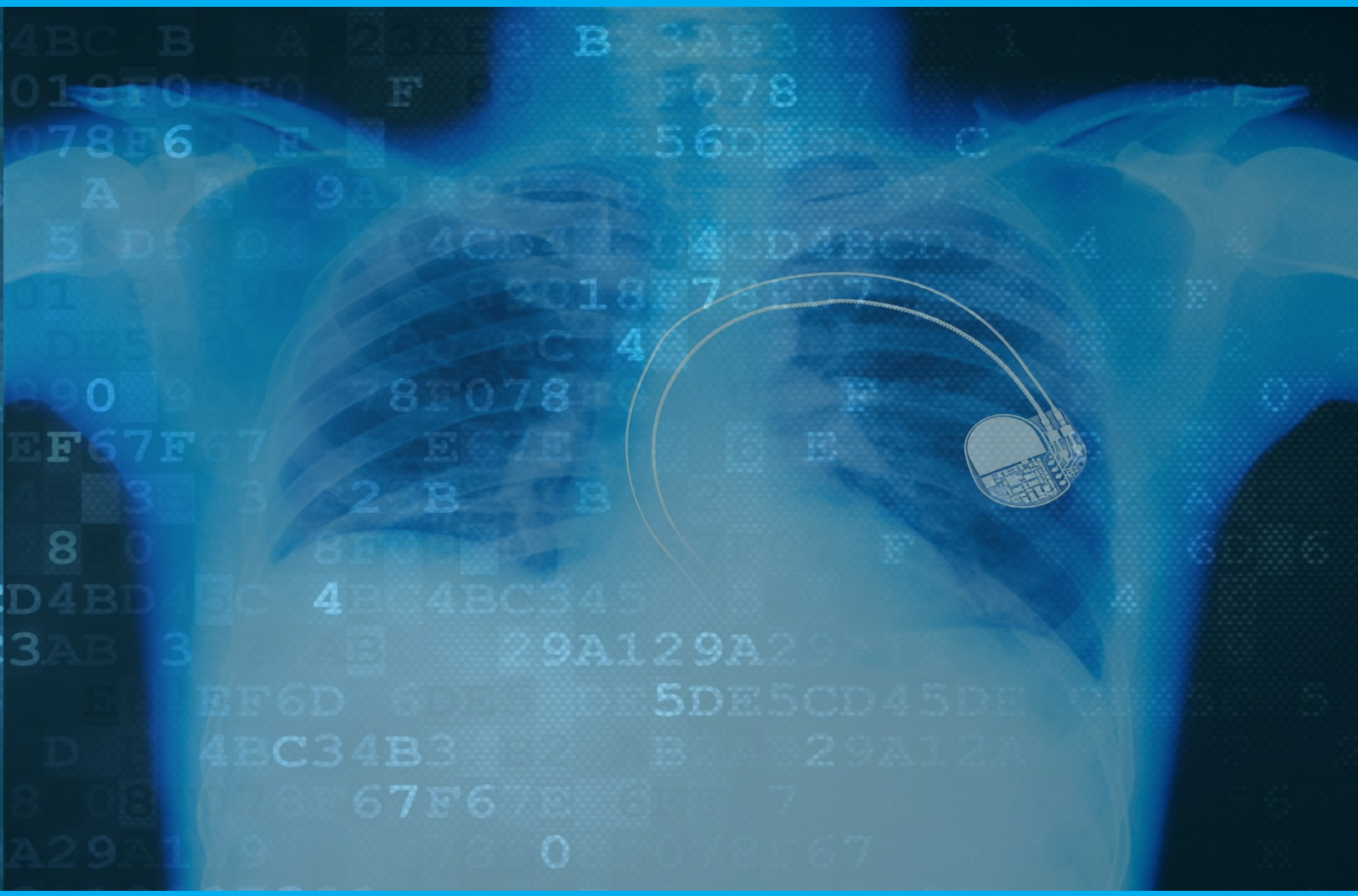
# Heartwear Security

## A lightweight security protocol for Implantable Medical Devices

G.H.J. Geneste

Delft University of Technology, Department Intelligent Systems



TU Delft

# Heartwear Security

## A lightweight security protocol for Implantable Medical Devices

by

G.H.J. Geneste

to obtain the degree of Master of Science in Computer Science
Software Technology track
with a specialisation in Cyber Security,
to be defended publicly on Thursday November 30, 2017 at 10:00 AM.

| | | |
|---|---|---|
| Student number: | 4081315 | |
| Project duration: | February 22, 2017 – November 23, 2017 | |
| Thesis committee: | Assoc. Prof. dr. ir. J. van der Lubbe, | TU Delft, chair |
| | Assist. Prof. Dr. Z. Erkin, | TU Delft, supervisor |
| | Assist. Prof. Dr. ir. J. Urbano, | TU Delft |
| | M. Deconinck, | Fox-IT |



TU Delft

# Preface

*Modern Implantable Medical Devices (IMDs) are enabled to communicate with an external device trough wireless channels. The non-invasive method allow health-care practitioners to investigate the effectiveness of a treatment delivered by the IMD and tailor it specifically to the patients' needs. The wireless communication capability however, also facilitates an adversary to gain illegitimate access to the live-saving medical device due to insufficient security measures to establish a secure communication channel. Since there is good reason to safeguard the IMD from prying eyes to avoid access to malicious entities, we present Heartwear, a lightweight security scheme to allow the establishment of a secure wireless communication channel between an IMD and legit external device. Heartwear takes advantage of heartbeat signals in order to provide both key establishment as well as unilateral authentication to protect against attacks of passive and active adversaries.*

A couple of years ago, I had the opportunity to study either Medicine at the Erasmus MC or Computer Science at the University of Technology in Delft. Both study directions appealed to me due to my interest in technology and sciences ever since I was a child. Although many people warned me for the complex environment I would find myself in, I choose for the bachelor Computer Science as I valued the desire to *create* above the possibility to *heal*. With this in mind, I worked hard to complete my bachelor. This fast evolving field allowed me to gain insights to the impact of technology in our society. While automation, robotics and data sciences bring many benefits, there is also a growing need for security solutions. Therefore, I decided to adhere to my curiosity and followed the specialisation in Cyber Security for the Master Computer Science. At the time, I did not expect this choice would empower me to push even further and combine my thesis with my silent interest in health care.

This Master thesis has been build upon the experience I gained throughout the years from very knowledgeable people who put their efforts in sharing their thoughts and findings with me. I am proud to see the final result of this journey that could not have been finished without the support of many people. First and foremost, I would like to thank my supervisor Zeki. He guided me the past nine months during our weekly meetings by providing feedback to keep me focused on the main topic of my thesis. More importantly, I am especially thankful for the conversations that made me not only to focus on the results, but also to improve myself as a person. A lot appreciation goes to the CybSec group, with whom I could always have interesting discussions on any topic during our social activities, lunch and weekly cake sessions.
I was also very fortunate to collaborate my thesis with Fox-IT. My daily supervisor Michel and the audits department enabled me to combine multiple aspects of cyber security into one thesis. Michel, thank you for connecting me with experts and the support throughout this process. Audits, I worship the creative ideas and motivation received on a daily basis during the past nine months which improved my work to make an actual difference.
I would however never been able to come this far without the sacrifices of my love Xander, who stood by my side since the start of my bachelor without hesitation. Xander did not only encourage me to go on if I wanted to give up, but he also assisted me in doing so with great patience. I am very grateful for all the infinite care, love and guidance which has brought me where I am standing now.
Sometimes I take a rough road to succeed, nevertheless I know my family will always have my back. Without any hesitation, they completely supported me for the study choices I have made. Special thanks are for my parents Sidney and Renee, and my sister Anne-Fleur who provided me the tools and the spirit to succeed.
Finally I want to thank my dear friends who provided distractions and good advice to relief me from the stressful times. You all brought so much fun to my life the past years. I sincerely wish this will never change, even after obtaining my Master degree.

*G.H.J. Geneste*
*Delft, November 2017*

# Contents

# 1 Introduction

*Implantable Medical Devices (IMDs) are systems that overcome life threatening conditions once inserted into the human body. In* 1958 *the first IMD, a pacemaker, was implanted to treat a chronic heart condition. The device lasted for three hours, the following model only for two days. The past 60 years IMDs evolved quickly into reliable cures for chronic diseases because they matured in lifetime, functionality and effectiveness. One of the notable improvements for IMDs has emerged in the* $20^{th}$ *century, where the functionality of the device was extended with wireless communication. This technological advancement enabled continuous monitoring and the change of treatments after implantation. Such improvements increase the variety of chronic diseases that can be treated to remedy more patients who suffer from life threatening conditions.*

The introduction of wireless communication for medical devices brings many benefits. With this technology, a health-care practitioner can connect with an implanted device without the need for physical access. This safe and non-invasive method allows him to investigate the effectiveness of a treatment by requesting data from the IMD. Upon inspection of the data, the health-care practitioner can decide to change the treatment specifically for the patient by reprogramming the IMD over the same wireless communication channel. This was all impossible without the introduction of wireless communication, since the IMD was then programmed before it is surgically implanted and never changed until it was explanted. With the current advancements in technology, IMDs can now provide a more variable treatment specifically tailored to an individual.

While the wireless technology can improve the delivery of a treatment, it also comes with potential hazards from a security perspective. The wireless communication channel transfers privacy sensitive data such as medical information and allows the transmission of commands from an external device to the IMD. Although the embedded system can be controlled to support the health of a human being, a user with harmful intent and access to the channel can also misuse this functionality to disable or even reprogram the device. Especially because of the critical information contained in the IMD, there is good reason to safeguard it from prying eyes to avoid access to malicious entities. Unfortunately current IMDs lack proper security features for protection against these attacks and allow a hacker to gain full control of the IMD, as will be explained further in Section 1.1.

The privacy and security concerns raise the need for a methodology which prevents unauthorised entities to access the IMD or eavesdrop on the information sent over the wireless communication channel. Securing the channel between an IMD and external device requires a cryptographic solution to ensure the confidentiality of the messages transferred. However, proposing such solutions also introduce the consumption of resources that may not be available. Additionally, the solution should not unnecessarily delay the access from an external programmer to an IMD in case of emergencies because harming the health of the patient should be avoided at all cost. Therefore the ideal solution requires a careful trade-off in which the privacy and security is maintained without affecting the main functionality of the IMD.

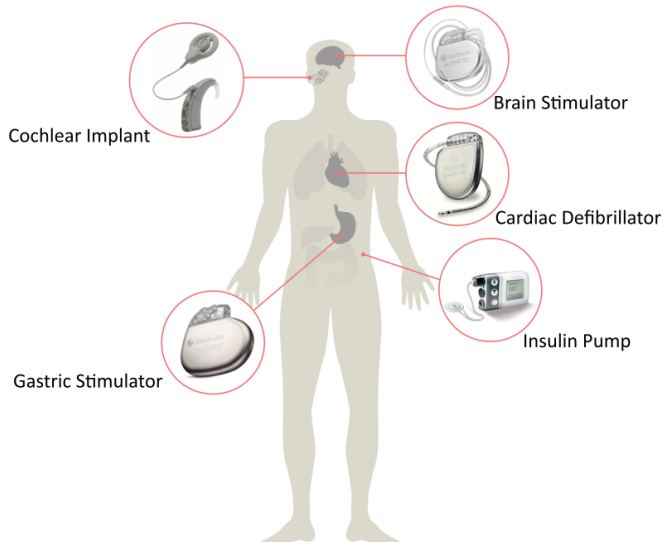



Figure 1.1: Examples of Implantable Medical Devices (IMD), image from [42]

Various security schemes have been proposed to eliminate these security concerns. As will be elaborated more thoroughly in Chapter 3, these solutions can be classified into proximity, proxy, biometric and third party based approaches. While each approach has their strengths and limitations, the biometric based approaches using heartbeat based security show promising results. Biometric based approaches depend on human characteristics which, if the correct biometric is used, are already measured by the IMD making it less complicated to incorporate into a security scheme. Despite its promising results however, heartbeat based security has so far only been used for either authentication or the support of key establishment and not to accomplish both goals into one security scheme.

This chapter gives a more thorough introduction to IMDs in general, explaining the current state of the wireless communication channel of IMDs. Thereafter a description of the security, safety and utility goals will be given, followed by the problem statement and contributions of this thesis. Finally a comprehensive outline of the document will be provided.

## 1.1. Implantable medical devices

IMDs are small devices inserted into the human body with the purpose to deliver a medical therapy. The architecture of the sophisticated devices differ per IMD but are in general as small as a penny containing a battery, embedded CPU, radio, sensor and actuator. Examples of a few IMDs are depicted in Figure 1.1: A modern implantable cardioverter defibrillator (ICD) is capable of administering an electrical shock to restore a normal heart rhythm. An infusion pump serves the purpose of delivering the right amount of drugs to a human at the right time. Implantable neurostimulators generate electrical impulses to provide pain relief and cochlear implants can improve hearing by those who suffer from extreme deafness. All these devices are examples of IMDs, which are capable to record the vital signs of the patient and in administering therapies to mitigate certain health critical conditions.

The battery-powered devices are implanted into the human body, a medical procedure that comes with certain risks. One of the main reasons for replacement of the devices is the status of the battery, a procedure that can only be performed surgically. In order to reduce these risk-involving and costly surgeries, the energy consumption of the device needs to be as low as possible. It therefore operates within an extreme lightweight environment, such that general IMDs are capable to last reliably and autonomously for approximately 10 years [14]. A more detailed explanation on the constrained environment of an IMD is discussed in Section 1.2.

## The need for telemetry

Modern IMDs now support wireless communication with external devices. This wireless connectivity to IMDs enable health care practitioners to remotely monitor their patient's vital signs and improve treatments based on the wireless received telemetry. More specifically, an external device supports these actions with the following functionality:

- **Receive telemetry:** An external device can read out sensory data for remote *monitoring*. Allowing the external device to validate the effectiveness of the current treatment, read out the history when treatment was applied and to detect anomalies.
- **Remote control:** Supports the functionality to *command* the IMD wireless in case of malfunction of current treatment, to reset the device and turn it off in case of emergencies.
- **Write access:** Allows an external device to *modify* treatment parameters, change functionality by updating the firmware and mitigate software bugs after implantation.

As example, the previously mentioned ICD can send an event of abnormal heart rhythms directly to a health care practitioner for analysis. The insulin pump can be programmed remotely to provide an additional amount of insulin and a neurostimulator can be adjusted with a hand held programmer. These medical devices all have a built-in functionality for wireless communication, making it cheaper, easier or more efficient to treat humans for a disorder.

The functionality of wireless communication has become important for this industry. To facilitate the wireless communication channel, a built-in radio is embedded in the system which sends and receives specific radio signals. Using radio frequencies (RF) however is not without risk. The range for an RF signal to be carried over air depends on the frequency used, and whenever there are multiple instances at the time the signal may be disturbed and becomes unreadable. To avoid such interference with other communication services, regulatory bodies have defined certain radio frequencies as standard telemetry transmission channel for IMDs:

- **Wireless Medical Telemetry Services (WMTS):** This standard operates in the 608 – 614, 1396 – 1400 and 1427 – 1432MHz radio frequency range. It has been introduced in 2002 because of the growing concern of interference with digital television. Specifications have been defined by the US Federal Communications Commission (FCC)[1].
- **Medical Implants Communication Services (MICS):** is a standard radio service for wireless communication between IMDs and external devices within a range of 2 meters. This standard has been defined by both the FCC as European Telecommunications Standards Institute (ETSI) in 2004 and operates on the 402 – 405 MHz band [2].

The standards defined by the FCC and ETSI describe the technical requirements for medical devices supporting this communication service. Non-enforceable guidelines have been developed by the US Food and Drug Administration (FDA)[3] to provide wireless technology considerations that can affect the safety and effectiveness of medical devices. While the discussed regulations and recommendations take safety and effectiveness into account, little deliberation had been given from a security perspective.

## Weakness of wireless communication in IMDs

By enabling an IMD to communicate over a wireless communication channel, adversaries interested to gain illegitimate control over an IMD theoretically also find additional entries to attack the IMD. But is it practically possible for someone with malicious intent to gain remote access to current implanted devices through its wireless communication interface?

[1] `https://www.fcc.gov/wireless/bureau-divisions/broadband-division/wireless-medical-telemetry-service-wmts`, Retrieved July 2017

[2] `http://www.etsi.org/deliver/etsi_en/301800_301899/30183901/01.03.01_60/en_30183901v010301p.pdf`, Retrieved July 2017

[3] `https://www.fda.gov/MedicalDevices/ucm077210.htm`, Retrieved June 2017

This question has been raised by Marie Moe [4], a security researcher who needed a pacemaker at young age and wanted to know if hackers can break her heart. It appeared to her that external devices have the ability to collect patient information from the pacemaker, sent over wireless communication channels. While only legitimate external devices should be able to connect, it was established that pacemakers and other IMDs *can be hacked*: it is possible to extract privacy sensitive information and even threatening a life by turning it off or changing the treatment.

Marie Moe was not the only one who investigated the security of wireless communication channels between IMDs and external devices: One of the first public investigations addressing the security and privacy properties of a commercial IMD was published in 2008 by Halperin *et al.* [21]. The researchers investigated an ICD by eavesdropping on the communication with off-the-shelf technology. It was discovered the ICD can be activated to sent telemetry data with by using a magnet, which initiated the *interrogation* state. From this state, the communication between a legitimate external device and ICD have been recorded during the execution of several commands. By simply replaying the recorded transmission on the 175kHz FM band, the researchers were able to reveal cardiac data, change the patient name and therapies without the need of a legit external device. Even without authentication, it was possible to command the ICD to release an electric shock.

Radcliffe investigated an insulin pump and found the device was vulnerable for both active and passive attacks [37]. By spending only $\$20,-$, an adversary could eavesdrop on the communication between the implanted sensor and insulin pump and use it for a replay attack. By impersonating the implanted sensor and sending old blood sugar data to the insulin pump can fool it to believe the blood sugar level is different than the actual value. When Radcliffe discovered the transmission format between the two devices and their independent unique identifier, he could gain control over the communication channel. An adversary could therefore fake sensor data and manipulate a diabetic into administering more insulin than needed, causing a hypoglycemic condition which potentially compromises the health of the patient.

The most recent study on the security of IMDs is performed by Marin *et al.* [32]. The work addresses both the short range communication channel on the $30-3000kHz$ band as well as the longer range channel on the earlier defined *MICS* band. The transmission between an external device and investigated ICD appeared to be obfuscated by XOR-ing the data with an output sequence from an LFSR. The researchers were able to reverse engineer the output sequence and found that the input sequence was the same for several different models. With this information the researchers could recover the privacy sensitive telemetry of different devices.
The same study discovered a vulnerability in the standby mode of the ICD, by replaying a specific message (identical for all investigated ICD models) the devices turned into an interrogation mode. The battery-life of the ICD can be drastically reduced by continuously forcing it into this mode because it actively polls for nearby external devices. An adversary could also successfully perform a replay attack by re-sending past transmissions. By combining the enforcement to interrogation mode and a replay attack, an adversary is able to gain complete control over the messages send on the long range communication channel.

## 1.2. Criteria for implantable medical devices

Given the results to what extent an IMD can be compromised, the question arises as to what solution will protect these devices sufficiently. A suitable solution needs to fit in the device and should take all criteria into account, which is a challenging effort due to the conflicting requirements:

The primary task of an IMD is to provide a treatment to a patient for its chronic disease. It is specifically designed to safeguard the health of the patient, a task that should not be interrupted. This important requirement however may be violated by the absence of security properties. As demonstrated by other researches, it is possible to attack an IMD and command it to stop or even change a treatment. A security solution can overcome such infringements by avoiding unauthorised access and ensure a higher level of safety. On the other hand, the introduction of a security solution also requires

[4] `https://www.wired.com/2016/03/go-ahead-hackers-break-heart/`, Retrieved July 2017

the consumption of additional resources. If these resources are not available or render the device to a non-functioning state, the safety of a patient can be endangered. It is therefore important to investigate the the safety, security and utility goals of the device before proposing a suitable security solution.

The safety, security and utility goals specify what should be satisfied, but not specifically how. Therefore, after defining these goals, the goals should be translated to more specific requirement definitions. These requirements support the derivation of a suitable solution by means of proposing a security scheme. After the proposal of such a solution, the performance of the scheme needs to be verified by reviewing whether the predefined goals are satisfied and by means of the execution of several experiments. The experiments and formal verification of the system will reveal to what extent the solution is practically feasible within the described landscape.

### 1.2.1. Security goals

In order to define a set of security goals, one needs to investigate what aspects of security are considered most important to be maintained by the future security scheme. These aspects are commonly derived by inspecting the protection of three different properties: confidentiality, integrity and availability. A security goal describes what an attacker should not be able to do, or what legitimate users only are allowed to do in terms of these properties. The preservation of these properties differ for each system because it depends on what scenarios the device needs to be protected against to. For this case specifically, the protection is described for each property separately in the context of IMD security. These properties together are referred to as the *CIA triad*, used first in [41], and is defined as follows:

- **Confidentiality**: Is the ability for an IMD or external device to ensure that (privacy) sensitive data (such as telemetry, cryptographic secrets, therapy settings, indication of medical condition of the patient) can only be viewed by authenticated parties. It should be impossible for an unauthenticated entity to read or even approximate the real content of the data sent over a wireless communication channel.
- **Integrity**: Describes the ability of an IMD to ensure that its functionality or data is modified only by authenticated parties. An unauthenticated external device should therefore not be able to change therapies or data on demand, regardless of whether this data is in transition or demanded by the unauthenticated device itself.
- **Availability**: Applies to the data, responsiveness and resources of the system. Throughout the lifetime of an IMD it should be continuously functional and responsive, especially in case of health critical events when an emergency caregiver must make quick decisions. In such emergency situations it may be necessary to change the therapy directly, reset the device or request the therapies delivered in the past. Additionally, the device should always be available to deliver its therapy under normal conditions, without the risk of having any of its resources depleted due to implementation fallacies or adversaries.

Preserving all three security properties is a tedious task. Security always comes at a certain cost, consuming resources such as energy, computing power, storage or possibly affects one of CIA properties negatively. Therefore mitigating all theoretically possible security issues is unfeasible due to the tensions among conflicting requirements. Thus defining security goals for a given application requires a careful investigation of the landscape the application operates in as well. The landscape of the application is often described within a threat model which may be derived from attacks shown to be practically feasible. While the topic of threat modelling will be elaborated in Section 2.1, according to the previously described practically possible attacks it is observed that the security scheme in general demands a solution which provides authentication in order to ensure a certain level of trust between an IMD and external device. A cryptographically secure solution would maintain the confidentiality and integrity goals. However, it should be tailored such that the availability goals are not compromised.

### 1.2.2. Safety and utility goals

It has been established the security goals should be designed based upon a certain set of defined attack vectors without introducing conflicts among the CIA properties. To complicate things further,

one also needs to take the safety and utility aspects of a system into account. While general solutions for preserving the confidentiality, integrity and availability on wireless communication channels have been addressed before, there exist tensions when applied into a highly constrained environment such as the IMD: General security solutions may require resources of the device that compromise the main functionality of the IMD making it useless to deploy. As a consequence, the desired security solution also depends on the environment the solution should operate in. These dependencies come forward by the definition of the safety and utility goals. It is therefore important to investigate the safety and utility goals to avoid proposing an unfeasible cryptographic solution.

For the IMD specifically, there exist four challenges due to the current landscape of the IMD. Each challenge identifies certain constraints or limitations that need to be taken into account for the design of a security scheme. These challenges follow to the definition of safety and utility goals as neglecting any of these challenges can impair the feasibility or safety of the security scheme. According to the architecture and landscape as identified in Section 1.1, the following challenges have been identified:

1. **Physical Critical Environment:** By definition, an IMD is inserted into the human body. This makes it very inconvenient to reach the device physically for events such as hardware changes or battery replacements. Thus the device needs to operate reliably and reduce the possibility of fails that demand physical access to resolve an issue. Other constraints require the device to be *bio compatible* [25]: The materials used should be resistant to the bodily fluids and not interfere negatively with the human body, otherwise it may cause the immune system to reject the device. Furthermore, thermal and RF radiation of the device needs to be minimised to avoid damage in the surrounding tissues. Lastly, the size of the device is also constrained to the space it will be implanted in, generally requiring it to be very small. Compliance with the physical critical environment therefore requires a security scheme to carefully investigate the implications of any change to the hardware architecture as it may affect the bio compatibility and reliability of the IMD and hence the health of the patient.

2. **Limited Resources:** Since the device requires compliance with the physical critical environment, an IMD is bound in the amount of components embodied in the device. For example, a physically large circuitry to support all functionality of the device may not fit into the organ the device is implanted in. Additionally, the device operates on battery-power and should last on average about 10 years. The implementation of a security scheme which consumes a high percentage of the battery power, reduces the current lifetime of the IMD and enforces the patient to undertake more life-risking surgeries to replace the IMD; an unacceptable consequence.
   These two constraints therefore require the device to efficiently perform all activities in terms of energy management, and can not depend upon many components such as storage because this could violate the physical size requirement.

3. **Law and Regulation**: Before medical devices are introduced to the market, they need to go trough quality and compliance testing by various regulatory bodies. Approval of the FDA for example, takes on average $3-7$ years [5]. Any solution proposed 7 years ago, may have become obsolete over time due to technological evolution. Even if the solution has been approved, the devices will stay on average for 10 years implanted in the human body. Therefore there is a need for a long-term security solution, such that it will at least match the lifetime of the IMD. When possible, it is desired to propose a solution with the least required architectural changes to reduce the approval time of the FDA.

