pad," *Physical Review A*, vol. 69, no. 5, p. 052319, 2004.

[85] R. Horstmeyer, B. Judkewitz, C. Yang, and I. M. Vellekoop, "Physical key-protected one time pad," Feb. 21 2013, uS Patent App. 13/773,490.

[86] Z. Zhang, H. Wang, A. V. Vasilakos, and H. Fang, "ECG-cryptography and authentication in body area networks," *Information Technology in Biomedicine, IEEE Transactions on*, vol. 16, no. 6, pp. 1070–1078, 2012.

[87] K. Venkatasubramanian, Venkatasubramanian, A. Banerjee, and S. Gupta, "EKG-based key agreement in body sensor networks," in *INFOCOM Workshops 2008, IEEE*, April 2008, pp. 1–6.

[88] Y. Dodis, L. Reyzin, and A. Smith, "Fuzzy extractors: How to generate strong keys from biometrics and other noisy data," in *Advances in cryptology-Eurocrypt 2004*. Springer, 2004, pp. 523–540.

[89] K. Lee, J. C. Eidson, H. Weibel, and D. Mohl, "IEEE 1588-standard for a precision clock synchronization protocol for networked measurement and control systems," in *Conference on IEEE*, vol. 1588, 2005, p. 2.

[90] A. Wander, N. Gura, H. Eberle, V. Gupta, and S. Shantz, "Energy analysis of public-key cryptography for wireless sensor networks," in *Pervasive Computing and Communications, 2005. PerCom 2005. Third IEEE International Conference on*, March 2005, pp. 324–328.

[91] G.-H. Zhang, C. Poon, and Y.-T. Zhang, "Analysis of using interpulse intervals to generate 128-bit biometric random binary sequences for securing wireless body sensor networks," *Information Technology in Biomedicine, IEEE Transactions on*, vol. 16, no. 1, pp. 176–182, Jan 2012.

[92] L. Sörnmo and P. Laguna, "Electrocardiogram (ECG) signal processing," *Wiley Encyclopedia of Biomedical Engineering*, 2006.

[93] S. Mahmoodabadi, A. Ahmadian, M. Abolhasani, M. Eslami, and J. Bidgoli, "ECG feature extraction based on multiresolution wavelet transform," in *Engineering in Medicine and Biology Society, 2005. IEEE-EMBS 2005. 27th Annual International Conference of the.* IEEE, 2006, pp. 3902–3905.

[94] F. Sufi, I. Khalil, and J. Hu, "ECG-based authentication," in *Handbook of Information and Communication Security.* Springer, 2010, pp. 309–331.

[95] L. Biel, O. Pettersson, L. Philipson, and P. Wide, "ECG analysis: a new approach in human identification," *Instrumentation and Measurement, IEEE Transactions on*, vol. 50, no. 3, pp. 808–812, Jun 2001.

[96] H. Garcia-Baleon, V. Alarcon-Aquino, and O. Starostenko, "A wavelet-based 128-bit key generator using electrocardiogram signals," in *Circuits and Systems, 2009. MWSCAS '09. 52nd IEEE International Midwest Symposium on*, Aug 2009, pp. 644–647.

[97] H. Garcia-Baleon and V. Alarcon-Aquino, "Cryptographic key generation from biometric data using wavelets," in *Electronics, Robotics and Automotive Mechanics Conference, 2009. CERMA '09.*, Sept 2009, pp. 15–20.

[98] J. P. Martínez, R. Almeida, S. Olmos, A. P. Rocha, and P. Laguna, "A wavelet-based ECG delineator: evaluation on standard databases," *Biomedical Engineering, IEEE Transactions on*, vol. 51, no. 4, pp. 570–581, 2004.

[99] D. Cvetkovic, E. D. Übeyli, and I. Cosic, "Wavelet transform feature extraction from human PPG, ECG, and EEG signal responses to ELF PEMF exposures: A pilot study," *Digital signal processing*, vol. 18, no. 5, pp. 861–874, 2008.

[100] J. Sahambi, S. Tandon, and R. Bhatt, "Using wavelet transforms for ECG characterization - an on-line digital signal processing system," *Engineering in Medicine and Biology Magazine, IEEE*, vol. 16, no. 1, pp. 77–83, 1997.

[101] C. Li, C. Zheng, and C. Tai, "Detection of ECG characteristic points using wavelet transforms," *Biomedical Engineering, IEEE Transactions on*, vol. 42, no. 1, pp. 21–28, 1995.

[102] S. Mallat, "Zero-crossings of a wavelet transform," *Information Theory, IEEE Transactions on*, vol. 37, no. 4, pp. 1019–1033, Jul 1991.

[103] S. Mallat and S. Zhong, "Characterization of signals from multiscale edges," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 7, pp. 710–732, 1992.

[104] A. Ghaffari, M. Homaeinezhad, M. Akraminia, M. Atarod, and M. Daevaeiha, "A robust wavelet-based multi-lead electrocardiogram delineation algorithm," *Medical engineering & physics*, vol. 31, no. 10, pp. 1219–1227, 2009.

[105] N. Thakor, J. Webster, and W. J. Tompkins, "Estimation of QRS complex power spectra for design of a qrs filter," *Biomedical Engineering, IEEE Transactions on*, vol. BME-31, no. 11, pp. 702–706, Nov 1984.

[106] A. Daamouche, L. Hamami, N. Alajlan, and F. Melgani, "A wavelet optimization approach for ECG signal classification," *Biomedical Signal Processing and Control*, vol. 7, no. 4, pp. 342–349, 2012.

[107] S. Callegari, R. Rovatti, and G. Setti, "Embeddable adc-based true random number generator for cryptographic applications exploiting nonlinear signal processing and chaos," *Signal Processing, IEEE Transactions on*, vol. 53, no. 2, pp. 793–805, Feb 2005.

[108] G. Strang, "Wavelet transforms versus Fourier transforms," *Bulletin of the American Mathematical Society*, vol. 28, no. 2, pp. 288–305, 1993.

[109] F. Rincón, J. Recas, N. Khaled, and D. Atienza, "Development and evaluation of multilead wavelet-based ECG delineation algorithms for embedded wireless sensor nodes," *Information Technology in Biomedicine, IEEE Transactions on*, vol. 15, no. 6, pp. 854–863, 2011.

[110] H. W. Stoll, *Product design methods and practices*. CRC Press, 1999.

[111] W. Burleson, S. S. Clark, B. Ransford, and K. Fu, "Design challenges for secure implantable medical devices," in *Proceedings of the 49th Annual Design Automation Conference*. ACM, 2012, pp. 12–17.

[112] C. Camara, P. Peris-Lopez, and J. E. Tapiador, "Security and privacy issues in implantable medical devices: A comprehensive survey," *Journal of biomedical informatics*, vol. 55, pp. 272–289, 2015.

[113] C. McCoy and R. T. Fowler, "You are the key to security: establishing a successful security awareness program," in *Proceedings of the 32nd annual ACM SIGUCCS conference on User services*. ACM, 2004, pp. 346–349.

[114] "IEEE standard: 802.15.6-2012 - IEEE standard for local and metropolitan area networks - part 15.6: Wireless body area networks," accessed: 2016-01-20. [Online]. Available: https://standards.ieee.org/findstds/standard/802.15.6-2012.html