4. **Emergency Access**: Administered care for any treatment in a medical facility could be dangerous or ineffective when the medical personnel is unable to alter settings or deactivate the IMD. As an example, Russo *et al.* [40] identified the safety related concerns if an ICD is not disabled during an MRI scan. Suppose the patient is admitted to the emergency room of an unfamiliar hospital, the IMD needs to be accessible trough the wireless communication channel regardless of the familiarity of the external device. Therefore, to ensure access during emergency situations, the security scheme should enable an IMD to authenticate with an external device without the need of pre-established knowledge.

[5] `http://www.medscape.com/viewarticle/807243_2`, Retrieved June 2017

These criteria address the utility and safety considerations for an IMD. Compromising the safety of a patient should be avoided at all cost, but this will limit the possibilities to preserve the security criteria of the system hence introducing a tension. Without taking the utility and safety goals into account, the security solution may be in conflict with one of the challenges as described in this section. Therefore, protecting IMDs against attacks on the wireless communication channel requires a careful trade-off to find a balance between the security, safety and utility goals.

## 1.3. Problem statement and contributions

The wireless communication channel between an IMD and external device calls for a mechanism to protect the IMD against the identified theoretical and practical weaknesses without affecting the functional requirements of the device. In order to sustain the capability of treating patients for chronic diseases by means of an IMD enabled with wireless connectivity, the medical industry should address the identified tensions between safety, security and utility. The goal of this thesis is therefore to overcome the introduced security and privacy risks caused by the extended connectivity between an IMD and external device, without endangering the ability of an IMD to deliver its life-saving therapy adequately. The underlying research question of this thesis is as follows:

*How can an IMD establish a secure wireless communication channel with a legitimate external device, such that an attacker is prevented from gathering or changing privacy sensitive information, without diminishing the adequacy of the treatment delivered by an IMD.*

This research question is focused on actively exploring specific mechanisms for improving the device security and privacy while looking into the tensions between the safety and utility goals within this environment. Due to the complexity of this problem, the research question is addressed by taking the following steps:

1. Address the *challenges* for secure wireless communication for IMDs.
2. Understand *(lightweight) protocols* for authentication and key establishment.
3. Investigate and analyse *current solutions* and *sources of entropy* used to tackle the challenges.
4. Derive the specific *requirements* a solution should fulfil.
5. Design a protocol as a *solution* to the problem of the described domain.
6. *Evaluate* the solution by means of analysing experimental results and a security analysis.
7. Suggest possible *improvements* for the protocol and lay foundation for future research.

With the focus on establishing a secure wireless communication channel between an IMD and external programmer, the result of this thesis will be a lightweight authentication protocol that can operate within the resource constrained environment of the IMD. Ultimately, the three main contributions of the thesis are:

1. *Lightweight authentication and key establishment protocol:* Based on a strong adversarial model characterised by common threats on wireless communication protocols and requirements of regulatory bodies, proposed security scheme will have the following properties:
   - A security scheme without requiring pre-established knowledge between IMD and external device
   - Compliance with recent law and regulation
   - Without requiring hardware modifications
2. *Efficient source of entropy:* Many solutions use different sources of entropy for key establishment and authentication protocols. Providing a certain level of randomness however can be very costly. One of the contributions of this thesis is efficient use of existing functionalities as a source of entropy for authentication.

3. *Implementation and Evaluation*: The challenge to find a balance between safety, security and utility will be addressed by validating a complete implementation of the proposed lightweight authentication protocol. Validation is performed by testing the fulfilment of the requirements. Possible conflicts are addressed by comparing the proposed solution against solutions of other researchers.

## 1.4. Thesis outline

This chapter provides a brief overview of the research conducted in this thesis as well as a description of the domain the research is performed in. Furthermore, the challenges and motivation are specified before the problem statement is addressed. The remainder of the thesis is organised as follows:

- *Chapter 2, Background on threat models and cryptographic protocols.* To ensure mutual comprehension, common ground has to be established. This chapter will address background information to achieve this common ground. In the chapter, an identification of threats and basic understanding of protocols and key establishment will be provided.

- *Chapter 3, Prior art of IMD security schemes.* This chapter will introduce and compare related work within the described domain of authentication protocols for IMDs over a wireless communication protocol. The related work also indicates the already explored directions for this problem which can be used as a source for the proposed solution as well as the open challenges within this field.

- *Chapter 4, Research challenges and methodology.* Before designing an algorithm, the methodology followed in order to do so will be described. The chapter identifies the research challenges specific for the described setting and the methodology followed during the process of proposing a security scheme.

- *Chapter 5, Heartwear: Heartbeat security for IMDs.* A chapter dedicated to the design of a lightweight authentication protocol to improve the security of wireless communication enabled IMDs based upon the extraction of heartbeat signals. The design is divided into multiple phases, first an approach to process and extract features of heartbeat signals is proposed. Secondly the result will be integrated into one complete security protocol.

- *Chapter 6, Experimental Setup and Implementation.* To verify the performance of the *Heartwear* security, a proof of concept will be implemented in order to perform experiments in a realistic setting. First the experimental setup will be described, followed by the implementation of certain components of the protocol and is concluded with the experimental results.

- *Chapter 7, Evaluation of Heartwear* This chapter first addresses the correctness of the proposed synchronisation method. Thereafter, the proof of concept will be validated against the experiments run in the previous chapter along with a detailed security analysis. Lastly the protocol is compared against the characteristics of prior art.

- *Chapter 8, Conclusion.* The work will be concluded in this chapter by looking back at the obtained results in relation to the problem statement. Furthermore, this chapter will also lay a foundation for future research. The chapter will be concluded with summary of the solution and contributions.

# 2

# Background on threat models and cryptographic protocols

*In the previous chapter the importance of secure wireless communication for IMDs has been introduced followed by an outline of the challenges within the described domain. For the design of a lightweight security scheme to establish a secure communication channel, a deep understanding of possible attacks, authentication and key establishment is needed. These topics will be addressed in this chapter by first introducing threat models in Section 2.1 where also common attacks, adversary models and laws and regulations are addressed. Thereafter the concept of authentication and key establishment protocols are discussed as well as the importance of sources of entropy. The chapter is finalised by the notion of profiles used to gain insight of the specific needs for a lightweight authentication protocol in a certain domain.*

## 2.1. Understanding threat models

A threat model is used to identify the security problems of the system that is being analysed. By identifying these security problems, policies and security requirements can be deduced. In general threat modelling approaches are centred on models of assets, models of attackers or models of software. Within the domain of IMD security the threat model approach of models of attackers is adopted as the weaknesses in the systems have been addressed from an attacker oriented point of view. In general the approach requires a clear view what kind of adversaries the IMD should be protected against. Depending on the adversary model, there exist a set of attack vectors which usually evolve from existing vulnerabilities and threats in the system:

Threats to any system can be described as a set of events that have the potential to cause loss or harm. In the case of IMDs, new threats arise because of the added functionality of wireless communication. While wireless communication allows efficient transmission of telemetry, it is also more easily intercepted or modified. It is good practice to identify the threats of the system [45], as it will lay the foundation of what attacks can be exploited in order to harm the security properties of the system. The adversary model is taken into account to determine what attacks should be mitigated and which are out of scope for the given scenario.

Threats are distinguished from vulnerabilities in the sense that a threat is a set of circumstances that have the potential to cause loss or harm whereas a vulnerability describes a weakness in the system. Exploiting a vulnerability to cause loss or harm is then referred to as an attack. Thus a combination of threats may result in a vulnerability which can support the successful execution of an attack. Mitigation techniques serve the purpose to protect the system from threats, which remove or reduce a vulnerability. For the remainder of this thesis, we will focus on the possible threats to the IMD communication protocol.

In order to design a security scheme to protect an IMD and external device from harmful events, an

attacker oriented approach for the threat model will be applied. This requires a deep understanding of different adversary models. In order to derive a suitable adversary model, the the common attacks on the wireless communication channel are first investigated after which a comprehensive description of different adversary models will be described.

### 2.1.1. Common attacks on wireless communication protocols

A wireless communication channel is a generalisation of all communication methods which do not require the participant to be physically connected to another entity to communicate. More specifically, this section is dedicated to the wireless communication protocols that establish a connection over a pre-defined radio frequency range. The well known Wi-Fi protocol for example, operates on the 2.4 and 5.0 GHz radio band, Bluetooth uses short wavelengths from 2.4 – 2.485 GHz and Near-Field-Communication (NFC) technology uses the much lower frequency of 13.56 MHz. Some protocols operate within much smaller distances (NFC for example) and others on large distances (Wi-Fi) however, typically they share the same principle of transmitting data over a wireless connection. While wired networks communicate over physically connected medium, a wireless network establishes a communication channel over air. As described by Zou *et al.* [52], the wireless functionality makes the communication protocol more vulnerable for attacks such as eavesdropping, Denial of Service (DoS), spoofing, Man in the Middle (MitM) attacks or modification of the messages while in transit.

The same survey [52] investigated the common attacks for wireless communication protocols and categorised them according to the previously explained CIA triad. Boyd [5] identified types of protocol attacks which appear to be a more generic description of attacks compared to the ones described in [52]. There is no universally accepted taxonomy of attacks on wireless communication channels available, however a comprehensive overview for the domain of wireless communication for IMDs is given below by considering the classifications of both studies:

#### Eavesdropping

Eavesdropping is a technique performing a passive attack since it does not require the adversary to disturb the communication channel directly. As long as an adversary is within transmitting boundaries, it can sniff and record all data transferred on the channel. This type of attack is often one of the first steps before an active attack is performed. To avoid leaking sensitive information, wireless communication protocols must rely on cryptographic techniques.

#### Modification

In general, all (partial) transmitted messages are vulnerable to modification attacks. Modifications could occur whenever any of the protocol message fields are altered or removed. *Spoofing* for example, may change the origin of the sender to fool the receiver into establishing a connection or privilege escalation. Other modification attacks include the splitting or re-assembling of one message. All types of modification attacks violate the integrity property of the protocol. Maintaining the integrity of the communication channel allows identification of modification of all parts of the message that must be kept together.

#### Replay

This type of attack is often engaged with other attack elements, and may lead to the leakage of privacy sensitive information or unauthorised control of a device. A specific case of a *Man-in-the-middle* (MitM) attack for example, records information seen in the protocol and sends it to the same destination during the same protocol run. This causes all traffic to reroute through the adversary instead of directly between sender and receiver. An adversary may also record information and send it to a different destination, possibly during a later protocol run.
Another special case of replay attacks is the *reflection attack*. In this scenario the adversary intercepts a message and reflects it back to the sender to establish a communication session or retrieve sensitive information based on the response. The attack can for example be feasible if multiple authentication attempts can exist in parallel of other authentication attempts.
Replay attacks can harm the authentication property of the communication protocol whenever there is no control of re-occurring sessions. This can be obtained by including the freshness property, which must guarantee a certain value has not been used before. These attacks may also compromise the

Table 2.1: An overview of the common attacks and affected property based on the CIA triad.

| | Confidentiality | Integrity | Availability |
|---|---|---|---|
| **Eavesdropping** | x | | |
| **Modification** | | x | |
| **Replay** | x | x | x |
| **DoS** | | | x |
| **Cryptanalysis** | x | | |

availability of the system, by replaying computationally heavy messages to deplete the available resources of a system.

#### Denial of Service

In a Denial of Service (DoS) attack, an adversary attempts to compromise the availability of a system. The method used to perform such an attack may involve *resource depletion*, *jamming* or *connection depletion*. In case of resource depletion, a device is instructed to perform a resource consuming task which is for example computationally heavy or requires a lot of energy such that the device becomes unresponsive. Jamming can also affect the availability of a device. By generating interference on the communication channel, legitimate users are prevented to communicate on that channel. Connection depletion can be achieved by *injecting* messages on the communication channel. This may overload the device with incoming communications and render it unavailable for legitimate communications.

#### Cryptanalysis

A cryptanalysis attack describes the investigation of an attacker to find secret details of the cryptographic methods included in the protocol. As an example, whenever the attacker can eavesdrop all communication between two devices, he may be able to find critical information about the secret key used to encrypt the channel. A *ciphertext only attack* is a form of cryptanalysis where an attacker can only try to gain information based on a set of ciphertexts. For a *known-plaintext attack*, an attacker has access to both the plaintext and its encrypted version. The easiest for an attacker is the *chosen plaintext attack* where the adversary may choose any input and receives the corresponding ciphertext.

These attacks can be performed on both the short and long-range distances of a wireless channel. Regarding long range attacks, the adversary can operate from a larger distance to eavesdrop on the communication but may also require more sophisticated equipment compared to short range attacks. Regardless of the range of the attack, an adversary is able combine multiple of the listed attacks in order to compromise the availability, integrity or confidentiality of the system. An overview of the attacks in relation with the impact and mitigation techniques is given in Table 2.1. Due to the variety of replay attacks, it could harm the confidentiality, integrity and availability property of the device. Cryptanalysis and eavesdropping only compromise the confidentiality of the system while DoS affect the availability property of the device.

### 2.1.2. Adversary models

In general, adversaries can be classified into two different categories. Depending on the capabilities, an adversary is able to execute a certain set of attacks. The adversarial resources may differ in quality, approach or knowledge, and also depend on the goal the adversary has set. One category contains all *active* attacks and the other only concerns *passive* attack vectors, where the class of adversary describes the activities an adversary is able to perform:

- **Passive Adversary:** Any malicious operation that does not involve generation or modification of traffic is contained in the capability set of a passive adversary. This may limit the adversary to a smaller set of possible attacks with the focus on eavesdropping. Despite the limitations however, an adversary will likely remain unnoticed as passive attacks do not interfere with the normal functioning of the system. A passive adversary can for example be able to eavesdrop on the wireless communication and record it for later use. Passive attacks may therefore be in favour whenever the attacker wishes to stay unnoticed.

- **Active Adversary:** An active adversary is able to execute a broad range of possible attack vectors. This type of attacker can gain full control over the communication channel by actively interfering with legitimate connections. Additionally, this adversary is also capable of initiating a communication channel by spoofing its identity. Depending on the capabilities of the attacker, it is also not limited to the approaches to execute an attack. This includes the use of specific equipment or physical interference with the target.

Both adversary models have their advantages from an attacker perspective: The passive adversary may be limited in capabilities but remains easily unnoticed during the execution of an attack. An active adversary however, has a wide varity of approaches to invade the wireless communication channel in order to reach its goal.

One of the strongest assumptions in adversary models is a combination of the active and passive adversary. An example of this has been identified as a *security assumption* by Boyd [5], where the adversary has complete control of the channel:

**Security Assumption.** *The adversary is able to alter all messages sent in a cryptographic protocol using any information available. In addition the adversary can re-route any message to any other entity. This includes the ability to generate and insert completely new messages.*

This means it is assumed that the adversary can control all communications between two entities by *observing, altering, inserting, delay* or even *deleting* all messages sent over the channel. Note that an adversary may not always be the malicious external party, but could also be a legitimate protocol participant (an insider) or a combination of both. Insiders with malicious intent are extremely hard to counter completely, because such entities often have information about the system and could know on beforehand how to circumvent the security measures taken. Depending on the adversary model, such scenarios should be taken into account for the design of the protocol or not.

### 2.1.3. Law and regulation

Another important section for IMD security specifically is to take into account certain laws and regulations. Such law and regulation can constrain the possible mitigation techniques or prioritise the security of certain properties.

Because of the raised concern in the research community as explained in Section 1.1, regulatory bodies also started recognising the importance of security in IMDs. The US Food and Drug Administration (FDA) states the increase in connectivity of medical devices also increases the risk of cyber security threats [13]. In the same statement, the FDA recommends mitigating and managing cyber security threats appropriately. While these recommendations are not binding, it is a clear advice to implement appropriate security measures. Additionally, it is not uncommon to incorporate the official advice into a regulation in the future.

The European Commission has matured even further in the acknowledgement to improve the security of devices that process privacy sensitive information such as IMDs. It recently agreed upon the reform of the data protection rules that entered into force on the 24th of May in 2016 for all Member States of the European Union (EU), called the General Data Protection Regulation (GDPR) [49]. The Directive will apply the 25th of May in 2018 and also affects the processing of data concerning health such as genetic or biometric data. Article 24-25 for example, describes the requirement for data processors, such as IMDs, to have appropriate technical and organisational measures implemented to maintain the confidentiality and integrity of the data. Devices that do not comply with the GDPR after the 25th of May in 2018, may result in notable fines for the device manufacturers.

Although current law and regulation has not focused on the security of medical devices, forthcoming law and regulation like the GDPR will. This also implies that the security measures taken for these devices must comply with future law before it will be accepted to the market. It is therefore also important to consider the approaching changes when designing a secure communication protocol.

## 2.2. Protocols for authentication and key establishment

Before looking at proposed solutions for secure wireless communication for IMDs from the research community, an introduction to protocols for authentication and key establishment will be provided. Protocols for key establishment and authentication need to satisfy predefined properties, in order to mitigate certain attack vectors. Whenever an attack is possible that should have been covered, the affected property is violated. Therefore, depending on the adversary model and security assumptions, a set of properties need to be deduced to mitigate possible attack vectors. Before such definitions can be defined, mutual comprehension needs to be established on protocols for authentication and key establishment in general.

### 2.2.1. Introduction to key establishment and authentication

Key establishment and authentication are often seen together, but may nonetheless exist apart from each other. Depending on the design goal, it might not be necessary to establish both. For example, Gollmann [16] has investigated the purpose of authentication and key establishment more closely which resulted in the definition of two goals:

**Goal.** *The protocol shall establish a fresh session key, known only to the participants in the session and possibly some trusted third parties.*

With this goal, confidentiality during the communication between two entities needs to be maintained without necessarily knowing each others identity. The goal also mentions *key freshness*, an important principle to ensure a new key is being used for every session. This property is desirable because any previously used key may have been exposed to other unknown entities, which causes the loose of control in who could have access to the keys. This goal is therefore considered a *Key establishment* goal, rather than authentication.

The second goal, as described by Gollman, is as follows:

**Goal.** *A cryptographic key associated with an entity B was used in a message received by entity A during the protocol run. The protocol run is defined by A's challenge or current timestamp.*

Although this goal also mentions a cryptographic key, there is however no explanation about its freshness or properties. Therefore, this goal is only related to entity authentication, because entity $B$ confirms his identity to entity $A$ through the acquisition of corroborative evidence. While there may be cases where entity authentication by itself may be useful, it is rarely used in practice. Diffie *et al.* state that 'it is accepted that these topics should be considered jointly rather than separately' [12]. Within the research community, there is clearly not an accepted definition about the joint relationship between key establishment and authentication. To clarify the different phases of a protocol, the remainder of the thesis considers key establishment and authentication separately.

### 2.2.2. Authentication protocols

An authentication protocol has been described as '*the process of establishing confidence in individual identities*' by NIST [19]. It is a process where an entity needs to gain trust in another by means of executing a protocol. One of the recommended protocols by NIST is the *2-step unilateral* authentication protocols [2]. The description in the NIST report can be translated to a very simple protocol where entity $Bob$ wants to authenticate himself to entity $Alice$:

| **Alice** | | **Bob** |
|---|---|---|
| Nonce: $N_A$ | | |
| | $N_A \longrightarrow$ | |
| | | $N_A$ |
| | | Sign nonce: $Sig_B(N_A)$ |
| | $\longleftarrow Sig_B(N_A)$ | |
| Validate and accept or reject | | |

In the two-step unilateral authentication protocol, $Alice$ sends a *nonce* $N_A$, which is a random and time-varying value generated by $Alice$. $N_A$ needs to be returned to $Alice$ by $Bob$ such that $Alice$ is convinced of $Bob$ his identity. $Bob$ achieves this by signing the nonce and return it to $Alice$, expressed as: $Sig_B(N_A)$, where $Sig$ is a cryptographic function. This method for authentication is called *challenge-response* where one party (the verifier) presents a question and another party (the prover) must provide a valid answer. The approach is not uncommon and works well, one example is the case of authentication with a password where one party asks if the other has knowledge of the password. In this example however, the protocol is vulnerable to the following attack:

An adversary $Charlie$ can capture the first challenge, replay it to $Bob$ and replay the response of $Bob$ (which is the signature) to $Alice$. This enables $Charlie$ to be authenticated as $Bob$, and if $Bob$ is considered a trusted party, now so will $Charlie$. This example shows that authentication is desired to have at least the following two properties:

1. $A$ should be convinced that $B$ has recently replied to a specific *fresh* message.

2. $B$ should be convinced that $A$ initiated the communication to $B$ directly.

By combining both properties in one protocol, *strong authentication* will be achieved. This assurance however only authenticates $B$ to $A$, whereas there are many examples that also demand authentication from $A$ to $B$. This requirement is called *Mutual authentication* and can be realised in its simplest form by running a single strong authentication protocol twice, but with reverse entities. More efficient protocols combine certain steps. A well known example that provides *strong authentication* and *mutual authentication* is the Transport Layer Security (TLS) protocol, which runs on top of the TCP layer in computer networks [11].

### 2.2.3. Key establishment protocols

As described in Section 2.2.1, key establishment is different from authentication and has other requirements. Boyd [5] has described two approaches for key establishment:

1. **Key transport:** One entity generates and distributes a (symmetric) secret key

2. **Key agreement:** All entities involved, jointly generate a secret (asymmetric) key by combining public and private keys.

Both approaches have their strengths and weaknesses. In case of the key transport approach, a key is decided by a single entity without the need of collaboration. Distribution of the key however, involves another security scheme in order to share the key in secret with other entities. For the key agreement approach, deciding upon a key is based on heavier computations such that multiple entities can compute the key locally without the need of sharing secret information. This makes the key agreement approach more resource consuming, but it overcomes the problem of distributing a secret without the need for another security scheme. Since the key transport approach only requires a single key distributed among all involved entities, it supports symmetric key cryptography. If the challenge of distributing a symmetric key is overcome without additional cost in resource consumption, the key transport supports a more efficient method to provide confidentiality compared to the key agreement approach.

Both approaches allow the establishment of a key which serves the purpose of maintaining the confidentiality and/or authentication in the protocol. As a consequence, a *good key* should thus only be known by trusted entities. Therefore, regardless of the approach, an important part for ensuring the property who has knowledge of the key also requires *key freshness*. This leads to the following properties of a good key:

1. **Freshness:** The key should be guaranteed to be new for every protocol run, to avoid possible re-use of older keys by an adversary.

2. **Confidentiality:** The key should be only known to the designated entities, and unknown to an adversary.

Whether symmetric or asymmetric key cryptography is involved in the security scheme, , these properties of a key should always be maintained when it is considered as a good key. For lightweight cryptography purposes, symmetric cryptographic primitives are favoured as it does not rely on the computationally heavier modulo operations like asymmetric key cryptography does.

### 2.2.4. Random number generation

For both key-establishment as authentication a function generating a random bit sequence is desired. The stronger the sequences numbers are, the less predictive the sequences (or numbers) used in any cryptographic protocol are for an adversary. Generating a random bit sequence however is another difficult challenge since a random bit sequence must be unpredictable. The requirements of a random number generation is illustrated by the following example:

Suppose the result of the flips of an unbiased "fair" coin with heads is represented as a *0* and tails as a *1*. The probability of producing a 1 is for each flip $\frac{1}{2}$, and does not affect future coin flips. This makes the sequence generated by the unbiased coin *independently and identically distributed* (i.i.d.), or random. Such a property will provide *forward unpredictability* since it is unfeasible to determine the next random numbers in spite of any knowledge of previous random numbers in the sequence. Any good random source however still needs to overcome the production of non-random numbers such as the occurrence of long sequences of zeroes or ones. One approach in validating the unpredictability of a generated sequence is by applying the NIST statistical test suite [39].

Generating random numbers can be achieved by using a non-deterministic source, or in other words an *entropy source*. This source along with a good *entropy distillation process* will avoid weaknesses in the produced sequences to maintain its unpredictability. Sources of entropy may be any process that can produce the desired randomness: the noise in an electrical circuit such as Physical Unclonable Functions (PUFs), timing of certain processes, the speckle effect of light waves or a combination of such random processes.

Creating a truly random bit sequence may be very time-consuming which can be undesirable when large quantities of random numbers are needed. Pseudo-random number generators can provide bit sequences that may appear to be more random than truly random sequences faster, but require additional hardware components and should be implemented without any flaws to avoid the loss of entropy and maintain the unpredictability. Another approach in validating the unpredictability of a generated sequence specifically for pseudo-random number generators is by checking if the output sequence satisfies Golomb's postulates [18]. Depending on the application, one might favour pseudo-random or random above the other.

## 2.3. NIST standardisation

The Information Technology Laboratory (ITL) of National Institute of Standards and Technology (NIST), proposes and sets standards in information technology, mathematics and statistics. This non-regulatory organisation has also developed cryptographic standards and guidelines for securing systems effectively with cryptographic methods.
In 2016, NIST published a draft report on lightweight cryptography [34] where it is acknowledged that the current NIST-approved algorithms may not be suitable for highly constrained devices that are interconnected. As it is not expected that one algorithm can meet all characteristics, it is suggested to develop profiles in order to identify which algorithm may be suitable for a certain application. For the creation of such profile, first the characteristics need to be identified, followed by the design goals in order to find a suitable cryptographic solution.

### 2.3.1. Characteristics

In [34], devices requiring lightweight cryptographic solutions are ones that are limited in resources, performance or energy. The constrained environment of a device is characterised by at least one of the following:

- **Area:** The physical available space, which can measured in gate equivalents (GE) or physical size. For any suitable cryptographic method, there should be sufficient NAND gates available

for computation.

- **Memory:** Whenever a cryptographic method requires a large amount of storage to store fixed data (such as S-boxes, hard-coded round keys) the ROM should have the space available. The ROM capacity should be sufficient if the method requires the storage of many intermediate values.
- **Performance:** Limitations in performance can be calculated based on the latency and throughput measured in amount of clock cycles. The clock-cycles need to be minimised in order to keep the latency low, whereas throughput is measured in the amount of cycles per byte.
- **Energy and Power:** Where *energy* resembles the number of cryptographic operations and is measured in Joule, *power* describes the amount of operations per second ($J/s$). By calculating the amount of ciphertext processed (in bytes) per Joule, and the expected bytes processed per day can predict the expected lifetime of a battery. The power and energy consumption need to be minimised in order to conserve the battery life-time as long as possible.

These characteristics support the creation of a profile in order to determine the possible bottlenecks for cryptographic solutions. After the development of a profile, a suitable cryptographic solution may be proposed according to the design goals that have been set.

### 2.3.2. Design goals and profiles

In order to create a profile as described by NIST [34], design goal(s) also need to be defined. The draft report describes a set of exemplary design goals which are: security strength, flexibility, low overhead for multiple functions, ciphertext expansion, side channel and fault attack mitigation, limits on the number of plaintext, related-key attacks. For example, one design goal could be that *authenticated encryption* is required for the design. After determining the characteristics and design goals, the preferred type of primitive is deduced to complete the profile. The proposed template is as given in figure 2.2:

Table 2.2: The profile template as proposed by NIST [34]

| **Profile** | *<Profile name>* |
|---|---|
| **Primitive** | *Type of primitive* |
| **Physical Characteristics** | *Name physical characteristics and provide acceptable ranges* |
| | *(e.g. 64 to 128 bytes of RAM* |
| **Performance Characteristics** | *Name performance characteristics and provide acceptable ranges* |
| | *(e.g. latency of no more than 5 ms)* |
| **Security Characteristics** | *Minimum security strength, relevant attack models,* |
| | *side channel resistance requirements etc.* |
| **Design Goals** | List design goals |

The profile template is however currently still under discussion. A more detailed profile has been given in the draft report published by NIST in April 2017 [4]. While these two profiles are a complete example, NIST opened a public comment period to determine how suitable the draft profiles and profile template. As a result, the public comments received as reported in [35] indicate it was expected to create profiles with a narrower focus and more finely-tuned constraints to result in more efficient solutions. This progress indicates NIST is also putting their efforts in standardising lightweight cryptography, a process proven not to be trivial.

# 3

# Prior art of IMD security schemes

*Maintaining the confidentiality, integrity and availability of an IMD whilst enabling wireless communication with an external programmer is a challenging task. Solutions in this direction have been proposed by numerous researchers, but vary in the adversary model and attack strategies that have been considered, resulting in different approaches. This chapter gives an overview of existing solutions aimed at securing the wireless communication channel between IMD and external programmer, describes the underlying techniques based upon the taxonomy and evaluates the trade-off between safety, security and utility of each category in this taxonomy.*

## 3.1. Taxonomy of authentication protocols in IMDs

The security concerns posed by allowing wireless access to an IMD have led to the development of various security schemes within the research community to protect the wireless communication channel. Each of the proposals however take security, safety and utility considerations into account differently. As a consequence comparing the trade-off performed to design such solutions directly may be misleading. In order to compare them appropriately, the proposals have been classified into four different categories based upon their authentication characteristics. The different approaches are depicted in Figure 3.1.

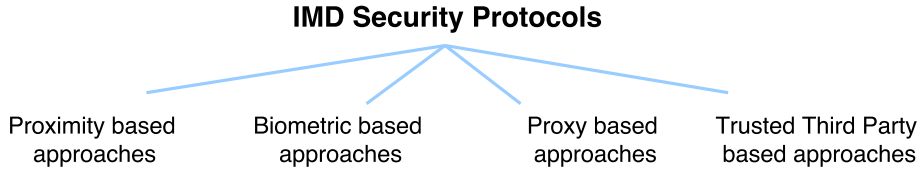



Figure 3.1: Taxonomy of approaches for authentication protocols in IMDs

Each approach will be discussed separately, and is accompanied by one or more corresponding protocols who all share the common goal of proposing a security scheme for the wireless communication channel between IMDs and external programmers.

### 3.1.1. Proxy based approaches

Proxy based approaches are characterised by the introduction of an additional external device. This device handles the communication between unknown external devices and the IMD itself. Some proxy devices maintain a connection with the IMD by means of a pre-established secret and overcomes the cryptographic overhead of key establishment and authentication protocols. Other proxy devices keep a long-term connection with the IMD to reduce the amount of times an authentication and key establishment protocol needs to be executed. The proxy device performs the computationally heavier cryptographic operations to establish a session with the unknown external programmer. The increase of power consumption on the side of the proxy device is less of a concern, as one can replace the battery more conveniently compared to an IMD which requires surgery. In general, a proxy device

therefore takes away the environmental constraints the security scheme should adhere to by adding a trusted external device between the IMD and external programmer.

One proxy based approach has been proposed by Denning *et al.* [8] by completely relaying all messages between an IMD and external programmer through the **Cloaker**. The Cloaker is coupled with the IMD by means of a pre-established secret to support symmetric key algorithms, whereas the Cloaker and unfamiliar external device communicate by means of heavier public key algorithms. In case of emergencies, the solution requires medical personnel to remove the Cloaker. This triggers the IMD into fail-open access to all external programmers. Thus in case the IMD observes the absence of the Cloaker, it will allow any incoming connection from any device in order to support emergency access.

A similar technique proposed by Xu *et al.* [50] is the **IMDGuard**, but differs in the fact that the proxy-device authenticates with the IMD by establishing a secret without prior knowledge. This technique is implemented where the IMD and IMDGuard extract the same secret from the measured heartbeats. In case of emergencies, the IMDGuard is removed allowing an external programmer to connect directly with the IMD by following a weak challenge-response scheme: the IMD will send a nonce $n_1$, and after waiting $t$ time another nonce $n_2$ is sent, the external device is authenticated if it correctly responds with $n_1 \oplus n_2$. While this emergency protocol requires the external device to be within reach for at least $t$ time, the challenge response protocol does not provide any guarantee to the IMD about the identity of an external device besides that it is able to wait for $t$ time.

**The Shield** proposed by Gollakota *et al.* is a security solution that introduces authentication without the requirement to any change anything to current IMDs. The introduces proxy device is a full duplex radio, allowing to transmit and receive simultaneously. One antenna jams all signals in its surrounding on the radio frequency the implant operates on. The other antenna is able to receive and transmit to create an antidote signal. This enables the shield to prevent any device other than itself from directly communicating with the IMD since only the Shield can decode the IMDs scrambled signal. Emergency access in this case is possible by removing the shield to allow an external device to communicate with the IMD.

### 3.1.2. Biometric based approaches

Biometric based approaches use human characteristics for authentication. The body has many elements that are unique for every individual such as the iris, fingerprint, scent and heart rhythm. These unique features are in general called physiological values and can be used for authentication or key establishment. The characteristics are especially interesting if it concerns a physiological value that is already being measured by the sensor of the IMD to monitor the vitals of the patient. If these measurements are considered to be independent and identically distributed random variables they can provide a good source of entropy for random number generation. Whenever such variables are also measurable externally it can serve as a generator for key establishment protocols.

Hei *et al.* provide a **two level access control** scheme for authentication [22] by introducing two subsequently executed access control schemes. The first authentication level requires an external device to provide basic biometric information like eye colour, height or fingerprints to the IMD which has this information also stored in memory as prior knowledge. These biometric traits have been converted into a specific digital format such that the comparison requires minimal resources of the IMD to mitigate most battery depletion attacks. After passing the first level of authentication, a detailed measurement of the patient's iris needs to be passed to the IMD. Completely matching the full iris scan is however still too heavy for the IMD, therefore only a partial match is required. The complete security scheme relies on pre-established knowledge that can not be refreshed troughout the lifetime of the IMD.

Rostami *et al.* apply heartbeat measurements as physiological value in **H2H authentication** [38]. First a session similar to TLS is set up to provide confidentiality and integrity by agreeing on a key, then the authentication phase takes place where the identity is proofed by means of a commitment scheme based upon heartbeat measurements. The measured value is encrypted with the key established in the TLS session and then committed to a value $w$, creating commitment $C$. Both devices exchange

the commitment after which both devices exchange $w$ such such that the IMD and external device can de-commit the measurements. The authentication succeeds if the exchanged heartbeat measurements lie within a predefined threshold. For emergency access any external device that is able to measure the heartbeats within the defined threshold and knows the protocol can connect, making this solution independent from prior-knowledge required for the external programmer.

Seepers *et al.* propose a fuzzy commitment scheme based upon heartbeat measurements in "**Secure Key-Exchange** for implants using heartbeats" [42]. **Secure Key-Exchange** also takes into account the measurement error of two simultaneous heartbeat readings by using the values as witness $w$ in a fuzzy commitment scheme. For conventional commitment schemes, the unique witness $w$ used to create a commitment must match exactly with the witness $w$ used to decommit the value as $w$ essentially functions as a decryption key. Fuzzy commitment schemes [26] however, accepts a $w$ that is close to the original $w$ but not necessarily identical. This allows the external programmer to use the externally measured heartbeats as $w'$ to decommit the commitment received from the IMD which succeeds if $w \approx w'$ rather than $w = w'$ for convential commitment schemes.

### 3.1.3. Proximity based approaches

This approach is based on the assumption that whenever an adversary is very close to the patient, there are alternative and easier methods to harm the health of a patient. Emergency mode is more easily implemented in this approach because all authentication requests can be directly accepted whenever a device is physically present within the required distance. This approach therefore allows any incoming request as long as the external device communicates by means of the same protocol and is within the pre-defined distance of the IMD. Most of the proximity based approaches however, do not provide confidentiality or integrity as for which an extension to the current authentication scheme is required. The following security scheme however contains both authentication as well as key establishment based upon a proximity based approach.

Halperin *et al.* applied this approach without requiring additional power to authenticate an external programmer with the IMD in **Zero Power Security** [21] and informs the patient whenever a device attempts to authenticate. To increase awareness by the patient, a piezo-element is added to the IMD which can audibly warn the patient after receiving an authentication request. A Wireless Identification and Sensing Platform (WISP) [46] is added to the solution to harvest energy from RF signals generated by the external programmer. This provides the security scheme with enough power to perform the key exchange. The key is exchanged over a modulated sound wave which is only sensible with a microphone that is in contact with the patient's body. Therefore, the solution does not consume additional power but will require additional hardware to be implanted underneath the skin and needs to be attached to the IMD.

### 3.1.4. Trusted third party based approaches

Another approach is the introduction of a trusted third party (TTP) in order to establish a secure communication protocol between two devices. A TTP is a third entity, trusted by the two devices and provides a mechanism such that both devices can communicate with each other. An example is the Certificate Authority (CA) functioning as TTP to distribute a public key which enables the set-up of a disposable symmetric key and is commonly used to connect to a website over the (resource consuming) HTTPS protocol. The technique can also be tailored to a security scheme for IMDs as proposed by the following researchers.

Marin *et al.* introduced a **semi-offline** protocol [32] where the device manufacturer has been appointed as the TTP. A master secret key (MSK) is generated upon initialisation of the IMD which is securely stored and never shared. All IMDs receive a diversified key $H_2(id)^{msk}$, where $id$ is the identity of the IMD. External programmers receive a temporal key $H_1(t)^{msk}$, which allows the derivation of the keys generated by the IMD for the given time period $t$. When $t$ has passed, the external programmer needs to update its key at the device manufacturer because the key has become obsolete. The remainder of the protocol runs symmetric key cryptography using the agreed key $e(H_1(t), H_2(id))^{msk}$. Outdated external devices (i.e. devices who do not have an updated $t$), can not communicate with the IMD anymore. Since this renders the programmers useless, they are also not qualified to access an

IMD in case of emergency.

To minimise the memory consumption of key storage, and reduce the overhead of computing the symmetric key, Xu *et al.* propose a hardware oriented solution for secure communication between an IMD and external device called "**Matched Digital PUFs** for Low Power Security" [51]. A physical unclonable function (PUF) utilises the unique intrinsic physical behaviour of integrated circuits (IC) to create a mapping which is easy to evaluate but impossible to predict. This solution applies PUF technology in an external programmer where the power-up of SRAM cells are exploited as random seed for the first cycle of a PRNG. This random seed is also stored in the IMDs non-volatile memory during the enrolment phase before implantation to ensure both devices generate the same random number.

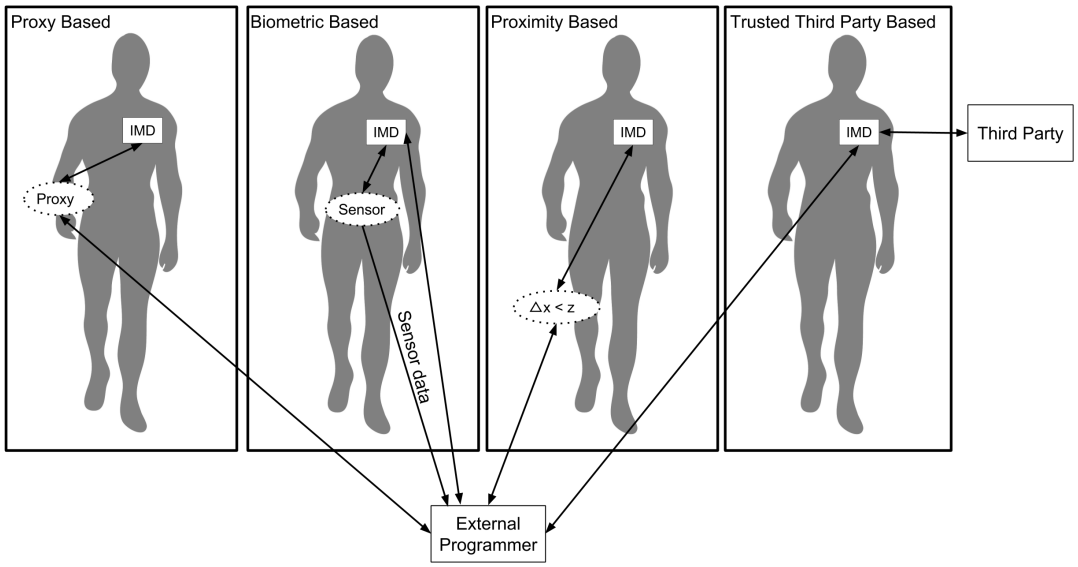



Figure 3.2: graphical overview of the proxy, biometric, proximity and trusted third party based approaches.

## 3.2. Identification of tensions in existing solutions

The described existing solutions all aim at securing the wireless communication channel between an IMD and external programmer by means of a proxy, biometric, proximity, or trusted third party approach, as depicted in Figure 3.2 and are specifically focused on establishing trust between two devices. The use of a security scheme however, can create tension between the safety, security and utility requirements of the IMD. These tensions have been identified by Halperin *et al.* [21] as the tension between security vs. resource consumption, security vs. accessibility and security vs. usability. To identify the complexity of satisfying all requirements, the described solutions have been analysed against these tensions.

While all studies attempt to minimise the resource consumption during the authentication process, only **Matched Digital PUFs** and **IMDGuard** also reduce the resource consumption for the process of key generation. Both provide a method to extract a key from a source of entropy accessible by the IMD, without the additional need of conventional and expensive PRNG's.
Regarding the accessibility of the solutions, all solutions except for **Semi offline** and **Matched Digital PUF's** do not require the establishment of shared secrets prior to implantation. Solutions without this requirement increase the accessibility for unknown external programmers, and thus support the emergency access requirement. All proxy based approaches however, (partially) remove the authentication process in case of emergencies which make the IMD more vulnerable to attacks.
In terms of usability, **Zero Power Security** requires major changes in the hardware architecture of the IMD due to the addition of several components which forces the design to go through a complex and long process for quality and compliance testing. Additionally, all proxy-based approaches require a patient to carry an additional external device to provide security. This introduces inconvenience for

the patient, lowering the level of usability for this solution.
An overview of these observations are provided in Table 3.1, where the biometric based approaches seem to balance the tension between security, accessibility, usability. If the physiological value used in the security scheme is also already being measured by the IMD for monitoring purposes, the biometric based approach additionally preserves the resource consumption for the generation of random numbers. The discussed security schemes however, do not completely rely on the physiological value and still require an additional random number generator. Additionally, **2-Level access control**, authenticates an external programmer by verifying its knowledge of a static secret value. This reduces the secrecy of the value when more external devices connect, as the designers of the scheme can not control the secure storage of the static value in external programmers.

Table 3.1: An overview of the analysis to identify tensions amongst related work

| Approach | | Accessibility | Resources | Usability |
|---|---|---|---|---|
| | Cloaker | x | | |
| **Proxy** | IMDGuard | x | x | |
| | The Shield | x | | |
| | 2-level | x | | |
| **Biometric** | H2H | x | | x |
| | Secure Key | x | | x |
| **Proximity** | Zero power | x | | |
| **TTP** | Semi-offline | | | |
| | Digital PUF | | x | |

Following this comparison, and as shown in table 3.1, it appears the **Matched Digital PUFs** and **IMDGuard** provide an addition in reducing the resource consumption without compromising the security goals. On the contrary, **H2H authentication** and **Secure Key exchange** maintain a balance between the security, accessibility and usability goals. This overview demonstrates that none of the described security schemes provide a resource efficient source of entropy while allowing secure access in emergency situations without the need of hardware specific changes. **IMDGuard** however, takes advantage of heartbeat based measurements to authenticate the proxy device with the IMD, while **H2H authentication** and **Secure Key exchange** additionally maintain the usability of the device. Therefore, the heartbeat based security solutions are analysed more thoroughly in the following section.

## 3.3. Comparison of heartbeat based security solutions

Protocols considering heartbeat measurements for IMD security have been previously proposed for either key establishment or authentication [38] [42] [50]. While these studies rely on the same principle, the challenge to secure the wireless communication channel of an IMD is approached differently. Therefore, this section is dedicated to comparing the state of the art by first addressing each solution separately to identify the objectives, adversary model and protocol specification. The section is concluded with a comparison of the solutions to identify open problems and challenges within this field.

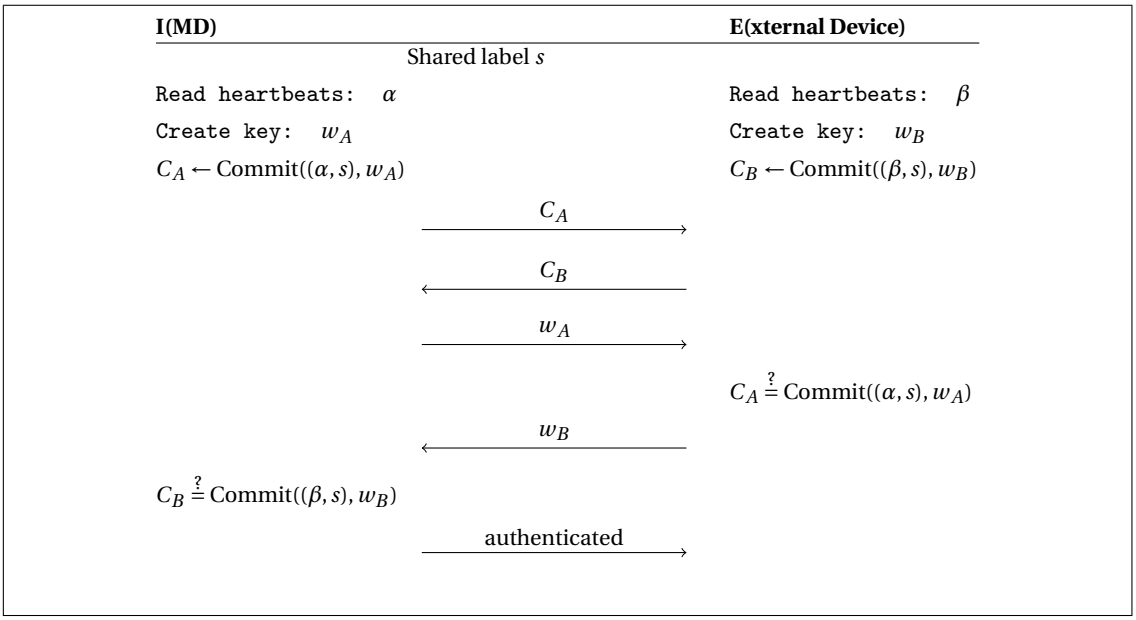
### 3.3.1. Analysis of H2H Authentication

H2H authentication as proposed by Rostami *et al.* [38], introduced a pairing protocol to protect against active adversaries. Heartbeat measurements are included in the solution in order to authenticate the two unknown devices with each other based upon the assumption that an external device can be trusted if and only if it has significant physical contact with the patient's body. The assurance an external device has physical contact with the patients body, is given by executing a commitment scheme which relies on heartbeat measurements.

The adversarial model of H2H authentication is created such that it is believed the attacker is present during the process where the IMD and external device attempt to authenticate. Additionally, it is assumed the attacker has complete control over the communication channel. It is stressed that, the presence of the adversary is also expected during an emergency situation. In case of an emergency situation, the authentication mechanism should still be executed as it does not depend on device-

specific keys or pre-established knowledge. Similarly, it is considered impractical for medical personnel to contact an authority for access credentials, implying public-key infrastructures are unsuitable.

The pairing protocol of H2H authentication is represented in Protocol 1,where the IMD and external device attempt to authenticate with each other. First, a secure channel has been set up trough




Protocol 1: *H2H authentication*, a security scheme to establish a secure communication channel between the IMD and external device proposed by Rostami *et al.* [38]

TLS to provide confidentiality and integrity only by a value $s$ which is not considered secret. In other words, when an IMD sets up a secure channel, it has no assurance that is has paired with a legit external device and only share a key for further communication. Second, both devices start measuring heartbeat signals simultaneously for the same time period. The measurements $\alpha$ and $\beta$ are bound to $s$ and then committed under key $w_A$ and $w_B$ respectively such that $C_A = Commit((\alpha, s)w_A)$ and $C_B = Commit((\beta, s), w_B)$. The IMD transmits its commitment $C_A$ first, and is followed by the external device who sends $C_B$. In the next step, the IMD sends $w_A$ such that the external device can verify $C_A \stackrel{?}{=} Commit((\alpha, s)w_A)$. If he succeeds, the external device sends $w_B$ to the IMD who verifies the commitment $C_B$. If correct, the pairing protocol completes successfully and both devices are authenticated.

The authors of H2H authentication demonstrate that the numbers generated from heartbeat measurements are truly random by applying a set of output sequences to the NIST suite of statistical tests [39]. Since the heartbeat can be modelled as a time-varying variable and is applied as value in the commitment scheme, *forward security* is ensured. In the experimental phase it was established two independent synchronous heartbeat measurements $\alpha$ and $\beta$ are often similar but not necessarily exactly the same, where the disparity is expressed as the error rate. To ensure the commitment verification succeeds, the authors have determined a threshold value $d$ and additionally verify if $dist(\alpha, \beta) \leq d$ to reduce the amount of false negatives when verifying the correctness of a received commitment $C$.

### 3.3.2. Analysis of Secure Key-Exchange

In Secure Key-Exchange, Seepers *et al.* [42] apply heartbeat measurements in order to successfully exchange a key between the IMD and external programmer. To facilitate a lightweight secure communication channel, the IMD employs symmetric key cryptography. This allows the external device to communicate with the IMD if and only if it has a shared secret with the IMD. To ensure the establishment of a shared secret in emergency situations, Secure Key-Exchange is proposed.

The authors consider an adversary model whose goal is to gain access to the IMD or obtain privacy

sensitive information. The protocol is designed to mitigate an active adversary who has full control of the channel, even during emergency situations. However, it is assumed the adversary is not able to measure the heartbeats as this requires physical access to the patient which is unlikely to have for an adversary. Additionally, it is assumed both devices are capable of measuring the heartbeat signals simultaneously.

Secure Key-Exchange is based upon a fuzzy commitment scheme, where the heartbeat measurements in this case are used to transfer a shared key from the IMD to the external device. After both devices measured the heartbeats simultaneously, the IMD generates a random key $K_{AB}$ and conceals it by committing it to its heartbeat measurement $\alpha$. The IMD sends its commitment $C_A$ along with a hash of the key $h(K_{AB})$ to the external device. By employing a strong error-correcting code, the external device can uncover the key $K'_{AB}$ by verifying the commitment with its own measurement $\beta$. Due to the error-correcting code, the commitment may still succeed although $\alpha \approx \beta$ holds rather than $\alpha = \beta$. The key $K_{AB}$ is verified by comparing $h(K_{AB}) = h(K'_{AB})$.

The error rate between independent simultaneous heartbeat measurements is compensated by employing a strong error-correcting code. During the simultaneous measurement, both devices are additionally required to classify any heartbeat misdetection to reduce the false negative rate. When the misdetection occurs, the device sends the value in plaintext to the other device to inform that specific measurement can be dropped. The complete protocol was implemented by transferring a 80-bit secret key and takes on average between 60 and 77.6 seconds to completely execute the key exchange protocol.

### 3.3.3. Analysis of IMDGuard

While the IMDGuard, proposed by Xu *et al.* [50], is a proxy-based approach, the security scheme also incorporates heartbeat measurements. The proposal uses heartbeat measurements to establish a shared key between the proxy device and IMD without the need for prior knowledge. In case the proxy device is lost, a new one should be able to pair with the IMD without the need for surgery, and can do so by measuring the heartbeats simultaneously with the IMD. Besides the explanation of the pairing protocol between the IMDGuard and IMD, the communication between the IMDGuard and external programmer is also described by the authors. This analysis however, will only consider the heartbeat based key exchange protocol between the IMD and the proxy specifically.

The adversary model for this security scheme also considers an adversary whose goal is to gain control of the IMD or retrieve privacy sensitive data. The attacker is expected to be capable of controlling the complete wireless communication channel. Additionally the adversary wishes not to be caught, thus physical contact with the patient is prohibited. The DoS attacks are specifically excluded in the adversary model, and the authors assume there is no adversary in an emergency situation.

To pair two devices, both start simultaneously measure the heartbeat signals. During the first round of the protocol, a set of heartbeats are measured and processed by both devices. For each processed measurement, the parity is calculated which is exchanged with the other device. In case the parities do not match, both devices discard the correlated measurement. The complete result $K_{AB}$ is hashed by the IMD and transmitted to the external device who verifies his own result $K'_{AB}$ by hashing it and comparing $h(K_{AB} = h(K'_{AB}))$. If the hashes match, a confirmation is send to the IMD. If they do not match, a repair process will be initiated by the external device. This process involves a similar parity checking technique, but now based upon a smaller segment of the measurement to detect measurement errors more accurately. This however also limits the extraction to less bits for each measurement and thus implies both devices need to measure more heartbeats.

The complete protocol establishes a key of 128-bits which requires the measurement of 43 consecutive heartbeats, if no measurement error occurs. To ensure the exact match of two simultaneous measurements, the transmission of at least $43 \times 2$ messages have to be exchanged. The authors have evaluated the temporal variance of heartbeat measurements to investigate if an adversary may be able to guess the secret sequence from heartbeat measurements if he has access to historic/future records of the patient. Due to the discovered statistical properties of the heartbeat measurements it is es-

tablished that an adversary can not guess more accurately even if he has access to patient specific of the past or the future. Additionally, the randomness of the extracted bit sequence is also verified by running the NIST statistical test suite [39].

### 3.3.4. Comparison of state-of-the-art

Comparing the state of the art solutions which incorporate heartbeat measurements into their security scheme in order to protect the IMD from adversaries who wish to gain illegitimate control or steal privacy sensitive data, reveals that all proposals consider an advanced adversary model who has complete control over the communication channel. Only IMDGuard does not expect the adversary to be present during an emergency situation. All three protocols are designed to mitigate a man-in-the-middle, spoofing and passive attacks, while none of them consider Denial-of-Service prevention.

Another similarity is that each study confirms the heartbeat measurements satisfy the properties to provide a random and time-varying sequence. H2H authentication and Secure Key-Exchange however, still require another (pseudo)random number generator in order to complete their protocol. H2H authentication employs heartbeat measurements for authentication only and requires a (P)RNG for the key establishment. Secure Key-Exchange take advantage of heartbeats to conceal a (P)RNG generated secret key. IMDGuard also uses heartbeats for key-establishment and differs from Secure Key-Exchange as they directly extract a symmetric key from the measurements without the need for any other (P)RNG, but requires a minimum of $43 \times 2$ message exchanges to establish a 128-bit key. None of the security schemes however, apply heartbeat measurements for both authentication as key establishment in one scheme without the need for another (pseudo)random number generator.

Regarding the strength of the security schemes, it was observed that H2H authentication is vulnerable to a reflection attack. By considering the scheme as represented in Protocol 1, where the external device is now an adversary without a legitimate $\beta$, the adversary can simply return the received commitment $C_A$ and $w_A$ to the IMD as there is no detection mechanism in the protocol for the IMD to detect the occurrence of a reflected value. Additionally, a more complex man-in-the-middle attack is possible if the adversary is able to transfer all communication trough his access point where the commitments will be reflected back to the devices. This would allow an attacker to eavesdrop on the communication and gain control over the IMD without being noticed.

Each security scheme overcomes the error rate of two independent simultaneous measurements differently. While H2H authentication verifies the correctness of a received measurement by a given threshold value, Secure Key-Exchange approves the disparity of measurements by performing strong error-correction. IMDGuard incorporated additional checkpoints for each heartbeat measurement to discard a measurement immediately if it is suspected to contain any error. This however also increases the amount of messages to exchange to at least $43 \times 2$, whereas Secure Key-Exchange and H2H authentication only require 7 and 5 transmissions respectively (excluding the amount of transmissions needed to establish a TLS session in H2H authentication).

While all three security schemes take different approaches to protect an IMD against an active adversary by means of symmetric key cryptography, it is concluded that the efficiency of the protocols can be improved by proposing a security scheme which incorporates heartbeat measurements for both authentication and key establishment. One open challenge is to create a security scheme such that there is no need for any other (pseudo)random number generator. Although all schemes consider a similar adversary model, the vulnerability in H2H authentication illustrates that it is not trivial to completely mitigate such advanced adversaries. Additionally, none of the protocols consider the challenge to synchronise the IMD and external device as all assume both devices are able to measure simultaneously. In terms of efficiency, the challenge is to limit the amount of transmissions required to detect measurement errors and minimise the rate two devices can not authenticate due to a measurement error which is also expressed as the false negative rate.

# 4

# Research challenges and methodology

*The identified vulnerabilities in current IMDs of their wireless communication capability makes it evident that, in order to maintain the privacy and security properties of a patient, there is a need for an authentication and key establishment protocol that satisfies the previously mentioned security, safety and utility goals. After investigating the different approaches proposing a solution within this landscape, it is concluded that it is very difficult to optimise all design goals at once. The analysis however, has also demonstrated that the heartbeat based approaches potentially allow a well balanced solution between the earlier described tensions.*

While several approaches for establishing a secure communication channel have been suggested, none of the suggestions achieve a complete balance between the tensions as identified by Halperin *et al.* [21]. The analysis performed in Section 3.3 shows which solutions remove the tension of one or two goals, but also indicate there is a need for one solution that solves three tensions at once.

Designing a complete protocol satisfying this need, is however a challenging task. As described by Boyd [5], many of the existing protocol problems result when designers are unclear about the protocol goals they are trying to achieve. To avoid common pitfalls in the protocol design, a methodology describing all aspects of protocol design and evaluation will therefore be followed. In this chapter, the research challenges that need to be overcome to ensure secure communication between an IMD and external programmer are outlined in Section 4.1. In Section 4.2 the methodology will be described in two phases, the design process and evaluation phase. Afterwards, the threat model of the scenario is presented which supports the choice of the design goals presented in Section 4.4. The chapter is concluded by describing the application setting in Section 4.5.

## 4.1. Research challenges

According to the analysis performed in Section 3.2, one approach to overcome the security and privacy concerns, without endangering the ability of an IMD to deliver its life-saving therapy adequately, is to reduce the three tensions as defined by Halperin *et al.* [21]. Since it has been established that **IMDGuard**, **H2H authentication** and **Secure Key exchange** apply heartbeat based methodologies to balance at most all but one of the identified tensions, the challenge in this field directs to the proposal of a security scheme which takes advantage of heartbeat measurements in order to ensure all three tensions are well balanced. This challenge is addressed in threefold: Firstly, appropriate heartbeat measurements need to be selected and processed such that they can serve as input for both key establishment as well as authentication. Secondly, the processed measurements need to be integrated into a security scheme describing a key establishment and authentication protocol. Finally, the applied security scheme must perform adequately in terms of the safety, security and utility requirements. The aspects of each challenge are considered separately, each of which will be addressed in this section.

### 4.1.1. Processing heartbeat measurements

The adoption of any physiological value in a security scheme requires an appropriate processing and selection procedure in order to retrieve valuable information that is considered effective for its application. A good entropy distillation process is needed when applying physiological values into a key establishment and authentication protocol to generate random numbers. Therefore to incorporate heartbeat measurements, the values need to be processed reliably in order to provide an unpredictable bit sequence. This challenge is addressed in two-fold:

#### Feature selection

Heartbeat signals can be measured in various ways, of which each approach may be more appropriate than another depending on the purpose of measuring heartbeats. When applied into a security scheme as a source of entropy, the selection of data needs to be to collected realistically for both the external programmer as well as the IMD without being easily reachable by adversaries. Depending on the characteristics of the selected data, particular features have to be chosen. The main challenge of this step is to choose features that contain suitable characteristics in order to extract them reliably without affecting the entropy value.

#### Feature processing

After deciding upon the data collection, typically retrieved from sensor measurements, and the feature selection step, the data should be processed such that it can be adopted into a security scheme. This is in particular a challenge as it requires advanced signal processing techniques to prepare the data first, after which a suitable feature extraction method should be performed. The process is not considered trivial as the preparation of the data should be able to remove artefacts and noise from the device dependent measurements. When the features have been extracted, they need to be processed such that the variables will become functional for further integration into the security scheme, while adhering to the properties of a good source of entropy.

### 4.1.2. Integration into a security scheme

Tackling the challenge of processing heartbeat measurements for a security scheme does not provide guarantee of protecting the wireless communication channel between IMDs and external programmers against adversaries. After the entropy distillation process, the values need to be integrated into one security scheme to protect the IMD from an adversary. Whilst the discussed solutions attempt to accomplish this common goal, they vary in the complexity of attacks to defend against. Therefore the second challenge is to integrate the measurements into a security scheme adopted to the environment of the IMD of which several factors contribute to in threefold. First one needs to define the complexity of attacks the IMD should be protected against, second the protocol should comply with a set of formal key establishment and authentication goals. The last step describes the challenge to synchronise the measurements of two independent devices in order to complete the security scheme successfully. These steps are described as follows:

#### Advanced threat model

The identified threats to current IMDs enabled with wireless communication capabilities may impact the health or compromise the privacy of a patient. To alleviate these security concerns specifically a risk oriented threat model needs to be created. This requires a designer to identify the adversary the IMD needs to be protected from and which threats should be mitigated by describing the attack strategy of an adversary. Since a compromised IMD may have a great impact to the patient, IMD security is considered to be important enough to assume an advanced adversary model. Protocols protecting against strong, active adversaries however is not straightforward, as illustrated by the break of such a model as described in Section 3.3.4 for IMD security.

#### Accomplish goals for key establishment and authentication

As recognised by Gollmann [17], protocol vulnerabilities occur due to the differences in interpretation of key establishment and authentication protocols in general. To avoid misunderstandings about the goals a protocol should achieve, it is therefore desirable to clearly identify authentication-, key- and user- oriented protocol goals separately as recommended by Boyd [5]. The challenge however is illustrated by the lack of agreement as to what desirable goals for authentication and key establishment are, especially if it concerns the design of a lightweight solution. To tackle this challenge,

the design goals for authentication and key establishment are assessed by following the risk oriented threat model while considering fundamental elements of authentication and key establishment.

#### Synchronisation of measurements

The security scheme will rely on the simultaneous measurement of two independent devices as an input for the key establishment and authentication protocols. Several works [42] [38] [50] take advantage of the measurements in a security scheme and address the challenge of measurement errors within their described scenario. An unaddressed challenge however is the difficulty of synchronisation. Heartbeat values are time-varying variables and need to be compared within the same timeframe to make the measurements meaningful for the purpose of authentication and key establishment. This requires two independent devices to be synchronised in time, without leaking secret information over an unprotected communication channel. To complicate the challenge of synchronisation even further, the method also needs to adhere to the same limitations as the complete security scheme such as minimisation of resource consumption.

### 4.1.3. Performance

The goals addressed in Section 4.4.1 and 4.4.2 concern the mitifation of security and privacy issues for the wireless communication channel between IMDs and external devices but does not devote effort to a balanced solution between the tensions as described by Halperin *et al.* [21]. While the utility properties of the security scheme will be expressed as extensional goals, the balance of the complete solution needs to be verified in the evaluation phase. The balance however, may differ depending on the IMD type. The challenge of evaluating the performance of the solution must therefore be carried out independently of a specific IMD design.

## 4.2. Methodology

To achieve the main research goal of enabling secure communication between an IMD and external programmer, we design a lightweight authentication and key establishment protocol based upon heartbeat measurements. To accomplish this objective, the followed methodology is divided into two stages: The first stage describes the steps to design the protocol, the second stage describes the process in order to evaluate the proposed protocol. By following the two staged methodology accordingly, all challenges as identified in Section 4.1 will be addressed. Both stages are explained in more detail in the following sections.

### 4.2.1. Design process

Designing a complete protocol to alleviate the the security concerns of wireless communication enabled IMDs, require the process to identify the threat model and decompose it into formal design goals. The formal design goals need to be satisfied when translated to the cryptographic mechanisms in order to create a complete security scheme. More specifically, the following steps will be performed:

1. Identify threat model
2. Extract design goals
3. Establish application setting
4. Organise the input methods
5. Translate to cryptographic mechanisms

The first two steps resemble the process of decomposing the identified threat model into formal goals for given scenario. This scenario is bound to the application setting as will be described in the third step of the design process. These three steps relate to a risk oriented approach while taking the utility oriented goals into account for the given landscape. The model and design goals will be described in Section 4.3, 4.4 and 4.5 respectively which overcome the first two aspects of the research challenge defined in Section 4.1.2. The heartbeat measurements will be organised and classified in the following step. More specifically, this stage addresses the research challenges as described in Section 4.1.1 in order to support the design goals as established earlier. The last stage of the design process concerns the

integration of all components into one complete security scheme by using cryptographic mechanisms and should satisfy the last aspect of the research challenge described in Section 4.1.2. Throughout the process, the performance requirements are taken into account. The analysis whether all challenges have been addressed appropriately however, will be performed in the evaluation step.

### 4.2.2. Evaluation

The research is aimed at proposing a security scheme by utilising heartbeat measurements in order to support the establishment of a secure wireless communication channel whilst maintaining a balance between safety, security and utility. The protocol is therefore evaluated based upon whether the identified challenges have been addressed by assessing the performance compared to the state-of-the art and the level of protection provided by the proposes security scheme.

Considering our protocol is designed to meet the requirements outlined in Section 4.4.1 and 2.2.1, with the objective to protect the IMD against the adversary described in Section 4.3, the focus of the first part of the evaluation will be on mitigating the attacks an adversary may perform. The risk oriented approach allows us to evaluate the security aspects of the protocol by describing realistic attack scenarios to assess how the solution provides protection. This scenario-based security analysis has also been conducted in similar work [32] [21] [42] as evaluation method.

To assess the performance in terms of accuracy and feasibility of the proposed solution, a proof of concept will be developed. The resource consumption of a hardware implementation however depends on the chipset, hardware acceleration, memory capacity and power management of the system. Since the hardware architecture may differ for each IMD type and device manufacturer, evaluating the resource consumption of the cryptographic primitives based on a experimental setup will be inaccurate. Therefore the proof of concept only serves the purpose of verifying the feasibility and quality of processing heartbeat measurements as a source of entropy and the correctness of the synchronisation method. The performance in terms of resource consumption and efficiency will be determined by means of a comparative study of the current state-of-the-art as described in Section 3.3.4. Additionally, the performance of the protocol will be put in perspective by comparing it against solutions who address the concerns of IMD security by one of the other four approaches.

## 4.3. Threat model for IMDs with wireless communication capabilities

Enabling an IMD with wireless communication capabilities allows external programmers to request medical information and improve the therapies tailored for every individual specifically, by means of a painless and convenient method. From the perspective of an adversary however, the wireless communication channel also introduces an entry to gain access to valuable information or to perform harmful activities. By applying a risk oriented approach, the creation of the adversary model and definition of his attack strategies, allow the derivation of the security requirements the authentication and key establishment. The protocol should comply with these requirements in order alleviate the security concerns of IMDs with wireless communication capabilities.

### 4.3.1. Adversary model

As established in Section 3.3.4 current state-of-the-art solutions with heartbeat based security for IMDs, always consider an advanced adversary model who is assumed to have complete control over the communication channel. More specifically, the capabilities of an advanced adversary model is described in Section 2.1.2 as one who may perform both active as passive attacks. Since a vulnerability in IMDs could compromise the health or privacy of a patient, IMD security is considered important enough by us to assume such a worst-case model. Additionally, the violation of the privacy properties are addressed in the adversary model by following the definition of the GDPR [49]. A complete description of the *goal*, *capabilities* and *boundaries* of the adversary model is as follows:

1. **Adversary Goal:** For this model an adversary is represented as a malicious entity whose goal is to gain control over or retrieve sensitive data from the IMD trough the wireless communication channel without being caught. A more detailed description of the notion of *sensitive data* is as

follows:
In accordance with the GDPR [49] that entered into force on the $24^{th}$ of May in 2016, sensitive data is in this case related to a special category classified as *data concerning health*. As defined in Article 4, paragraph 15, *data concerning health* represents: "Personal data related to the physical or mental health of a natural person, including the provision of health care services which reveal information about his or her health status.".
For example, data containing the therapy settings along with the patient's ID is considered as sensitive data. On the other hand, whenever the same data is encrypted and only readable by trusted entities, it does not reveal any sensitive data to other parties and is therefore not violating the privacy property of the patient.

2. **Capabilities:** To protect against an adversary who's goal is to gain control over retrieve sensitive data, one needs to identify what activities an adversary can undertake. These activities are captured by defining the capabilities of an adversary. Our security scheme will be designed for a strong adversarial model where the adversary has complete control of the channel by performing active and passive attacks. This means the adversary may eavesdrop on, modify, drop, and replay messages. It is realistic to design for a strong adversary model since it has previously been established that a successful attack can have a high impact on the patient [37] [21]. The use of this model is also supported by related work [42] [38] where similar adversary models are described.

3. **Boundaries:** The adversary is allowed to attempt any attack in order to gain control of the IMD by sending commands or retrieve sensitive information, but will be bound to a set of predefined capabilities. These capabilities are however assumed to be effective after the implantation of the IMD, implying an adversary can not compromise the device before it is completely set up. Similar to previously discussed adversary models, it is also believed the adversary may be present during the event of an emergency where it can interfere with the wireless communication channel. However, the described adversary wishes to stay unnoticed during an attack. This limits the adversary to wireless attacks only, as it seems unlikely to stay unnoticed when the attack requires physical access to the patient. Thus in case of an emergency, he may attempt to compromise the communication channel by means of wireless attacks but will not be able do so physically.

### 4.3.2. Attack strategy

With the adversary model in place, a classification of the attack strategy can be performed. The attack strategy entails the possible scenarios an adversary can undertake based upon the common attacks on wireless communication channels as in Section 2.1.1, and the currently feasible attacks identified by recent studies [32] [37] [21]. Although it is preferable to protect against all possible attack strategies, it is unfortunately impossible due to the tensions between the security, safety and utility goals as established by Denning *et al.* [9]. To ease the tension of these goals, the desired authentication and key establishment scheme should at least be protected against the following three strategies:

#### Impersonation of external device

Motivated to gain similar privileges as a legit external device, an attacker can attempt to impersonate the external device. Upon success, the attacker will be enabled to receive telemetry, remotely control the IMD and gain write access. These privileges allow an attacker to reach his main goal to gain access to the IMD and retrieve sensitive data. Radcliffe has shown the example of impersonation in [37], where an IMD believed an external device delivered legit information. As a result, the IMD could be convinced to deliver a wrong amount of insulin to the patient.

#### Off-line secret guessing

Researchers [32] [21] have shown it is possible to capture the (encrypted) data transferred over a wireless communication channel. The information contained in the transmission may still be valuable after a certain amount of time as it can be used for blackmail, identity theft or targeted advertising. An attacker can therefore still be interested in such information, even after the connection has been closed. Although limited in computational power, it is considered the adversary will attempt to reveal the secret within 10 years after transmission, the same amount of time as the average lifetime of an IMD.

#### Information re-use

With similar motivation as for impersonating an external device, an adversary can attempt to re-use previously gathered information. This includes the interception of old data in order to replay it at a later stage against an IMD. Depending on the collected data, such replay could provide an adversary with a (limited) set of commands like the initialisation of a connection or modification of a treatment. For example, as established in Section 3.3.4, H2H authentication [38] appears to be vulnerable to a reflection attack which falls in the category of information re-use. Upon successful execution of the reflection attack, the adversary is capable of gaining control over the IMD.

Taking these strategies into account, the authentication and key establishment scheme should protect an IMD against most of the attacks listed in Section 2.1.1. However, the scheme will not be designed to provide protection against attacks that aim to harm the availability specifically. While scenarios such as battery depletion and radio frequency jamming are a concern on the long term, it does not pose a threat immediately to either the safety or security of a patient. Whenever an attacker attempts to jam the signal, the patient will become aware of it and is likely to verify the noticed malfunction with an expert who will investigate the malfunction further.

## 4.4. Design goals for authentication and key establishment

In designing an authentication and key establishment protocol for the wireless communication channel of IMDs, a risk-oriented approach is taken where the focus lies on the attack strategy of an adversary. Following the first stage of the methodology, the next step is to extract specific design goals based upon the defined threat model. Since the security scheme requires authentication and key establishment, two classes of goals are considered: goals concerning entity authentication (also termed as *user-oriented goals*) and goals concerning key establishment (which is termed as *key-oriented goals*). Additionally, there is also a need to satisfy utility related requirements as described in Section 1.2.2. These requirements will be reflected into the *extensional goals*, to ensure the solution is balanced between safety, security and utility. A complete overview of the design process to extract the formal requirements is given in Figure 4.1.

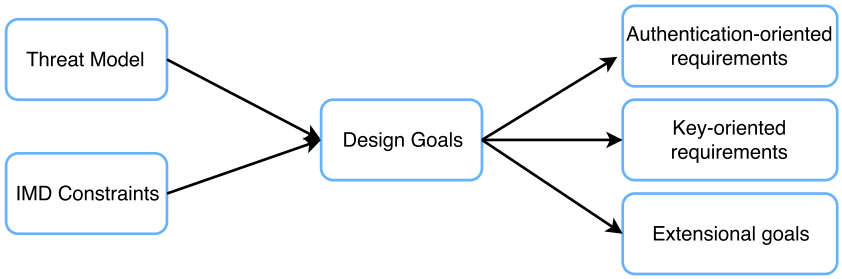



Figure 4.1: A graphical representation of the complete design process to extract the formal requirements for the security scheme.

### 4.4.1. Authentication-oriented requirements

Considering again the fundamental elements used in authentication, as explained in Section 2.2.1, certain properties lead to the definition of entity authentication. By taking the identified threat model as a basis, a subset of the goals from Section 2.2.1 have been selected in order to gain assurance the protocol is able to mitigate the considered attacks.

#### Unilateral authentication

According to the attack strategy, there is a need for a process where the IMD is assured of the identity of the external device by means of trust establishment. Achieving this goal will provide assurance to the IMD that the external device is legit and not impersonated, mitigating the first attack strategy as outlined in Section 4.3.2. There is no mention of the adversary impersonating the IMD, since this approach does not lead him to its goal. Therefore only unilateral authentication needs to be achieved: *Unilateral authentication occurs if only one entity is authenticated to the other.*

#### Entity authentication
Entity authentication resembles the process of convincing an entity *A* that another entity *B* has the identity it claims to be. From the perspective of a communication protocol design, this requires *A* to believe that *B* replied to a specific challenge. To achieve such belief, the IMD needs to challenge the external device who will be convinced of the identity of the external programmer if and only if the challenge has been answered correctly. In other words:
*The IMD (once) has had knowledge of the external programmer as its peer entity.*

#### Engagement
The last property for authentication requires the IMD to be convinced the external device replied to the challenge recently. In other words, it is relevant for the IMD to know that the legitimate external device is ready to engage in a communication with the IMD. Assurance about the engagement of the external programmer avoids the possibility of information re-use. Combining this goal with the goal of entity authentication leads to a *strong* definition of entity authentication as defined by Boyd [5], where the IMD is freshly aware of its peer entity. In other words, the challenge needs to be generated such that it is only used once and changes over time. More formally this goal is defined as:
*The external device is ready to engage in communication with the IMD.*

### 4.4.2. Key-oriented requirements
In general, a key in a cryptographic protocol is used to ensure the confidentiality property of the application. The cryptographic mechanisms are needed to refrain an adversary from gaining confidential information while the IMD and external programmer are communicating with each other. The key-oriented requirements are extracted based upon the assumption our protocol operates by means of a shared secret, since public-key infrastructures are considered unsuitable[42] [50] [38] within this landscape. The protocol will therefore rely on symmetric key cryptography based upon the establishment of a good key. As discussed in Section 2.2.3, a key is considered a good key if it satisfies a set of desirable properties. More specifically aligned with the attack strategy, this translates to the following goals:

#### Key confidentiality
The contents of the messages sent between an IMD and external programmer can only be hidden and revealed by using a key accordingly. To ensure the messages will be kept confidential outside of the trusted parties, the key should remain secret for untrusted entities as well. This requirement mitigates one aspect of the passive attack of eavesdropping and translates to the following goal:
*The key is only known to the IMD and (authenticated) external device.*

#### Key freshness
An extension to key confidentiality is the assurance of key freshness. During key establishment, the IMD needs to be able to verify the key is new and not replayed from another session otherwise the adversary is capable of re-using information in order to reach its goal. This requires the key to be dependent in time and session, formally described as:
*The key should be created such that it is considered fresh by the IMD.*

#### Key strength
In order to avoid the possibility of off-line secret guessing by an adversary. The key used to ensure confidentiality during the communication between an IMD and external device should be strong. As defined in the attack strategy, an adversary may spend 10 years time to reveal the secret. To avoid educated guesses from the adversary the created key should therefore be unpredictable. To avoid brute-force attacks, the key should be of such strength that the average time to guess a key should take more than 10 years, hence:
*The key is of such strength that it can not be broken within at least 10 years.*

#### Key confirmation
The last phase of key establishment enables the IMD to confirm both the IMD and external programmer are using the same key for further communication. Without this confirmation, a false key may have been established and disable the IMD to communicate appropriately with the trusted external programmer, therefore the following goal is defined:
*The key needs to confirmed by the external programmer to the IMD.*

### 4.4.3. Extensional goals

Since the protocol becomes useless whenever either one of the entities are unable to execute the protocol due to its shortcomings in capabilities, the extensional goals need to be defined. The extensional goals serve the purpose of establishing the readiness of the entities who engage in the protocol for key establishment and authentication. As explained in Section 2.3, shortcomings have their foundation in the availability of resources of the IMD. The readiness also depends on the utility requirements, because the security solution is considered nonfunctional if it affects the main functionality of the IMD which is the delivery of a treatment. This defines the following extensional goals:

#### Computational efficiency

The proposed protocol for key establishment and authentication should be efficient with regards to the amount of cryptographic computations required to complete the protocol. The protocol must have an adequate and realistic computational efficiency for use in real-time authentication requests on resource constrained devices such as the IMD.

#### Communication efficiency

The goal of communication efficiency aims to minimise the number and length of messages that need to be sent and received during the protocol. Therefore, the applied security scheme must reduce the number and length of the messages that need to be sent and received during the protocol.

#### No pre-established knowledge

In case of emergencies and other unpredictable events, there may be a need for an unknown external programmer to connect with the IMD for short-term therapy changes. With the introduction of a security scheme, legit external devices should still be able to connect within a reasonable amount of time without the requirement of time-consuming online pairing or off-line enrolment.

#### Minimal hardware requirements

There exists a number of IMD manufacturers who all produce a wide variety of IMD types with different characteristics. Thus in order to ensure the solution can be applied to these different instances, the protocol need to run independent on specific characteristics. Therefore the goal is to design a protocol which has the potential to perform well, regardless of the cryptographic primitives.

## 4.5. Application setting

The environment the IMD operates in depends on the requirements of the medical industry and the development process, enforced law and regulation and the wishes of the patients. The influence of these various parties illustrate the complex landscape to which the security scheme must adhere to. To obtain a well defined solution with clearly defined requirements, and to avoid misinterpretations of the goals of the security scheme, the application will be simplified by the following assumptions:

First, the goal of the security scheme is to provide a method for the IMD to establish a secure wireless communication channel with a legit external device. While there may be additional attack vectors due to the implementation aspects of the protocol or following messages exchanged after the establishment of a communication channel, our solution therefore is limited it to the security aspects related to the establishment of the channel only.

Furthermore, due to the long development process from manufacturer upon implantation of the IMD into a patient, it is considered unfeasible to know whether an IMD has been compromised along this process. The same reasoning holds for the usage of an external device. While hardware security remains an important topic for the security of IMDs, it is considered to be out of scope for this research. Therefore, it is assumed legitimate devices operated by trusted entities will follow the protocol as designed and can not be compromised. This assumption however does not rule out the possibility of spoofed external devices.

As already substantiated in Chapter 3, due to the constrained environment it has been established that public key infrastructures are considered too heavy in terms of resource consumption for IMDs. Therefore, the design of the security scheme will only consider the use of symmetric cryptographic

primitives. This design choice is in line with the reasoning of prior art as well as the draft on lightweight cryptography as published by NIST [34].

A very important consideration is the scenario how two devices are able to build trust among each other. While this problem is addressed in public key infrastructures by means of a trusted third party, this can not be applied into our security scheme due to the prior assumptions stating that public key infrastructures should be avoided in the solution. However, another approach to establish trust is upon physical contact. While this will hold for every active IMD as it is implanted in the patient and an honest device, the external device should proof its legitimacy by showing it has significant physical contact with the patients body.

Given these assumptions, and the goals set earlier in this chapter, the security scheme should be designed such that it satisfies all requirements and protects the IMD from the described threat model. However, all attacks performed that violate at least one of the described resumptions will not be considered as a protocol vulnerability. Additionally, the goal of this thesis is to design a solution for the given problem. The solution should mitigate feasible attacks performed against the IMD conforming the threat model based upon the design of the solution. Other attacks originating from the same threat model may exist due to implementation errors. The proposed solution will therefore assume a perfect implementation according to the design as will be presented in Chapter 5.

# 5

# Heartwear: Heartbeat security for IMDs

*In this chapter we present Heartwear, a security scheme for IMDs to ensure secure wireless communication by incorporating heartbeat measurements into cryptographic methods. Heartwear combines authentication and key establishment by adopting the same source of entropy efficiently to provide a secure wireless communication channel in order to access IMDs, even during emergency situations without the need of hardware specific changes.*

Previous work has incorporated heartbeat measurements into a security scheme to overcome the security and privacy concerns of the wireless communication enabled IMD in various ways. **H2H authentication** [38] first sets up a secure channel to provide confidentiality by means of a key exchange protocol. Heartbeat signals are used for authentication trough the exchange of measurements to proof each others identity. **Secure Key Exchange** [42] takes advantage of heartbeat signals by using the measurements to symmetrically encrypt a key before it is transmitted over an unprotected wireless communication channel. This approach differs from H2H authentication because it does not require the set up of a secure channel first, but commits to a key with the heartbeat measurements as a hiding value. **IMDGuard** [50] extracts a symmetric key directly from the measured heartbeat signals, by sending a minimum of $43 \times 2$ separate message exchanges.

Each of these three protocols process heartbeat signals to accomplish either authentication or key exchange. To address all security and privacy properties however, both methods are necessary. *Heartwear* therefore distinguishes itself from previous work, by incorporating heartbeat measurements in order to achieve both authentication as well as key establishment into one security scheme. The objective will be satisfied by carefully extracting significant characteristics from a heartbeat measurement, which will serve as the only RNG of the protocol. Additionally, *Heartwear* also overcomes the challenge of synchronising two independent devices by means of a novel method that does not require additional resource consumption from the IMD.

This chapter presents the design of *Heartwear* as a complete security scheme. Section 5.1 describes the practice of processing the selected measurements in order to prepare them for the entropy distillation process. The integration of the heartbeat values and a complete overview of the security scheme is illustrated in Section 5.2, which addresses both the authentication as well as key establishment phases. The chapter is concluded by providing a routine to synchronise two devices for this security scheme in Section 5.3.

## 5.1. Heartbeats as a source of entropy

Prior to the effective use of heartbeat features, a careful investigation needs to be performed such that the features are able to satisfy the authentication and key oriented requirements accordingly. First an appropriate method to measure heartbeat signals needs to be selected. Section 5.1.1 describes the benefits of several techniques to measure heartbeat signals and is concluded by choosing the most appropriate technique. Second, the heartbeats need to be modelled accordingly in order to identify

distinct characteristics which could serve as input for the entropy distillation phase. The last step is the entropy distillation process as described in Section 5.1.3 which includes the quantification of the extracted heartbeat values. By completing all these steps accordingly, the challenges as addressed in Section 4.1.1 will be overcome.

### 5.1.1. Data collection

In general, there exist two methods of measuring heartbeat characteristics externally and are commonly used for different purposes. An IMD however, especially those specifically designed to treat cardiac disorders, generally measure heartbeat values based upon an electrocardiogram (ECG) [29]. Another method for measuring heartbeats is by means of a photoplethysmogram (PPG) or Blood Pressure (BP) reading. While ECG sensors measure the electric potential difference which induces a cardiac contraction, PPG and BP measurements measure the pressure difference of the skin by placing a suitable sensor on the finger of the patient.

All three techniques are non-invasive methods for heartbeat measurements. While pressure based techniques PPG and BP do not require careful sensor placement to retrieve meaningful data, ECG recordings do. Since an ECG records the heart's electrical activity by the placement of an electrode pair, the placement of the sensors determine what specific characteristics of the heart are measured. However, PPG and BP measurements are less accurate when it comes to feature extraction because the changes in blood volume is affected by the strength of contraction, dissipation of the blood pressure through the body and motion artefacts. The representation of ECG measurements are only exposed to slight artefacts due to interference of nearby electrical devices and small noise differences if the electrodes have been placed correctly.

The application setting as described in Section 4.5 has established that an external programmer becomes a trusted entity if it can prove that it gained physical access to the patient. Consequently, since the heartbeat measurements ought to be used in the authentication process, the data should provide information that can only be obtained by means of physical contact. Additionally, since the protocol relies on the establishment of a shared secret trough heartbeat measurements, the internal and external device should be capable in extracting the characteristics of the measurements accurately.

The work of Seepers *et al.* [43] substantiates the qualities of the non-invasive techniques to measure heartbeats. In this work it is motivated that, while blood pressure based techniques can serve as a source of entropy in terms of the statistical properties of the generated bit sequence, they are also vulnerable to remote attacks. Due to the limitations in accuracy, blood pressure based techniques often tolerate a relatively high measurement error when comparing two independent readings. The higher tolerance allows an adversary to guess the secret measurements with a higher accuracy by analysing the patients skin differences from images obtained with a 50-FPS camera (equal performance to a regular webcam) . The results however also suggest that this method is unfeasible when ECG measurements are enforced in the security scheme as they improve the measurement accuracy and therefore maintain a lower tolerance for measurement errors.

Considering the requirement of measurements that should only be possible to obtain upon physical contact, and the differences between PPG, BP and ECG measurements as substantiated in the work of Seepers *et al.* [43], it is concluded that the ECG measurements are most suitable for the application of heartbeat security. This mitigates possible remote attack scenarios, and allow a higher accuracy in measurement due lower measurement error rate. Therefore *Heartwear* requires both the IMD and external device to measure ECG signals in order to be able to complete the protocol.

### 5.1.2. Modelling heartbeats for feature selection

Heartbeat signals provided by ECG signals are represented as a time varying waveform. The ECG waveform is measured in milivolt (mV), showing the electric potential differences over time of the heart. Two regular heartbeats from an ECG are graphically represented in Figure 5.1. The most visual parts of one ECG waveform consists of a very large upward deflection, called the *R-peak*. The *P-wave* occurs slightly before this peak and is proceeded by a downward deflection, called the *Q-valley*. Directly after the *R-peak* the *S-valley* develops and is followed by the *T-peak*. All of these visual parts

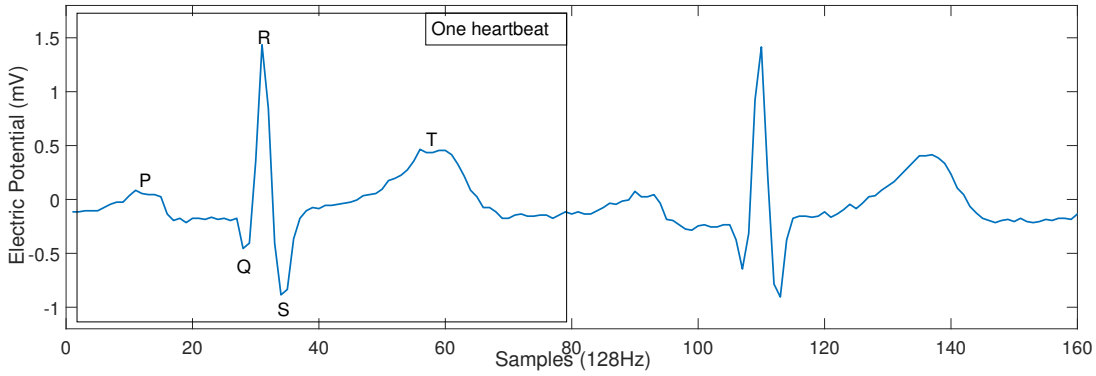



Figure 5.1: Two heartbeats, including the annotation of the graphical deflections

have also been annotated accordingly in Figure 5.1.

The most distinct feature of the ECG recording is the *R-peak*, the highest upward deflection which represents the contraction of the heart. Medical implants already incorporate the measurement of R-peak values for various reasons, including the continuous monitoring of a patients heart rate, detection of irregularities and long term analysis. As a consequence, many types of IMDs already have the required sensors for R-peak detection in place as confirmed by medical device manufacturer Medtronic who provided a list [1] of features implemented in several IMDs. These devices are for example, but not limited to, pacemakers, insertable cardiac monitors (ICM), cardiac re-synchronisation therapy (CRT) and implantable cardioverter defibrillators (ICD). This makes the *R-peak* a suitable candidate to incorporate into our security scheme.

### 5.1.3. Quantification of heartbeats

While the *R-peaks* are a biometric feature that can be measured accurately by various IMDs, there is no guarantee it provides the properties of a reliable source of entropy. As specified by Jain *et al.* [24], a good biometric for authentication is one that is easily measured for the general population and unique for every individual. Additionally, for key generation, it should vary over time and provide a suitable and unpredictable bit extraction method to avoid off-line secret guessing. This requires the *R-peaks* to be quantified, such that they provide these properties.

As described previously, *R-peaks* occur only once for every single heartbeat. Measuring the amount of *R-peaks* within a given time-frame allow the calculation of the heart rate (e.g. in beats per minute) of an individual. Another physiological phenomenon can be quantified by computing the time between two consecutive *R-peaks* and is commonly referred to as the inter-pulse interval (IPI) or RR-interval. Given the time an *R-peak* occurs is represented as $R_i$ and the time the next *R-peak* is found at $R_{i+1}$, then the correlating IPI is calculated as: $IPI_i = R_{i+1} - R_i$. Figure 5.2 graphically represents the identification of IPI values.

For symmetric key cryptography, the key is usually expressed as a binary sequence. In order to extract a key from IPI values, the measurements therefore need to be encoded to a binary format such that particular bits may form a key together. Without going into implementation details yet, a key can be extracted from IPI values as follows: First one needs to record $n$ IPI values and convert each value to a binary representation of $l$ bits long. By carefully selecting $m$ suitable bits of each value, and concatenating the bits together, an $n \times m$ bits binary sequence can be created. The selection of a suitable bit depends on the amount of entropy the bit can deliver and the possible measurement error for the specific bit.

A similar approach for the generation of a random bit sequence has been followed by [42] [38] [50] and applied to either authentication or key establishment in their protocol. These studies also inves-

[1] `https://www.medtronicacademy.com/medtronic-device-features-2`, Retrieved November 2017

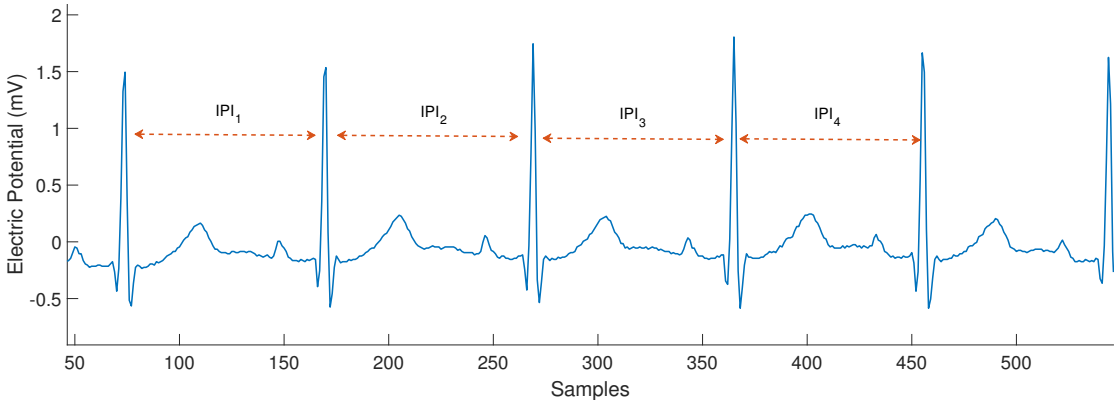



Figure 5.2: Graphical representation of IPI values

tigated the quality of the *entropy distillation process* by performing statistical tests [39] and calculation of Shannon's entropy [44]. According to their results, the first 5 least significant bits (LSB) of an 8-bit binary representation of an IPI value are considered suitable as a RNG, where the $5^{th}$ LSB has an entropy of 0.98 [38]. It is therefore concluded that each IPI value, represented as an 8-bit binary string, contains at least 5 suitable bits for the generation of a random bit sequence.

## 5.2. Integrating heartbeats for key establishment and authentication

For providing authentication and key establishment, *Heartwear* will adopt the recorded ECG signals by distilling specific bits from the calculated IPI values. The use of IPI values should establish a level of trust to the IMD such that the predefined requirements for authentication are satisfied, while the bits extracted from those IPI values also serve the key generation method. To accomplish this, the security scheme will rely on the concept of a challenge-response method and several other cryptographic primitives as described in Section 5.2.1.

The formal representation of *Heartwear* is given in Section 5.2.2, where the execution of the protocol will be described accordingly. Prior to the execution of the protocol, the external device and IMD do not share any information on beforehand. To establish secure communication between two devices, *Heartwear* uses the assumption an IMD can trust an external device trough physical contact of the patient. While *Heartwear* needs to protect the IMD against an advanced adversary model, it is also given the IMD will follow the protocol as designed since the adversary will only attempt to reach its goal by attacking the wireless communication channel which excludes the adversary to compromise an IMD before implantation.

### 5.2.1. Cryptographic preliminaries

*Heartwear* has been designed to it satisfy all requirements as described in Section 4.4. To achieve this, the security scheme approaches the problem by applying a symmetric key cryptography to maintain the confidentiality of the messages sent over the wireless communication channel. More specifically, the complete protocol consists of a combination of the following cryptographic primitives: a variant of the challenge response protocol, a key establishment scheme and a one-way hash function.

The challenge response protocol in *Heartwear* is a variation of the two-pass unilateral authentication protocol (**two-pass)** as defined in ISO/IEC 9798-2 [23]. In Two-pass, the verifier generates a nonce and sends it to the prover, where a nonce is a time-varying number that is only used once. The prover replies to the verifier with a token consisting of a message encrypted by a shared secret. On receiving this message, the verifier may now deduce that the prover can be trusted, if the shared secret is only known by trusted entities.

The shared secret in *Heartwear* is represented as the key used by a symmetric cipher to achieve encryption. Since the key should be chosen such that it is computationally unfeasible to be deduced during its lifetime in order to avoid off-line secret guessing attacks, the key needs to have a certain

security strength. To illustrate, suppose the minimal security strength of a key should be $x$-bits and each IPI can provide $\frac{x}{n}$ bits, it will then require $n$ IPI measurements in order to extract a complete key. For the remainder of this chapter, we assume $n$ IPI measurements are required in order to create a key with a strength of $x$-bits.

*Heartwear* also incorporates a one-way hash function as a building block. In short, a hash function $h$ creates an irreversible one-to-one mapping of an arbitrarily long input $m$ to a fixed length output, called the message digest and denoted as $h(m)$. Given that the hashing function used in *Heartwear* is pre-image and collision resistant it should be unfeasible to 1) compute the input $m$ for a given digest $h(m)$ and 2) find any two different inputs $m$ and $k$ such that $h(m) = h(k)$. While a hash function is typically carried out to verify the integrity of a received message, *Heartwear* applies this technique to verify whether both the IMD and external device have computed the same symmetric key without exchanging any information about the key itself.

Table 5.1: Explanation of symbols used in *Heartwear*

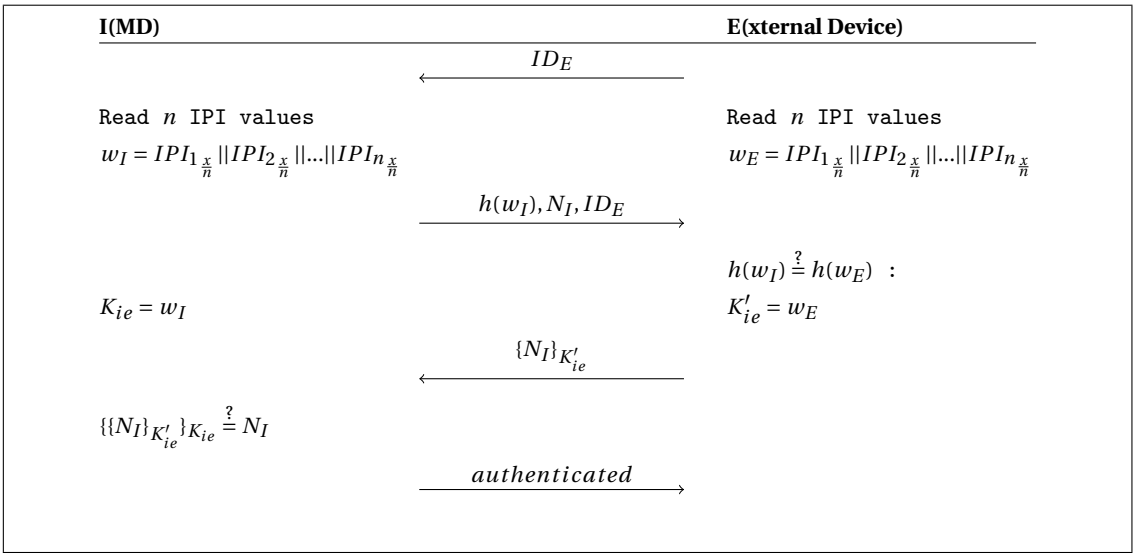
| **Symbol** | **Representing**: |
|---|---|
| $I$, $E$ | IMD and External device |
| $ID_E$ | An identifier of $E$ |
| $w_I$, $w_E$ | Witness generated by $I$ and $E$ respectively |
| $IPI_i$ | The $i^{th}$ measured IPI value |
| $IPI_{i\frac{x}{n}}$ | $\frac{x}{n}$ bits of the $i^{th}$ IPI value to ensure $x$-bit security |
| $\|\|$ | Concatenation of values |
| $h(.)$ | A one way hash function |
| $N_I$, $N_E$ | Nonces chosen by $I$ and $E$ respectively |
| $F \stackrel{?}{=} G$ | Verify that $F$ and $G$ evaluate to the same value |
| $K_{ie}$ | The shared secret calculated by the IMD and external device |
| $\{M\}_K$ | Symmetric encryption of message $M$ with key $K$ |

### 5.2.2. Heartwear: The protocol

The complete *Heartwear* protocol is a combination of the cryptographic primitives as explained in Section 5.2.1. Without relying on any pre-established knowledge, both entities need to establish a symmetric key and due to the unilateral authentication requirement, the external device needs to authenticate to the IMD. In order to do so, the external device aims to gain trust from the IMD through the measurement of IPI values to which both devices have access to. An explanation of the symbols used in the description of the protocol design is provided in Table 5.1.

The foundation of establishing trust by the external device to the IMD is employed by extracting $\frac{x}{n}$ bits from $n$ IPI's which are considered random in order to achieve a $x$-bit secure secret. The IMD and external device both need to measure the heartbeats simultaneously of the same patient to provide this trust, based upon the assumption an entity can be trusted if it can gain physical contact to the patient. The measurement of IPI values need to be processed such that the rightful bits can be extracted in order to generate the shared secret. Both the IMD and external device create the result independently and are called the witness $w_I$ and $w_E$ respectively. The witness of the external device is used to create a token for the two-pass unilateral authentication protocol. Additionally, the witness remains secret by parties who do not have physical access to the patient, by ensuring the witness is not sent in plaintext over the communication channel. Therefore, the witness serves as secret for the key establishment phase. A complete overview of *Heartwear* is represented in Protocol 2.

In *Heartwear*, the external programmer initiates a session by sending an identifier $ID_E$. Immediately after initiation, both the devices start measuring $n$ IPI values simultaneously of the same patient. For each IPI value, $\frac{x}{n}$ bits are extracted and concatenated in order to create witness $w_I$ and $w_E$. Note that measuring simultaneously with two unconnected devices is not trivial, since it requires careful synchronisation. Section 5.3 will describe in detail the proposed synchronisation method to enable simultaneous measurements, until then it is assumed both devices measure the same time-interval to




Protocol 2: *Heartwear*, a security scheme to establish a secure communication channel between the IMD and external device

collect $n$ IPI's.

As part of the challenge response protocol, the IMD sends a nonce $N_I$ as a verifier, together with the received identifier $ID_E$ of the prover and a hash $h(w_I)$ from its own generated witness $w_I$. Given the characteristics of an ideal hash function, $h(w_I)$ does not reveal any sensitive information about the secret $w_I$. Upon receiving the response from the IMD, the external device will hash $w_E$ in order to compare the two witnesses. Since $h(w_I) = h(w_E)$ only holds if $w_I = w_E$, the verification of $h(w_I) \stackrel{?}{=} h(w_E)$ allow the external device to belief both entities have generated the same witness. Given this confirmation, the external device establishes $w_E$ as the shared key $K'_{ie}$.

In order to confirm the key $K'_{ie}$ to the IMD, the external device encrypts the nonce $N_I$ with a symmetric cipher. Due to the characteristics of symmetric key cryptography, the IMD is able to reveal the nonce $N_I$ by decrypting the received ciphertext with the key $K_{ie}$. If the decrypted message results in the nonce $N_I$, the external device confirmed to the IMD that both devices obtained the same key. The IMD then completes the protocol by sending an authentication verification to the external device. Upon completion of the *Heartwear* protocol, both devices can communicate securely with shared key $K_{ie}$ as long as the session is active.

This protocol provides key establishment and authentication within the exchange of 4 messages. The authentication-oriented requirements as described in Section 4.4.1 are satisfied due to the incorporation of the two-pass unilateral authentication protocol where the challenge response also allows the IMD to have knowledge of the external device as its peer entity. The engagement goal is achieved due to the definition of the nonce. Since the nonce is a time-varying number which may only be used once, the IMD is freshly aware of its peer entity.

By following the *Heartwear* protocol, three out of four key-oriented requirements are also satisfied. First, the key confidentiality property is maintained since the witness is not transferred trough an unprotected channel. Due to the physical access assumption, only legitimate entities are able to extract the key. Second, the key is extracted based upon a time-varying variable. Therefore, the characteristics of the concatenated IPI values maintain the key freshness property. Lastly, the key is confirmed by the external device by encrypting the received nonce $N_I$ with the extracted key $K'_{ie}$. This step also allows the IMD to confirm both devices have obtained the same key.

Given this analysis based upon Protocol 2, it has been established that all authentication-oriented requirements are satisfied, as well as three out of four of the key-oriented requirements. The last requirement describes the required strength of the protocol, which will be evaluated after Chapter 6 as

the strength of the key depends on the statistical characteristics of the generated bit sequence.

## 5.3. Synchronisation of two devices

Essential to succeed a complete round of *Heartwear* is the synchronisation between the IMD and external device such that they can measure the IPI values simultaneously. Because the IMD and external device do not share any pre-established knowledge, the use of counters or clocks to verify if both devices are synchronised without additional resource consumption seems unfeasible. Since the external device is less restricted in resources, this challenge is tackled by deploying the synchronisation method to the external device such that the IMD does not suffer from additional computational overhead. The additional subroutine will be executed on the external device during one round of *Heartwear*, and is described in Algorithm 1.

**Algorithm 1** - Synchronising the measurement of $n$ IPI values

1: **Init:** $W \rightarrow IPI_{1\frac{x}{n}} || IPI_{2\frac{x}{n}} || ... || IPI_{n\frac{x}{n}} || ... || IPI_{n+i}$ from $t_0$ to $t_i$
2: **Init:** $h(w_I) \rightarrow h(IPI_{1\frac{x}{n}} || IPI_{2\frac{x}{n}} || ... || IPI_{n\frac{x}{n}})$ from $t_d$ to $t_{d+n}$
3: **for** $j = 0$ to $j = W.length - n$ **do**
4: $\quad w_E \rightarrow IPI_{1+j\frac{x}{n}} || IPI_{2+j\frac{x}{n}} || ... || IPI_{n+j\frac{x}{n}} \in W$
5: $\quad$ **if** $h(w_I) == h(w_E)$ **then**
6: $\quad\quad$ **return** true
7: **return** false

Suppose the external programmer initiates the execution of *Heartwear* according to Protocol 2, and the IMD starts measuring $n$ IPI values after receiving the request of the external programmer. Then the subroutine of Algorithm 1 starts directly after the external device initiates the protocol. This allows the external device to start measuring at least at the same time as the IMD. Thus the external device collects $n+i$ IPI values starting at $t_0$ and stores them in $W$, whereas the IMD generates $w_I$ from time $t_d$ to $t_{d+n}$. The time $t_d$ represents the time between $t_0$ and the time it takes for the IMD to receive the authentication request from the external device, $t_{d+n}$ is the time it takes to measure $n$ IPI values starting from $t_d$. The external device captures a larger set of IPIs in $W$ starting from $t_0$ to $t_i$ where $t_i$ represents the time the external device receives $h(w_I)$ from the IMD.

The subroutine represented in Algorithm 1, enables the external device to determine which subset of collected IPI values is equal to the set of IPI's collected by the IMD. Since the witness generated by the IMD is concealed by the hash $h(w_I)$, the algorithm applies a method to find the matching set of IPI values by executing the for loop. For each round the external device creates a hash of the generated witness, where $w_E$ is a collection of $n$ IPI values from $IPI_{1+j\frac{x}{n}}$ to $IPI_{n+j\frac{x}{n}} \in W$ . Due to the characteristics of a hash function, the output of $h(w_I)$ should be equal to $h(w_E)$ if and only if $w_I == w_E$. So whenever the `if` statement is satisfied, the external programmer is ensured both witnesses are equal and the external device can exit the subroutine.

In case the hash values do not match, the algorithm proceeds to the next round of the for-loop. As defined on rule 4 from Algorithm 1, the contents of witness $w_E$ will now contain $n$ IPI values from index $1+j$ to $n+j$ of $W$ such that $w_E \rightarrow IPI_{1+i\frac{x}{n}} || IPI_{2+i\frac{x}{n}} || ... || IPI_{n+i\frac{x}{n}}$ . The witness will be hashed again and checked against $h(w_I)$ to verify if both witnesses are the same. This process is repeated for a finite amount of times (in the algorithm the length of $W-n$ times) or until a match is found. The external device succeeds in finding the matching set of IPI values because it collects IPIs over longer time period such that the witness of the IMD occurs within the given timeframe. A formal proof of correctness that $w_I$ is a subset of the measurements performed by the external device is given in Section 7.1.

# 6

# Experimental setup and implementation

*This chapter explains the setup and implementation of the proof of concept (PoC) developed according to the Heartwear design. The design is based upon a collection of freely available databases of both healthy subjects as well as subjects who suffer from a heart condition. The PoC realises both the feature selection and processing of the ECG signals as well as an implementation of the synchronisation method. The conducted experiments are followed to verify the quality of the entropy distillation process in terms of entropy and the bit error rate.*

To convert the ECG values into IPI values in order to extract bits from them such that a witness can be generated, the datasets need to be processed accordingly. Processing raw ECG signals can be performed on different platforms, for which we have chosen to perform all steps to create the bit sequences in Matlab R2016a. The PoC has been implemented on a system with an Intel Core i7-4700MQ CPU @ 2.40GHz and 8.00 GB RAM, running Windows 10 64-bit home edition. On the operating system Matlab R2016a has been installed to process the ECG signals, together with Java 8 to run Matlab. The synchronisation method is developed in Eclipse release 4.5.2 for Java 8. The complete system setup allows the implementation and execution of various experiments of the *Heartwear* protocol.

In Section 6.1 the experimental setup is outlined by first describing the datasets used to perform the experiments with. Thereafter, the data is pre-processed in order to prepare it for the IPI detection method which is represented in Section 6.1.3. After the experimental setup, the implementation of the bit extraction and synchronisation method will be discussed in Section 6.2. The quality of the entropy distillation method and performance of the protocol will be addressed in Chapter 7.

## 6.1. Experimental setup

The core of the *Heartwear* protocol relies on the bit extraction method obtained from measured IPI values. As established in Section 5.1.1, ECG signals are considered the preferred method to collect IPI values for the presented solution. Lacking the ability to obtain IPI measurements directly from a patient, the PoC will be built upon freely available, and representative, datasets. To ensure the selection of representative data, we first describe how the data is collected in practice and which datasets are chosen to perform the experiments with in Section 6.1.1. Thereafter the raw ECG signals need to be processed to increase the performance of the peak detection method. The noise reduction and detrending method will be outlined in Section 6.1.2. The last step of the experimental setup describes the peak detection method and IPI calculation in Section 6.1.3.

### 6.1.1. Dataset

For the experimental setup, a freely available databank called *Physiobank*[1] is accessed to select appropriate datasets for our experiments. Physiobank contains a large amount of well-characterised digital recordings of physiologic signals, including many sets of ECG recordings from both healthy subjects and patients suffering from various heart conditions. Multiple collections from Physiobank, in specific the ECG databases, will be used for the experimental setup in order to verify the feasibility and limitations of *Heartwear* from the perspective of applying heartbeat signals for our scenario. A datasets contains at least one multichannel ECG representing a recording of one patient to whom multiple electrodes are attached to at the same time to provide simultaneous views of the heart by investigating the results retreved from various leads. Before going into detail which datasets have been selected, we first will describe the process of collecting ECG signals briefly.

#### Electrode placement

An ECG signal is recorded by placing electrodes on the body to measure voltage fluctuations. A combination of two electrodes form a lead, and depending on this combination each lead provides a different view of the hearts electrical activity. Most commonly method to diagnose the heart in the medical industry is the placement of 10 electrodes to create a 12-lead ECG recording [7]. Without going into too much detail, 6 out of 10 electrodes are placed in a specifically defined area to create leads `V1,V2,V3,V4,V5,V6`. The remaining 4 electrodes are placed on each limb of the subject and provide measurements of leads `lead I, lead II, lead III, aVR, aVL, aVF`. While each different lead does not affect the wavelength (or period) of the ECG recording, the waveform itself however (or amplitude of the independent deflections) may vary. For example the upward deflection of the *R-peak* is visually smaller compared to the deflection of the *S-valley* for lead V1 trough V3, whereas the the *S-wave* becomes smaller in lead V4,V5 and V6. A more detailed explanation on electrode placement and ECG interpretation can be found in other literature such as [10]. The analysis of two leads from the same subject within the same time frame simulate the representation of two simultaneous recordings of two independent devices as the recording is extracted from different sensors. Depending on the lead configuration however, the comparison of one set of leads may be more representative than another as will be elaborated further in Section 6.3.2.

#### Data collections included in the experiment

All the selected datasets originate from Physiobank and have been described in detail by Goldberger *et al.* [15]. Each selected dataset is named after the officially documented name from [15] and will be described briefly to provide insights to the relevant characteristics of the dataset for our experiments.

The first dataset used in our experiment is the *MIT-BIH Normal Sinus Rythm* Database (nsrdb) and is selected because this dataset has also been analysed by related work [42] [38] [50] to extract IPI values for their solution. The database contains 18 2-lead ECG recordings obtained from 5 men, aged 26 to 45 and 13 woman aged 20 to 50. The long-term ECG recordings are found to have no significant arrhythmia's. Every sample contains a recording of approximately 2h long recorded at a sampling frequency of 128Hz and provides a 2-channel recording of lead I and lead II.

To identify the discrepancy in entropy for patients suffering from a heart condition, the *MIT-BIH Arrythmia Database* (mitdb) is adopted. This dataset contains 48 half-hour ECG fragments from 47 subjects of 25 men aged 32 to 89 years old and 22 woman aged 23 to 89 years who all suffer from arrhythmia. For our experiment only 15 half-hour recordings are used with the same lead configuration: the upper signal is obtained by lead II, while the lower signal represents the recording of lead V1. Each ECG sample is recorded at a sampling frequency of 360Hz.

The third database is provided by Physikalisch-Technische Bundesanstalt and is referred to as the *PTB* collection. The database contains 549 high-resolution recordings from 290 subjects of 16 input channels each. While the obtained recordings are digitised at 1000 samples per second (1kHz), higher resolution samples are available on special request to the contributors of the database. Each record includes 12-leads measured signals following the conventional electrode placement as described previously.

[1] `https://physionet.org/physiobank/`, Retrieved November 2017

### 6.1.2. Signal processing

Each record for each dataset is imported in Matlab and consists of one large `.mat` file and a `.info` file. The `.mat` file contains all raw measurements of one record whereas the `.info` file describes the source name, duration, sampling frequency, interval, gain, base and units of the accompanied `.mat` file. A brief inspection of each `info` file gave insights to the general characteristics of the datasets, and are summarised as follows:

- All ECG records are measured in $mV$
- Each ECG record is an $n \times m$ matrix where each *row* contains the samples of one signal obtained from one lead, each *column* contains a sample of each signal.
- $1 \leq n \leq 16$ and depends on the amount of channels contained in the ECG recording. $38400 \leq m \leq 1000000$, based on the sampling frequency and duration of the recording.
- Each measurement is represented as a collection of raw units, varying in base and gain.

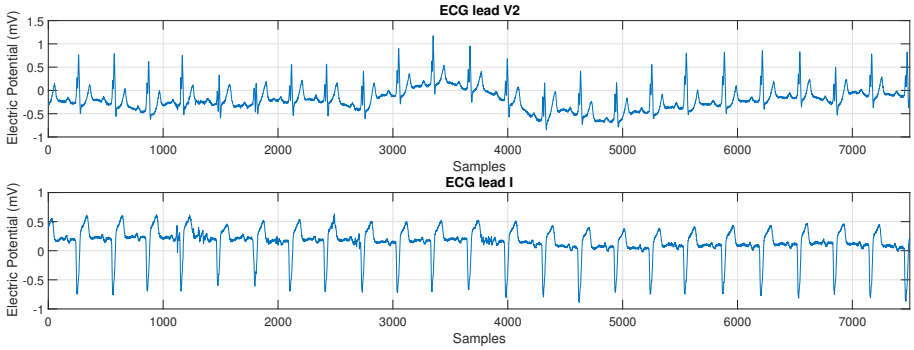



Figure 6.1: Representation of an unprocessed two-channel ECG signal from mitdb (record 107 on interval 16400-25800)

To represent a multi-channel ECG signal visually in Matlab, one needs to execute the a matlab script as provided by Physionet[2]. By extending the script with baseline and gain correction as recommended in the `.info` file, the signals can be loaded into Matlab appropriately. The result of one ECG record loaded into Matlab is represented in Figure 6.1. A visual inspection of the data reveals that the ECG records consist of raw signals containing a nonlinear trend and noise. In order to extract the ECG features of interest reliably, it is therefore concluded we need to detrend and remove the noise such that the result can be interpreted more accurately.

While there exist various approaches [48] [30] [27] to eliminate non-linear trends and noise from raw ECG signals, it is decided to address two commonly applied techniques and select the approach which accentuates the R-peaks the clearest. The first technique to process ECG signals appropriately has been suggested in a tutorial style by Matlab[3]. To detrend the ECG signals with this approach, a low-order polynomial was fit to the raw data, where the order $o$ represents the largest exponent of the polynomial. The fitted polynomial of order $o$ is subtracted from the original input to remove non-linear trends from data. As proposed by Tarvaine [47], another approach for detrending ECG signals is to remove the low-frequency component. This can be achieved by applying a fast Fourier transform (FFT) to the ECG signal to identify the low and high frequency components. By removing the low frequency components and restoration of the ECG signal by an inverse FFT, the signal should be free from trends and noise accordingly. Both techniques rely on specific parameter settings, for which the parameters have been set according to a trial-and-error approach. The final results of both techniques are represented in Figure 6.2.

As can be observed from Figure 6.2, the attempt to remove noise and trends from the data by fitting a

[2] `https://www.physionet.org/physiotools/matlab/plotATM.m`, Retrieved September 2017
[3] `https://nl.mathworks.com/help/signal/ug/remove-trends-from-data.html`, Retrieved September 2017

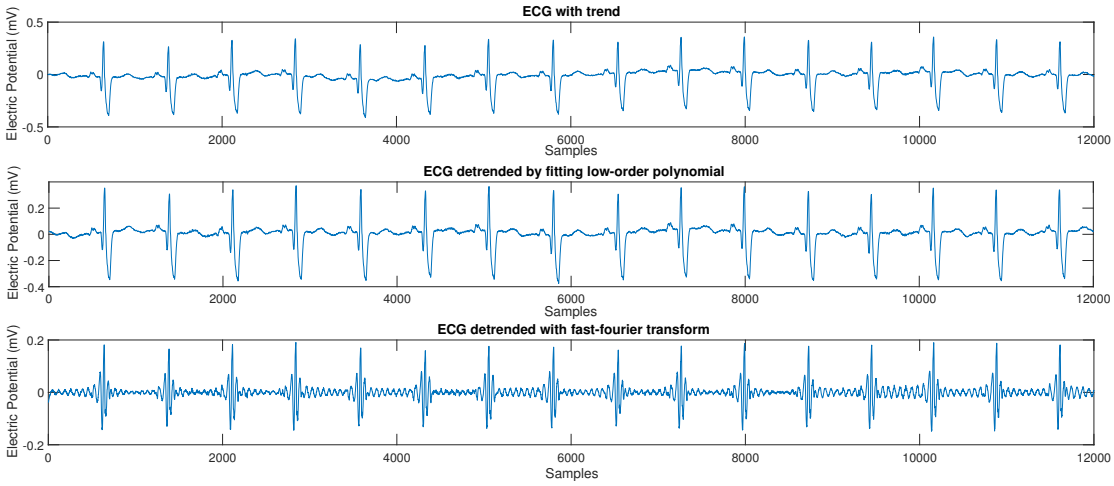



Figure 6.2: Comparison of two detrending techniques.

low-order polynomial did not return significant improvements but also did not loose significant characteristics of the data. The FFT approach however returned a very stable result where the complete signal aligns well on the x-axis. On the other hand, the FFT technique also lost the significant *T-peak* of the heartbeat signal. Nevertheless, it is decided to process all signals by means of the FFT technique since the prominent feature, the *R-peak* remains clearly visible.

### 6.1.3. IPI detection

The next step in the processing phase is to detect *R-peaks*. While there have been several methods proposed throughout the years for accurate *R-peak* detection [31] [47] [28], this approach will be based upon the standard peak detection technique for ECG signals as provided by Matlab[4] which, according to Matlab, results in a hitrate of 100% and no false positives.

The function `findpeaks` detects local maxima given an input vector $x$. Parameters $MinPeakHeight$ and $MinPeakDistance$ set boundaries within what range the peak should occur and allow a more accurate execution of the peak detection method. After setting the boundaries for the peak detection method correctly, based on the minimal height a peak may occur and minimal distance between two peaks, it was observed that a 100% hit rate and no false positives are only achieved if the selected signals distinctively represent the *R-peaks*. Figure 6.3 shows the results after applying peak detection to the processed ECG signal together with the original input of the same ECG.

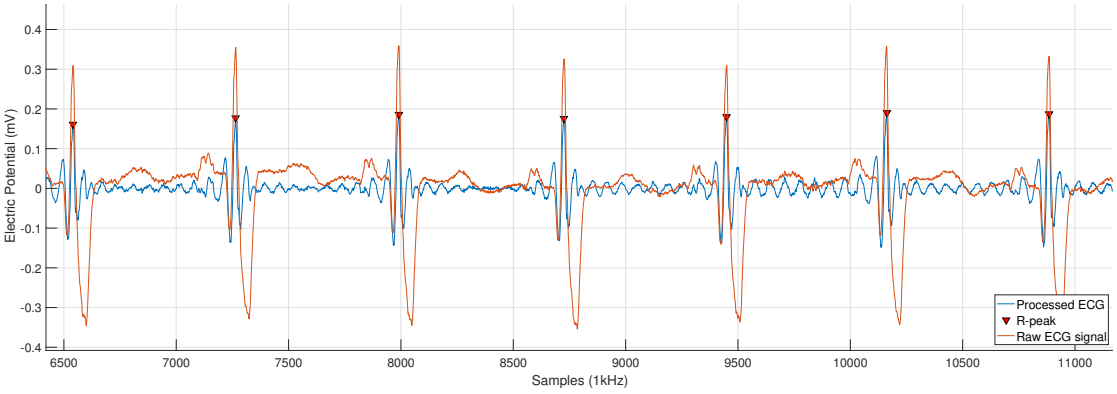



Figure 6.3: Graphical overview of a processed and unprocessed signal from the PTB database, sample s0010, lead I.

The IPI values of a signal can now be calculated by subtracting the location of one *R-peak* $R_i$ from

[4] `https://nl.mathworks.com/help/wavelet/ug/r-wave-detection-in-the-ecg.html`, Retrieved August 2017

its subsequent *R-peak* $R_{i+1}$ such that $IPI_i = R_{i+1} - R_i$ and will be expressed in our PoC as the amount of samples between two consecutive *R-peaks*. The complete set of IPI values are stored in a separate vector for each ECG signal to allow potential troubleshooting during the experimental phase without the loss of information. In line of related work [43] [38], each IPI value is converted to an 8-bit binary string in order to select suitable bits. The process of selecting the bits will be identified during the experimental phase as described in Section 7.2.

## 6.2. Implementation

After calculating each IPI value accordingly, and conversion to an 8-bit binary string, the data has been prepared but do not yet provide a random bit sequence. In order to create a random bit sequence, an entropy distillation process is needed. The concepts behind this process to support the cryptographic primitives of *Heartwear* is described in Section 6.2.1. Thereafter, the implementation details of the synchronisation method will be presented. The synchronisation method serves the purpose of enabling two devices to measure a matching set of IPI values simultaneously in order to establish a shared secret and authenticate.

### 6.2.1. Entropy distillation process

In order to create a random bit sequence based upon the measurement of IPI values, an entropy distillation process needs to be implemented. While the quality of this process will be verified in the evaluation phase, the implementation can be described conceptually as follows:
For each IPI value represented as an 8-bit binary string, a predefined set of $m$ bits should be selected to form one segment of the shared secret. By concatenating the segments of each collected IPI value, the complete secret can be generated and will be used as symmetric key in the *Heartwear* protocol.

The key generated based upon one ECG measurement, requires to be an exact match with the key generated from an independent simultaneous ECG measurement in order to ensure two devices are able to establish the same key. However, as for any sensor, it is not uncommon to experience some *inter sensor variability* among independent measurements. The variability is reflected as the difference between two bit sequences from two simultaneous measurements. One technique to strengthen the bit sequence against the *inter sensor variability*, is to convert the binary sequence into Gray coding [20]. Gray coding is characterised as a binary representation of which two successive values only differ in one bit. To illustrate the benefits of Grey coding, consider the following example: suppose the external device has measured $IPI_1^E = 71$ samples, and the IMD concludes $IPI_1^I = 72$ samples large. The binary representation of $IPI_1^E$ =`01000111`, whereas $IPI_1^I$ =`01001000`. While the Hamming distance between 71 and 72 is only one, in binary format the hamming distance has increased to 4. By representing the IPI values in Grey coding where $IPI_1^E$ =`01100100`, and $IPI_1^I$ =`01101100` the Hamming distance is reduced to 1. Thus, applying Grey coding to the IPI values reduces the impact of measurement errors to the bit sequences.

### 6.2.2. Java implementation

The success of a legit external device to connect with an IMD when following the *Heartwear* protocol, also require two devices to synchronise their measurements as described in Section 5.3. Therefore, as part of the Proof of Concept, the subroutine has been implemented and executed in Java 8 on the same system as described in the introduction of this chapter and is included in Appendix A. While an external programmer may not be compatible with Java implementations, Appendix A only serves as an example how Algorithm 1 can be implemented but is not restricted to any language. The complete implementation consists of one main method, two functional methods and 5 supporting methods to simulate the environment.

The supporting method `generateBitString` feeds a bit sequence to the external programmer, representing legit measurements. Additionally, another method called `imdMeasures` simulates the behaviour of the IMD which starts measuring after receiving a connection request. The `imdMeasures` method takes a subset of the measurements (defined as `imdMeasurement`) from `generateBitString` after a random delay to simulate the delay in measuring. The subset will be hashed by the method `hashMeasurement`.

In the main method, the external programmer starts the execution of `findCommonSecret` wich resembles the process of the external programmer that takes the first `lengthCommonSecret` measurements starting from position 0 of the measurement to create his secret, hashes the concatenation and compares it against the received measurement from the IMD. If the values are unequal, the method will loop again but now starting from position 1 of the generated bit string. For this implementation specifically, the process is repeated until the common measurement has been found, or if the for-loop has iterated through the entire measurment. The amount of times the algorithm iterates trough the for-loop however, is not restricted to any value and is preferably connected to the time it takes the external device to receive the hashed value from the IMD.

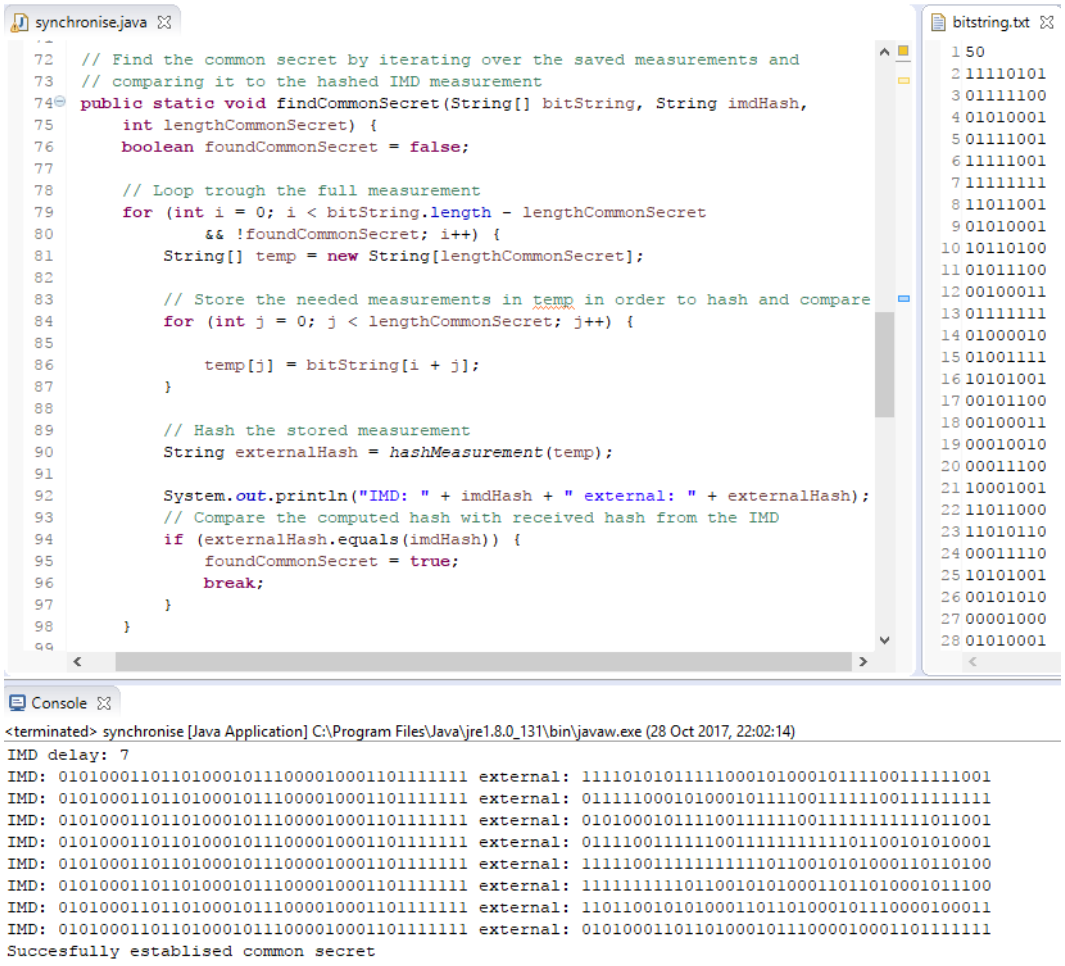



Figure 6.4: An example of the synchronisation method successfully synchronising the IMD and external programmer. On the right a segment of the bitstring, on the bottom the output of the execution.

Figure 6.4 contains one segment of the implementation and the result of one successful execution where the secret consists of 5 concatenated 8 bit IPI values (hence the total length of one secret in this example is 40 bits). The file bitstring.txt represents the complete measurement performed by the external programmer. The output of the execution is represented in the bottom frame of the figure. The first line describes the amount of delay (in IPI measurements) the IMD experienced before measuring its secret. The following 7 lines provide the complete measurement of the IMD and the tried measurement collections by the IMD before establishing the same secret.

## 6.3. Experimental Results

The *Heartwear* security scheme relies on the strength of entropy of the collected bits from IPI measurements and the error rate of two simultaneous measurements. The experiments executed in this section need to identify the quality of the entropy distillation process in general based upon the execution of statistical tests and Shannons entropy calculation [44]. To simulate the bit error rate (BER), or the inter sensor variability, for each bit, two independent simultaneous recordings are needed. Several studies [42] [38] [50] have replicated the scenario by calculating the IPI values from two different channels of the ECG records retrieved from the *MIT-BIH Normal Sinus Rythm* and *MIT-BIH Arrythmia* databases. Our experiments extend the calculation of the BER by applying the same approach to the high-resolution recordings from the *PTB* database.

### 6.3.1. Entropy

After representing an IPI value as an 8-bit binary sequence, the entropy $H(X)$ for each bit (modelled as a discrete random variable) of the binary string can be calculated by using the definition of Shannon's Entropy [44]: $H(X) = -\Sigma_{i=1}^{n} P(x_i) log_2 P(x_i)$ where $P(x_1)$ represents the probability of the bit being a 1 and $P(x_2)$ the probability of the same bit being a 0, note that $P(x_1) + P(x_2) = 1$. The bit entropy can attain any number between $(0, 1)$, of which 1 is the maximum entropy value, and 0 the minimum.

The first experiment calculates the entropy for each bit in the 8-bit IPI binary representation for each dataset. According to the definition of Shannon's entropy, we need to identify the probability $P(x_1)$ and $P(x_2)$ for a given bit sequence. As this experiment serves the purpose to identify the entropic bits of an 8-bit IPI value, the bit sequence is represented as the concatenation of one bit position for all IPI values. To illustrate, suppose a dataset contains $n$ IPI's, the amount of 1's and 0's occurring in all IPI values on the LSB position are summed up, such that $\#1 + \#0 = n$, then $P(x_1) = \frac{\#1}{n}$ and $P(x_2) = \frac{\#0}{n}$. These probabilities allows us to calculate the entropy for the LSB of an 8-bit IPI value. This calculation has been performed for each bit position and calculated for dataset separately. The result of this experiment is presented in Figure 6.5.

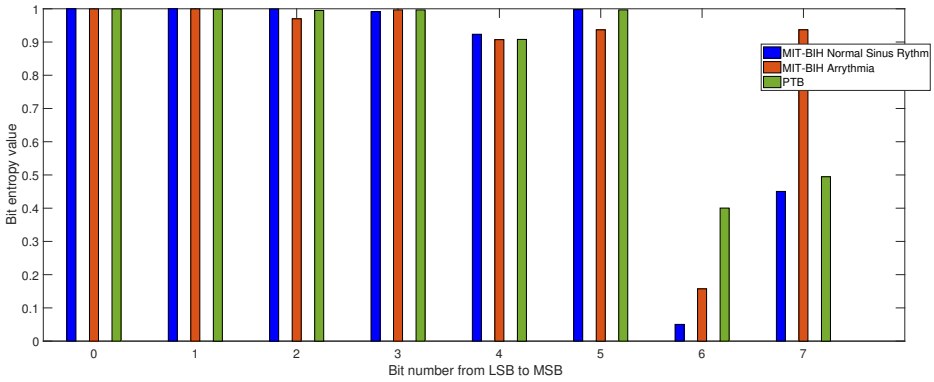



Figure 6.5: Bit entropy for each database from LSB to MSB

The result of a small variation on this experiment is given in Table 6.1. For this experiment the entropy of each bit is calculated by taking the IPI results into account of all datasets together. Each dataset contains the first 10 ECG signals of 10 different patients to calculate the entropy, resulting in 30 ECGs which contain a total of 120.976 IPI values. From the MSB to the LSB, Table 6.1 shows that bit 5 to 0 have an entropy value greater than 0.92.

Table 6.1: The average entropy of all IPI values contained in the first 10 ECG's from the *MIT-BIH Normal Sinus Rythm*, *MIT-BIH Arrythmia* and *PTB* dataset.

| Bit: | 7 (MSB) | 6 | 5 | 4 | 3 | 2 | 1 | 0 (LSB) |
|---|---|---|---|---|---|---|---|---|
| Entropy: | 0.60 | 0.09 | 1.0 | 0.93 | 1.0 | 1.0 | 1.0 | 1.0 |

### 6.3.2. Bit error rate

The bit error rate (BER) represents the difference between the bits of the same IPI measurement in a simultaneous reading of two devices caused by an error in measurement. The error rate depends on the interference of other electronic devices, body movement or other artefacts measured by a sensor. A high bit error rate however, causes the two independent key extraction phases to create an unequal secret which leads to an unsuccessful protocol run of *Heartwear* and is considered a false negative. To minimise the false negatives of denying authentication, it is therefore important to investigate the significance of the BER.

**H2H authentication** [38] defines the BER as the probability, for a given bit of an IPI, that two devices read the same IPI at different locations but output differing bit values. The work estimates the BER by means of calculating the IPI values of two external ECG leads since they lack the ability to obtain IPI measurements from IMDs. Our experiment follows a similar approach, where the BER is calculated in Matlab by extracting IPI values of two channels for each ECG measurement. Assuming both channels detect the same amount of IPI's, the presumably identical collection IPI's are compared by computing the Hamming distance for each bit position.

Table 6.2: The binary representation of four simultaneous IPI recordings obtained by device **I** and **E**, represented as grey encoded binary string and decimal value.

| | I | | | | | | | | E | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Decimal | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 (LSB) | Decimal | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 (LSB) |
| $IPI_1$ | 181 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 182 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 |
| $IPI_2$ | 181 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 181 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 |
| $IPI_3$ | 179 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 178 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 |
| $IPI_4$ | 184 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 184 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 |

As an example, given the values represented in Table 6.2, the hamming distance for the LSB sequence of $I$ and $E$ is determined by comparing the amount of bit difference thus: $1100_{LSB_I}$ and $1110_{LSB_E}$ have a hamming distance of 1. The BER of the LSB for this example is therefore: $BER_{LSB} = \frac{HammingDist}{SequenceLength} = \frac{1}{4}$. This approach will be performed for the other bit positions as well, for each complete dataset.

To compare the BER for each dataset, the average BER of all 10 ECG recordings per dataset is computed by following the previously described method. For *MIT-BIH Normal Sinus Rythm* and *MIT-BIH Arrythmia* the only 2 channels available (of lead (I,II) and (II,V1) accordingly) are used to perform the peak detection on. For the *PTB* dataset, 2 out of 19 channels may be selected, for the first experiment the channels of lead I and V4 are acquired. The average BER for each dataset is represented in Table 6.3 and shows a significant discrepancy between the BER of the *PTB*, *MIT-BIH Normal Sinus Rythm* and *MIT-BIH Arrythmia* dataset.

Table 6.3: The average BER of each bit for every dataset.

| Dataset | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 (LSB) |
|---|---|---|---|---|---|---|---|---|
| *PTB* | 0.010 | 0.001 | 0.002 | 0.013 | 0.034 | 0.046 | 0.056 | 0.093 |
| *MIT-BIH Normal Sinus Rythm* | 0.064 | 0.022 | 0.130 | 0.232 | 0.279 | 0.343 | 0.389 | 0.416 |
| *MIT-BIH Arrythmia* | 0.137 | 0.034 | 0.128 | 0.170 | 0.375 | 0.380 | 0.420 | 0.433 |

Upon closer inspection of the peak detection method of the *MIT-BIH Normal Sinus Rythm* and *MIT-BIH Arrythmia* database, it is observed that the majority of errors are caused by an incorrect R-peak classification and heartbeat misdetections which are reflected in the BER. Figure 6.6 shows a segment of the peak detection performed on one of the ECG signals in the *MIT-BIH Normal Sinus Rythm* dataset, where the peak misdetection is visible of Lead II.

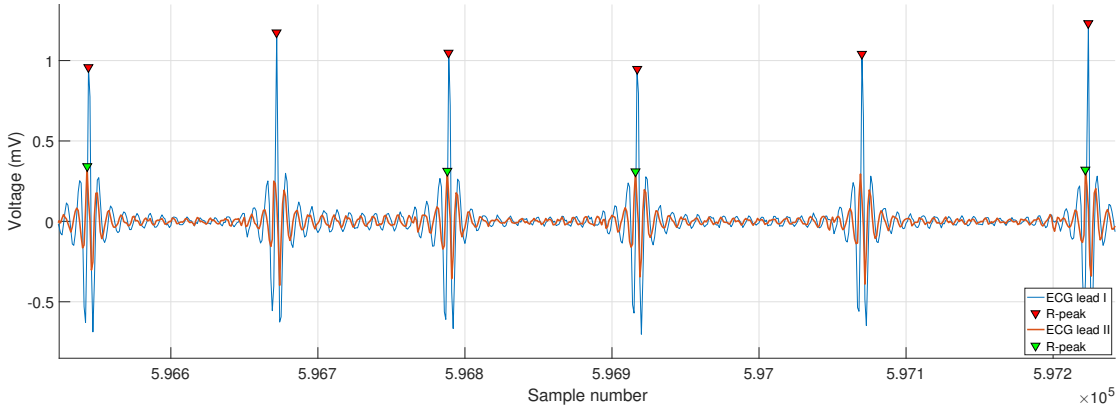



Figure 6.6: The peak detection of one ECG from the *MIT-BIH Normal Sinus Rythm* dataset

The average BER of the *PTB* dataset however, appears to be much lower. By inspecting the the performance of the peak detection method, it is established that there occur no misdetections and the location of the peak detected for both channels are aligned more accurately. An example of this observation is given in Figure 6.7. By examining the alignment of the ECG signals further, it can also be concluded that the signals of the *PTB* dataset appear to be much more similar among each other compared to the signals of the *MIT-BIH Normal Sinus Rythm* of Figures 6.7 and 6.6 respectively.

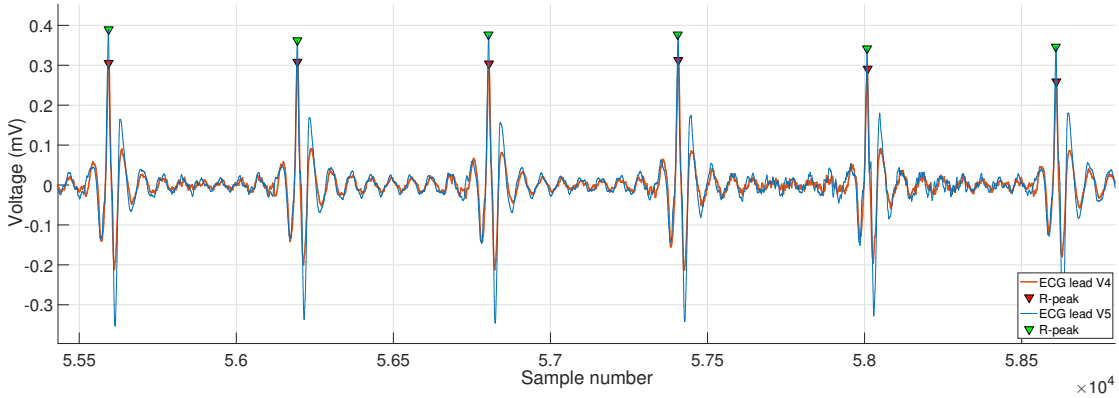



Figure 6.7: The peak detection of one ECG from the *PTB* dataset.

While strong entropic bits are required to generate a truly random bit-sequence, the reliability of the *Heartwear* security scheme also depends on the BER of two independent signals. The performed experiments support the evaluation of *Heartwear*, and allow the elimination of certain bits of an IPI binary sequence to avoid security or usability issues. As will be discussed more thoroughly in the next chapter, the trade-off between security and utility of the security scheme therefore depends on the BER and entropy as calculated during the experimental phase. 7.

# 7

# Evaluation of *Heartwear*

*The performance of the Heartwear protocol will be evaluated from a security, safety and usability perspective. This requires a theoretical proof of correctness of the synchronisation method, a security analysis and the estimation of the cryptographic overhead of Heartwear. Based upon the proof-of-concept implementation and the experiments executed for this study, the benefits and limitations of the entropy distillation process can be identified to asses the feasibility of our proposed security scheme.*

This chapter is organised as follows: First a formal proof to demonstrate the correctness of the *Synchronisation* routine is given in Section 7.1. Subsequently, the security goals are verified by means of a security analysis provided in Section 7.2, including an inspection of the quality of the source of entropy as well as a verification of the attack strategy. The chapter is concluded by comparing the cryptographic overhead with other solutions in Section 7.3.

## 7.1. Correctness synchronisation method

The correctness of the protocol to synchronise two devices such that the same set of $n$ IPI values are collected relies on the following concept and maintains similar notation as given in Table 5.1: Since the external device initiates the protocol, the external device starts measuring IPI values for $W$ at least at the same time as the IMD will do. According to Algorithm 1, the external device stops collecting IPI values until it receives $h(w_i)$ on time $t_i$. By executing the complete synchronisation method, the external device should be able to find a $w_E \in W$ such that $w_I == w_E$, if and only if witness $w_i \in W$. The following proof shows that the external programmer will be guaranteed $w_i \in W$, assuming both devices measure without any error.

Given the external device $E$ starts collecting $m$ IPI values for witness $W$ starting on time $t_0$, and the IMD collects $n$ IPI values to construct $w_I$ starting on time $t_d$ s.t. $t_0 \leq t_d$ and it takes $t_n$ time for the IMD to collect $n$ IPI values. $E$ stops filling $W$ after it receives the $h(w_I)$ from the IMD. This requires $t_n + t_\epsilon$ time where $t_\epsilon$ represents the transmission time to send $h(w_I)$ to $E$. Now $E$ has collected $m$ IPI values over a period of $t_m$ time, where $t_m = t_d + t_n + t_\epsilon$. With $0 \leq t_d, t_\epsilon$, it holds that $t_n \leq t_m$.
Since both devices are assumed to measure IPI values without error, and the IMD measures $w_I$ within the time span of the IPI collection $W$, we can conclude that $w_i \in W$.

## 7.2. Security analysis

The goal of the adversary as defined in Section 4.3, is to program or retrieve sensitive data from the IMD trough the wireless communication channel without being caught. *Heartwear* has been designed to protect the IMD from an active adversary with this goal, who has full control of the channel. Protecting an IMD in this landscape, requires *Heartwear* to create a shared secret from IPI values to provide both authentication as well as key establishment. The feasibility of establishing a shared secret however, depends on the randomness of IPI values and the reliability of the measurements expressed as BER. The security analysis will therefore first cover the strength of the source of entropy with respect to

its unpredictability and reliability. Thereafter, the attack strategy of the adversary is followed to verify if *Heartwear* is capable of protecting the IMD from the defined threat model.

### 7.2.1. Validation of the source of entropy

The generation of random numbers in the *Heartwear* security scheme is achieved by utilising certain characteristics from ECG signals, a time varying waveform which is considered to be a non-deterministic source [1]. To produce the desired randomness, IPI values are obtained from the ECG signals and converted to an 8-bit binary string. According to the experiments performed in Section 6.3, each bit-position of the binary string delivers a certain entropy value. Upon inspection of these results, it is established that the most significant bits of an 8-bit IPI value contain typically less entropy compared to the least significant bits. Overall however, the average bit entropy by applying Shannon's entropy [44] and as presented in Table 6.1 looks promising. From LSB to MSB, bit $0-5$ are considered to provide a sufficient level of entropy, despite the slightly lower value of the $4^{th}$ bit. To substantiate this decision, a segment of the bits have been extracted and applied to the NIST suite of statistical tests [39].

The NIST statistical test suite contains a set of statistical tests to calculate the probability a bit-sequence has been generated by a random process in terms of a $p$ value. If any of the tests result in a value of $p \leq 0.01$, the hypothesis stating the bit-sequence is random should be rejected. In case of the verification whether the $0-5^{th}$ bits are suitable to use as a random bit-sequence, a concatenation of IPI values containing only the $0-5^{th}$ bits are fed to the NIST statistical test suite, of the PTB dataset only. The result of the tests is given in Table 7.1 where $p$ represents the probability a perfect random number generator would have produced a sequence less random than the tested sequence. The tests performed are similar as executed by Rostami *et al.* [38], our work however does test the linear complexity as this test was made to identify the randomness of pseudo random number generators. Since none of the $p$ values are smaller than 0.01, the hypothesis that the bit-sequence is random can be accepted. Therefore, it is concluded that selecting $0-5^{th}$ bits of an 8-bit IPI value are capable of delivering a randomly generated bit sequence.

Table 7.1: P-values of the NIST statitsical test suite for $0-5^{th}$ bit of every IPI in the PTB dataset.

| NIST Test | p-value |
|---|---|
| Frequency | 0.021 |
| Runs | 0.462 |
| Longest runs | 0.280 |
| FFT | 0.184 |
| Universal | 0.061 |
| Approximate entropy | 0.303 |

Directly applying the generated bit-sequences to *Heartwear* however, may unnecessarily refrain legit external devices from connecting with the IMD due to the BER. To avoid such false negatives, it is essential to investigate the possible impact of the bit error rate. To simulate two simultaneous recordings between two devices, the IPI extraction method has been applied to every ECG record of two simultaneous lead measurements. As depicted in Table 6.3, the BER varies significantly among the different datasets. The high BER of the *MIT-BIH Normal Sinus Rythm* and *MIT-BIH Arrythmia* can be related to the placement of the electrodes: As explained by the author of the *MIT-BIH Arrythmia* database [33], normal heartbeats are difficult to discern in the second channel of the ECG record due to the lead configuration. The second channel of the *MIT-BIH Arrythmia* database is obtained trough lead VI, a signal where small disturbances (ectopic beats) of the heart rate are prominently present in the ECG record while *R-peaks* remain hidden as they entail a much lower deflection. This makes the peak detection method nearly impossible since the peaks occurring in the record may not represent the *R-peak*. With similar reasoning the high BER of the *MIT-BIH Normal Sinus Rythm* can be explained. For this dataset the second channel is obtained from lead II, which appears to produce very small *R-peaks*, as shown in Figure 7.1.

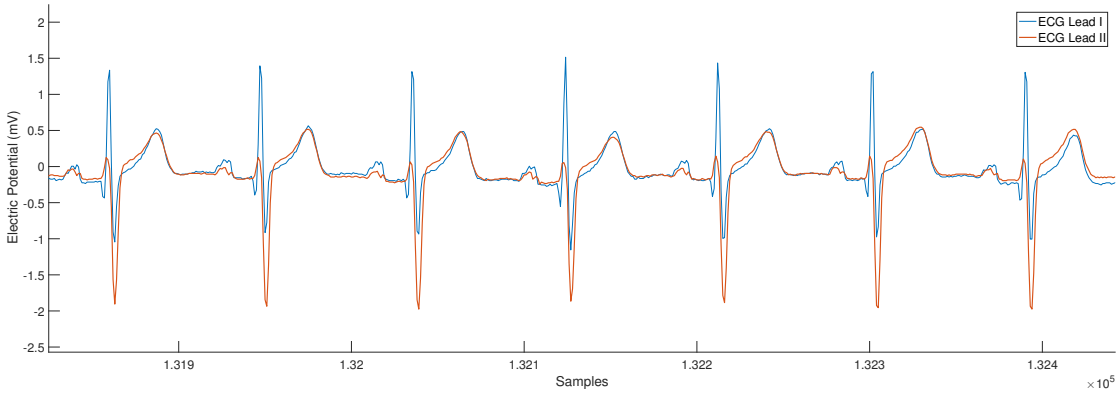



Figure 7.1: A segment of the two channels from the ECG record in the *MIT-BIH Normal Sinus Rythm* dataset

A more representative estimation of the BER has been extracted from the *PTB* dataset of which first the importance of the lead configuration can be verified since this dataset contains the collection of 12-lead ECG measurements. By selecting different lead configurations, it is observed the BER varies significantly. As an example, Figure 7.2 represents the peak detection results when using lead configurations II and lead V1 (similar to the *MIT-BIH Arrythmia* dataset), whereas Figure 7.3 shows the result of applying the same method by using the signals obtained from lead V4 and V5.

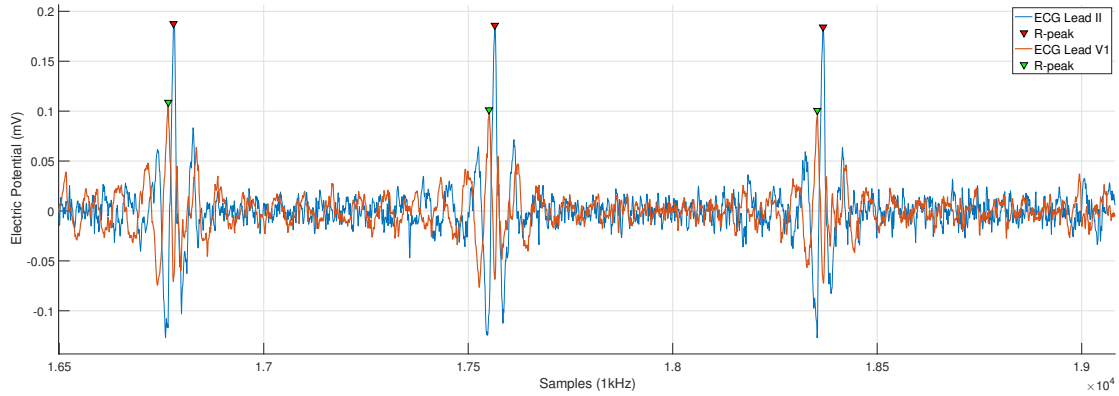



Figure 7.2: Peak detection of an ECG record for lead II and lead V1 of the *PTB* dataset.

Although the results of Figure 7.2 do not reveal a visual impact as extreme as with the *MIT-BIH Arrythmia* and *MIT-BIH Normal Sinus Rythm* database, there is however a slight shift in peak detection visible due to the shape of the *R-peak* deflection of ECG lead II. These results indicate the importance of the ECG lead configuration. Therefore, it is concluded that more careful lead configuration should be applied to optimise the BER and entropy value.

The reason for misdetections in the *MIT-BIH Arrythmia* and *BIH Normal Sinus Rythm* databases can be explained by the lead configurations, and according to Seepers *et al.* [43], also depends on the sampling frequency. A lower sampling frequency often creates a lower BER (due to a reduction of misdetections) but also reduces the entropy of the binary sequence. Our observations additionally find a strong relationship in the lead configuration. By selecting the leads of an ECG signal more carefully, the BER improves significantly without influencing the entropy value. The improved results of the BER and entropy are depicted in Table 7.2 and are based on lead configurations V4 and V5.

To extract a strong key from specific bits retrieved from the IPI binary representations, a minimum recommended key strength needs to be achieved. Since 2014, the minimum recommended key strength as defined by NIST [3] has been defined to be at least 112 bits. As established earlier, bits 0 to 5

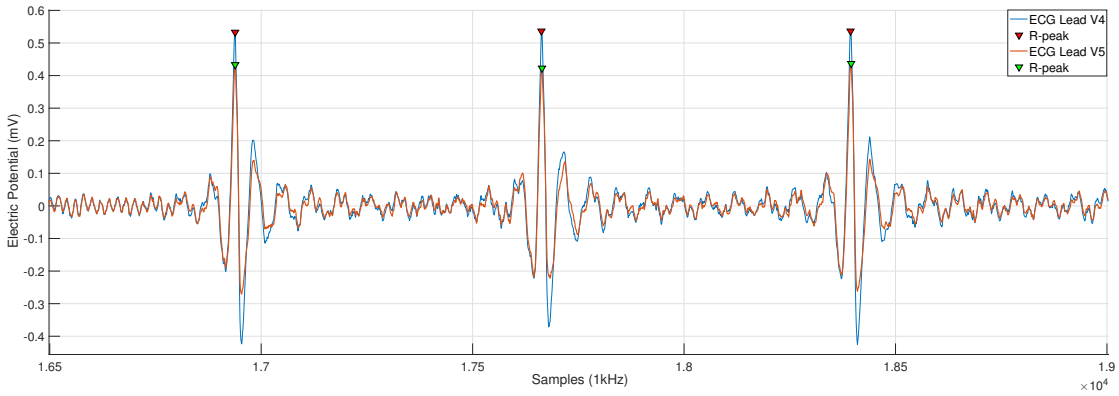



Figure 7.3: Peak detection of the same ECG record with lead configuration V4 and V5 of the *PTB* dataset.

Table 7.2: The entropy and BER values after improving the lead configuration to V4 and V5

| Bit: | 7(MSB) | 6 | 5 | 4 | 3 | 2 | 1 | 0(LSB) |
|---|---|---|---|---|---|---|---|---|
| Entropy: | 0.50 | 0.38 | 0.99 | 0.99 | 1 | 1 | 1 | 1 |
| BER | 0.004 | 0.001 | 0.0034 | 0.0022 | 0.0073 | 0.0184 | 0.0388 | 0.054 |

may be included in the creation of a random bit string. Which bits will actually be selected, also depends on the desired maximum false negative rate and the time one round of *Heartwear* should take. To illustrate, suppose it is decided to extract bit 5 and 4 to create a 112 long bit-string, which requires 56 IPI measurements. The BER represents the probability that two devices read the same IPI but output a different value for each bit. The BER is according to the results in Table 7.2, equal to $0.0034 + 0.0022 = 0.0056$. Therefore, the true positive rate (or the probability the extraction of one complete key does not contain any measurement errors) can be calculated by simulating the probability of a binomial distribution $Bin(n, p)$, where $n$ represents the amount of measurements needed to create a 112-bit key, and $p$ the probability a measurement does not contain a measurement error, thus $Bin(56, 1-0.0056)$. The probability of zero measurement errors over 56 measurements is then: $P(X = 0) = \binom{56}{0}(1-0.0056)^{56-0} = 0.9954^{56} \approx 73\%$. A similar approach can be taken to calculate the probability of zero measurement errors if only the $4^{th}$ bit is extracted such that $Bin(112, 1-0.0022)$. This results in a probability of approximately 78%, while it requires the measurement of 112 consecutive IPI values instead of 56 in the previous case. Depending on the performance requirements of the medical device manufacturers, and the BER of specific IMD models, different bits may be selected in order to maintain a high true positive rate while decreasing the total amount of IPI measurements.

### 7.2.2. Verification attack strategy

*Heartwear* provides an authentication and key establishment scheme to protect an IMD against the advanced adversary model as described in Section 4.5. According to the attack strategy, the adversary may attempt to either impersonate an external device, perform off-line secret guessing or re-use information in order to gain access of retrieve privacy sensitive information from the IMD. The feasibility of successfully following one of the strategies is investigated separately supported by a suitable scenario. Although there exist many scenarios to defend against, the threat model for *Heartwear* has been designed to alleviate the current privacy and security concerns. Since these concerns have their foundation in recent attacks demonstrated by [32] [21] [21] and the common attacks on wireless communication channels as described in Section 2.1.1, the scenarios will be based on similar settings.

#### Impersonation of external device

With the assumption that any external device may be considered legitimate if it can gain physical contact to the patient, the adversary can only attempt to impersonate an external device *without* physical contact. Suppose an adversary sends an authentication request to the IMD, then he needs to convince the IMD it is a legitimate programmer by solving the challenge as part of the *challenge response* protocol correctly. This leaves the following scenarios open:

- *Case 1:* The adversary attempts to brute force the shared secret of 112 bit.
- *Case 2:* The adversary targets the IMD with a remote attack.
- *Case 3:* The adversary tries to recompute $w_I$ from the hash.

For *Case 1*, the adversary could approach the impersonation by guessing the shared secret established by the IMD, similarly to the attack performed by Marin *et al.* [32]. Since a hash of the secret $w_I$ is sent by the IMD, the attacker can verify its guessed $w_A$ by hashing his value and match it against the received hashed value. Given the secret is (at least) 112 bits strong, extracted from an unpredictable sequence generator, then the best attempt for an adversary is to try all $2^{112}$ possibilities by means of a brute-force approach. This approach remains however unfeasible as the adversary has a probability of $p = 0.5$ to guess one bit position correctly. With no ability to verify how many bits are (in)correct, the adversary needs to guess the complete sequence of 112 bits such that $h(w_I) = h(w_A)$ in order to solve the challenge correctly at once. Suppose a system is capable of calculating $10^6$ keys per $\mu s$, exhaustive key search will take on average $5.1922 \times 10^{27} \mu s$, or $164 \times 10^{12}$ years for an attacker to guess the secret correctly.

In *Case 2*, the attacker may attempt to gain an advantage by estimating IPI values from a distance. Successfully detecting heartbeats without physical access has been studied by Calleja *et al.* [6] who have been able to detect $IPI$ values with an accuracy of 75% by reading images from a camera placed 50cm from a patients face. A valid scenario for such an attack would be where an adversary has compromised the webcam of the patients laptop or the camera of his phone to complete the protocol by guessing IPI values based on the estimation of blood pressure by performing sophisticated image processing techniques. Seepers *et al.* [43] investigated the attack and concluded that by remotely monitoring certain PPG (rPPG) values can be obtained with similar accuracy as contact PPG (cPPG) values. For ECG to ECG measurements however, the researchers have been unable to obtain an accurate representation remotely. Seepers also substantiate the likelihood of accurately estimating ECG values by rPPG measurements in the future. Their results suggest that the technique require highly pulsatile body-parts to be exposed during the authentication process. It also requires very stable luminance and movement conditions, making it unlikely to successfully launch a remote attack using rPPG to estimate $IPI$ values based on ECG measurements.

*Case 3* describes the effort of an adversary who tries to gain knowledge from the hashed witness $h(w_I)$ received from the IMD to increase the likelihood of guessing the 112 secret bit sequence correctly. Gaining knowledge from an ideal hash function however would violate the basic hash property of pre-image resistance. The hash function used in *Heartwear* is considered to be an ideal one-way function, making it therefore unfeasible for an attacker to gain any knowledge from $h(w_I)$ to guess the secret with a higher probability.

#### Off-line secret guessing

Off-line secret guessing describes the attempt of an adversary to retrieve any information of previously recorded sessions which may be valuable after the session has been closed. In most cases of off-line secret guessing attacks such as demonstrated by Marin *et al.* [32], the adversary will try to gain any knowledge of a (pre-established) key such that he can decrypt any future encrypted message. *Heartwear* however, ensures the *freshness* property for every new session because each session key is generated based upon the time varying properties of the ECG wavelet. Therefore, the adversary can only attempt to decrypt the collected information by actively guess the key. Although an attacker may have more time during an off-line secret guessing attack, the average of $164 \times 10^{12}$ years to successfully perform an exhaustive key search attack remains, and is still considered to be unfeasible. For this scenario it is therefore concluded that *Heartwear* is capable of keeping the privacy sensitive information secret for at least the minimal requirement of 10 years as described in Section 4.4.2.

#### Information re-use

When the re-use of old information is useful for the adversary, he will be interested in eavesdropping on the communication to execute a variety of attacks in a future session such as replay and MitM attacks. The considered scenarios are based upon the replay attacks performed by various researchers

[21] [37] [32] and the theoretically feasible attack on "H2H authentication" as described in Section 2.2.2. The cases are distinguished as follows:

- *Case 1:* The adversary performs a man-in-the-middle attack.
- *Case 2:* The adversary re-uses the information of another protocol run.

*Case 1* sets the scenario of a MiTM attack, similar to the case as described in Section 2.2.2 where the "H2H authentication" [38] protocol was found to be vulnerable for. The adversary will attempt to break the challenge-response protocol by capturing the challenge, replay it to the external device, wait for its response and sends it back to the IMD. This attempt would be of our concern if the messages sent over the communication channel reveal any sensitive information. Capturing and replaying the responses in this case however, will not provide additional information compared to an ordinary eavesdropping attempt. As established earlier, eavesdropping will not provide any significant information to the adversary as he still requires the knowledge of the shared secret to be meaningful which is not the case as each session establishes a new and fresh key.

Re-using information as for *Case 2* can be successful when the adversary has extracted the key of an older protocol run and is able to use it for another run as sucessfully executed by Halperin *et al.* [21] and Radcliffe [37]. As for any replay-attack approach, the technique may only be possibly successful if the security scheme would re-use old keys. However, since the entropy distillation process guarantees the creation of a fresh and strong key, this attack vector will not gain the adversary with illegitimate access.

## 7.3. Cryptographic overhead

The *Heartwear* protocol has been designed to reduce the energy overheads of the cryptographic methods with respect to other approaches such as [42] [38] [50]. One aspect is the amount of time it takes to authenticate two devices. While the formal description of the protocol does not seem to require much time in order to complete one run, a large amount of time is spent to generate the shared secret. Since the key should have a strength of at least 112 bits, and we can reliably extract 2 random bits per IPI given our current experimental results, the average to create a 112 bit sequence requires the measurement of at least $\frac{112}{2} = 56$ IPI values. Since the contraction of the heart occurs on average once per second, one IPI value appears every second. When 56 consecutive IPI values are required to create the shared key, the authentication protocol will take approximately 56 seconds before two devices are authenticated. While this time may seem a long time, Seepers *et al.* [42] are satisfied with their key establishment protocol to run on average between 60 and 77.6 seconds as a proof of concept.

The major factor of energy consumption when applying this protocol can be expressed as the cost of transmitting messages over a wireless communication channel. An estimation of these costs can be illustrated by assuming the `ZL70101 Medical Implantable RF Transceiver`[1] is used to transmit and receive messages by a medical device. This is a realistic scenario because this specific transceiver has especially been designed for IMDs and operates on the FDA approved MICS band. The energy consumption per bit ($J/b$) can be calculated by summing the current $I$ (measured in $A$) consumed when the transceiver is in send (TX) and receive (RX) mode. Since the power equals voltage times current, the power (in $W$) is calculated by multiplying the amount of voltage $U$ fed to the transceiver and equals $U = 2V$. After multiplying $P = 2 \times (0.0005 + 0.0005) = 2mW$, hence the energy transferred per second equals $2mJ$.
Since the transceiver can handle $267kbps$ when in `2FSK High Rate` mode, the energy consumption per bit for transmitting or sending costs: $E_n = \frac{0.002}{267.000} = 5nJ/b$. To put this in perspective, the value is compared to the energy consumption of cryptographic operations implemented on hardware. According to the research done by Patrick *et al.* [36] and published by NIST, the most expensive block cipher (AES) was concluded to cost $289,5pJ/b$ or $0.289nJ$. This means that transmitting and receiving one bit costs on average $\frac{5}{0.289} \approx 17$ times more energy compared to the most expensive cryptographic primitive. This makes the energy consumption for each transferred bit therefore more significant compared to the energy consumption of each bit for the heaviest cryptographic primitive.

[1] `https://www.microsemi.com/document-portal/doc_view/127877-zl70101-1645-19-fullds`, Retrieved Oktober 2017

# 8

# Conclusion

*The introduction of wireless communication capabilities to IMDs allow the health care practitioner to tailor the delivered therapy to the patient over time in order to increase the treatment effectiveness. An IMD however is a crucial component for patients who rely with their lives on the IMDs functionality, which may be threatened by exploiting previously identified vulnerabilities due to the insufficiently protected wireless communication channel between an IMD and external device. Protecting the wireless communication channel however, requires a trade-off between the safety, security and utility aspects of the IMD.*

Several researchers have put their efforts into proposing a security scheme by means of four different approaches. Despite the promising results of previous work utilising heartbeats in a protocol to overcome the vulnerabilities of a wireless communication channel, none however have proposed a security scheme where heartbeat based key establishment and authentication have been combined into one protocol. In this research, the feasibility of incorporating a security scheme to balance security, safety and utility has been demonstrated by focusing on the following research question:

*How can an IMD establish a secure wireless communication channel with a legit external device, such that an attacker is prevented from gathering or changing privacy sensitive information, without diminishing the adequacy of the treatment delivered by an IMD.*

In this chapter, we return to the research question and discuss how *Heartwear* achieves this goal. Moreover, the chapter provides future research directions by identifying remaining open problems and improvements.

## 8.1. Overview of addressed challenges by the proposal of *Heartwear*

In this thesis, a security scheme called *Heartwear* has been proposed by combining authentication and key establishment into one efficient protocol based upon a random bit sequence generated by the activity of the heart. The protocol makes use of a set of cryptographic primitives while distilling entropic bits from IPI values, extracted from the heart. The IPI values have been obtained by performing a peak detection method applied to the processed ECG signals, transformed into an 8-bit binary sequence of which certain bits can be selected and fed to the *Heartwear* protocol as a random bit sequence generator in order to provide the desired level of security with low impact on the utility goals of the IMD.

### 8.1.1. Verification of challenges tackled by Heartwear

Contrary to some of the prior art described in Chapter 3, *Heartwear* is the first to combine key establishment as well as authentication into one security scheme while utilising IPI values. Many of the related work has not investigated the value of creating a (pseudo) random number generator with the functionality already embedded in an IMD. Work that investigated the possibilities of extracting entropy from heartbeats however, did not combine authentication and key establishment in one security scheme. Instead, either key establishment or authentication has been provided by this method. As a

consequence, other work relies on the generation of a (pseudo) random bit sequence for the cryptographic primitives from a hardware implementation level.

One of the first challenges addressed to approach this problem is to gain an understanding of (lightweight) authentication and key establishment protocols, their application and the tensions of the landscape to protect an IMD against an adversary. This provided insights to the current difficulties experienced when proposing a solution as an answer to the research question. Additionally, a comparison is performed among related studies with the aim to protect the IMD from the introduced vulnerabilities of the IMDs enabled with wireless communication. The related work has been classified and compared, after which the biometric approach was chosen as main component to satisfy our security goal. After identification of the problem and tensions that occur within the restricted environment an IMD operates in, the IMD specific goals have been designed which have been derived by first describing the application setting of the problem after which more formal requirements can be extracted from the threat model. Thereafter the fifth challenge is overcome by designing a protocol based upon the requirements as established previously in the document. By assessing the cryptographic preliminaries, a synchronisation method is proposed as well as the complete *Heartwear* security scheme. The protocol was implemented and verified in the following chapter by validating whether the previously established requirements have been satisfied as well analysing the experimental results conducted after the implementation phase.

### 8.1.2. Performance of Heartwear compared to prior art

Our result presents *Heartwear*, a security scheme to protect an IMD against an advanced adversary without the need of pre-established knowledge and independent from model specific hardware architectures. Our solution outperforms the prior art in in terms of reliability and efficiency in terms of communication efficiency, speed, possible key strength, efficient use of the source of entropy and mitigation techniques.

*Heartwear* outperforms the **Secure key exchange** protocol of Seepers *et al.* [42] in terms of communication efficiency since a complete protocol run of **Secure key exchange** requires a total of 6 transmissions from initialisation until the key confirmation whereas *Heartwear* only require 4 which includes the initialisation step, a complete challenge response and key confirmation phase. **Secure Key exchange** also does not create a shared key based upon the measurement of IPI values, but only apply the heartbeat signals to support a fuzzy commitment scheme. The complete protocol can handle at most an 80-bit secret and takes on average $60 - 77,6$ seconds to execute. *Heartwear* completes the protocol in about 58 seconds based on the recommended key strength of 112 bits.

**IMDGuard**, proposed by Xu *et al.* [50], does not employ IPI values in a fuzzy commitment scheme but as a source to establish a common secret between a proxy device and IMD. Their protocol requires a minimum of $43 \times 2$ transmissions for the key establishment only, in order to increase the probability both devices extracted the same secret of 128 bits. *Heartwear* significantly increases the communication efficiency with its maximum of 4 transmissions to complete the protocol. On average the **IMDGuard** however requires a total of 61 IPI's due to the amount of IPI's that are likely to be discarded, making the key establishment phase approximately as fast as *Heartwear* to complete authentication as well as key establishment.

**H2H Authentication** as presented by Rostami *et al.* [38] first requires the setup of a secure channel by means of the TLS protocol where the IMD performs the operations of an ordinary TLS client and the programmer those of an ordinary server. During the setup of the secure channel, a key is exchanged by following the RSA public key cryptosystem which is computationally more expensive to symmetric key implementations due to the modulo and factorisation computations. The authentication phase occurs in **H2H authentication** by means of following a commitment scheme based on the random binary sequence extracted from IPI values. The commitment scheme succeeds if the decommitment falls within a given threshold of the locally stored binary sequence. In contrast to what the authors of **H2H Authentication** claim however, the pairing protocol was observed to be vulnerable to a reflection and man-in-the-middle attack as described in Section 3.3.4. *Heartwear* is resistant to these attacks as it is based upon a challenge-response protocol without the need of sending valuable information over

the wireless communication channel.

Additionally, none of the three solutions have provided a synchronisation method, essential for all approaches that require two independent devices to measure simultaneously. *Heartwear* distinguishes itself by proposing a synchronisation method which does not add any cryptographic overhead to the IMD as the complete routine is implemented on the external device. The correctness of the synchronisation method has been verified by the PoC implementation presented in Section 6.2.2 and the formal proof in Section 7.1.

A main concern of *Heartwear* however, is the need for an exact match of certain bits of the extracted IPI values. The probability to reject a legitimate external device in *Heartwear* depends on BER and is identified by **H2H authentication** to be equal to 0.01 for each IPI when extracting bit 4 and 5 counting from the LSB. Seepers *et al.* [42] has also investigated the BER and concluded it is unfeasible to rely on exact IPI measurements. *Heartwear* however has been able to achieve a BER of 0.005 when extracting bit 4 and 5 from each IPI, which is half the BER as reported in [38]. This improvement is explained because it is argued the computation of the BER by performing *R-peak* detection to the *MIT-BIH Normal Sinus Rythm* and *MIT-BIH Arrythmia* database is not realistic due to the lead configuration of these ECG records. Due to the BER differences specific to the chosen lead configurations of 12-lead ECG recordings indicate that the BER highly depends on the lead configurations. Our protocol does not rely on the extraction of specific bits, and for the implementation of *Heartwear* one is free to choose any combination of bits from an IPI value as long as the key conditions will be met. This allows *Heartwear* to be optimised on either execution time or false negative rate, depending on the amount of entropic bits extracted from one IPI value.

## 8.2. Future work

*Heartwear* is, to the best of our knowledge, the first to alleviate the tensions between safety, security and utility by combining heartbeat based authentication and key establishment protocols in order to protect an IMD from prying eyes and active adversaries who wish to connect to the IMD over an insecure wireless communication channel. The protocol shows promising results in terms of achieved security strength, minimal cryptographic overheads and independence to specific IMD models. There exists however, substantial opportunity to improve the *Heartwear* security scheme.

### Collaboration with medical industry

First and foremost, the design and implementation of the *Heartwear* security protocol as well as other state of the art solutions [32] [42] [50] has not been performed in collaboration with the main stakeholder which is the Medical Industry. Although the design has been established by compliance with the GDPR and collaboration with law firms, doctors and patients, the medical industry has so far not been actively involved with the design. Since the solutions have only been verified within a simulated realistic setting, rather than a true environment, it is still an open question how the solutions will perform in a real-life scenario.

### Automated lead configuration and peak detection

The performance of *Heartwear* typically relies on the BER and accuracy of the peak detection method of the implementation. The BER has been observed to be related to the lead configuration, while a change in lead configuration also affects the accuracy of the peak detection method. To maintain the accuracy of the peak detection method for our experiments, the input parameters have been improved by means of a trial-and-error approach. A possible direction for future work is therefore to investigate automated methods in order to identify the model parameters faster and in a more reliable fashion.

### DoS prevention

During the design of the threat model, a strong adversary model has been considered who follows a pre-defined attack strategy. The attacks incorporated to the threat model included all common attacks on wireless communication channels as identified by Zou *et al.* [52] except from Denial of Service (DoS) attacks which may attempt to perform battery depletion or jamming attacks. A possible improvement to *Heartwear* is therefore include DoS prevention mechanisms into the security scheme without affecting the performance of the current proposal.

#### Possibilities in other domains

This thesis has been dedicated to propose a solution for the specific case of enabling secure communication for an IMD to connect over a wireless channel with an external device. The security scheme has been designed for a extremely resource constrained environment, but is not limited to other applications. We therefore invite other researchers to look into other applications in need of a lightweight security scheme which provides a authentication and key establishment mechanism by utilising the measurement of heartbeats.

## 8.3. Summary of *Heartwear* and contributions

The objective of this research was to enable an IMD and external programmer to communicate over a secure wireless communication channel to protect the IMD and the privacy sensitive information against strong adversaries. *Heartwear* alleviates the tensions between the safety, utility and security goals by proposing a security scheme with minimal cryptographic overheads which can be tailored to the performance requirements in terms of the false negative rate and execution time according to the designers wishes.

This thesis presents a technique to efficiently create a random number generator by utilising the time-varying characteristics of a heartbeat signal. By measuring the signal simultaneously, the complexity of sharing a secret over an unprotected communication channel has been overcome based on the assumption an external device is legitimate if it can get physical access to the patient and by the knowledge that heartbeats obtained from ECG records can only be read upon physical contact. It was observed that the reliability of *Heartwear* to extract a matching shared secret depends on the BER, which has been reduced by half for specific bits in the collected heartbeats compared to the results of related work. While other studies identified the BER and entropy of IPI values are characterised by the sampling frequency of the ECG record, we have been able to highlight that the BER also depends on specific lead configurations used to extract ECG signals in which the *R-peaks* may appear more prominent compared to other configurations.
Additionally *Heartwear* distinguishes itself from related work through the proposal of a novel technique to synchronise two devices since other studies who relied on the simultaneous measurement of heartbeats by two independent devices assumed a synchronisation method was already in place. Furthermore, *Heartwear* differs from the state of the art as it applies heartbeat security into one scheme for both authentication as well as key establishment unlike other studies who only achieved either one and therefore relied on additional (P)RNGs.

Lastly, *Heartwear* has to the best of our knowledge, the only security scheme which has been designed in accordance with the upcoming GDPR that will apply for all devices actively processing sensitive data in Europe on the 25th of May in 2018. Our security scheme is also able to comply with stronger key requirements since *Heartwear* is unrestricted to the amount of IPI values needed for key establishment. This enables IMDs that will be introduced to the market to be law compliant and ready for stricter demands in the future in terms of key strength.

The obtained results show that the establishment of a shared secret trough measurements of the human heart is possible. By improving the detection mechanism offline, just as for (fuzzy) commitment schemes who need to set a certain threshold, shows promising results in the reliability of applying *Heartwear*. We are proud to communicate that given these results, *Heartwear* provides an efficient and dynamic key establishment and authentication protocol with an improved accuracy of heartbeat based security protecting an IMD against impersonation, off-line secret guessing and the re-use of information.

# A

# Synchronisation Method

```
1  import java.io.File;
2  import java.io.FileNotFoundException;
3  import java.io.PrintWriter;
4  import java.util.Random;
5  import java.util.Scanner;
6
7  public class synchronise {
8
9    public static void main(String[] args) throws FileNotFoundException {
10     String fileName = "bitstring.txt";
11     int lengthBitString = 50;
12
13     // Generate bitString
14     String[] bitString = generateBitString(lengthBitString);
15     writeToFile(lengthBitString, bitString);
16
17     // IMD takes a measurement and hashes it
18     // lengthCommonSecret is the amount of IPIs the secret contains
19     int lengthCommonSecret = 5;
20     String[] imdMeasurement = imdMeasures(bitString, lengthCommonSecret);
21     String imdHash = hashMeasurement(imdMeasurement);
22
23     // External programmer compares IMD hash to it's own hashes
24     findCommonSecret(bitString, imdHash, lengthCommonSecret);
25   }
26
27   // The IMD performs its measurement
28   public static String[] imdMeasures(String[] bitString,
29       int lengthCommonSecret) {
30     // Give the IMD a random delay
31     Random rand = new Random();
32     int delay = rand.nextInt(bitString.length - lengthCommonSecret);
33     // String for storing the IMD measurement
34     String[] imdMeasurement = new String[lengthCommonSecret];
35     int imdMeasurements = 0;
36
37     // Perform measurement
38     for (int i = 0; i < bitString.length; i++) {
39       // Wait for delay to pass and take measurement
40       if (i >= delay && imdMeasurements < lengthCommonSecret) {
41         imdMeasurement[imdMeasurements] = bitString[i];
42         imdMeasurements++;
43       }
44     }
45
46     System.out.println("IMD delay: " + delay);
47     return imdMeasurement;
48   }
49
50   // Hash a measurement
```

```
51  // Method is a mockup of a true hash function.
52  public static String hashMeasurement(String[] measurement) {
53    String hash = "";
54
55    // Concatenate measurement
56    for (int i = 0; i < measurement.length; i++) {
57      hash += measurement[i];
58    }
59
60    // Hash concatenated measurement:
61    // Any hash algorithm can be used.
62    // SHA256 has been used here just as an example
63    hash = sha256(hash);
64
65    return hash;
66  }
67
68  // Find the common secret by iterating over the saved measurements and
69  // comparing it to the hashed IMD measurement
70  public static void findCommonSecret(String[] bitString, String imdHash,
71    int lengthCommonSecret) {
72    boolean foundCommonSecret = false;
73
74    // Loop trough the full measurement
75    for (int i = 0; i < bitString.length - lengthCommonSecret
76        && !foundCommonSecret; i++) {
77      String[] temp = new String[lengthCommonSecret];
78
79      // Store the needed measurements in temp in order to hash and compare
80      for (int j = 0; j < lengthCommonSecret; j++) {
81
82        temp[j] = bitString[i + j];
83      }
84
85      // Hash the stored measurement
86      String externalHash = hashMeasurement(temp);
87
88      System.out.println("IMD: " + imdHash + " external: " + externalHash);
89      // Compare the computed hash with received hash from the IMD
90      if (externalHash.equals(imdHash)) {
91        foundCommonSecret = true;
92        break;
93      }
94    }
95
96    if (foundCommonSecret) {
97      System.out.println("Succesfully establised common secret");
98    } else {
99      System.out.println("Failed to establish a common secret");
100   }
101
102 }
103
104 // Generate a full measurement
105 public static String[] generateBitString(int lengthBitString) {
106   String[] bitString = new String[lengthBitString];
107
108   for (int i = 0; i < lengthBitString; i++) {
109     // leading zero's get omitted, in order to prevent this make a bitstring of 9 bits and chop
      off the leading 1
110     Random rand = new Random();
111     bitString[i] = ""
112         + Integer.toBinaryString(
113             (0x100) | rand.nextInt((int) Math.pow(2, 8)))
114             .substring(1);
115   }
116
117   return bitString;
118 }
119
120 public static void writeToFile(int lengthBitString, String[] bitString)
```

```
121         throws FileNotFoundException {
122       PrintWriter wr = new PrintWriter("bitstring.txt");
123       wr.println(lengthBitString);
124
125       for (int i = 0; i < bitString.length; i++) {
126         wr.println(bitString[i]);
127       }
128
129       wr.close();
130     }
131
132     public static String[] readFromFile() throws FileNotFoundException {
133       Scanner sc = new Scanner(new File("bitstring.txt"));
134       int lengthBitString = sc.nextInt();
135       String[] bitString = new String[lengthBitString];
136
137       for (int i = 0; i < bitString.length; i++) {
138         bitString[i] = sc.next();
139       }
140
141       sc.close();
142       return bitString;
143     }
144
145     public static void measurementToString(String[] measurement) {
146       for (int i = 0; i < measurement.length; i++) {
147         System.out.println(measurement[i]);
148       }
149     }
150 }
```

# Bibliography


[1] Rajendra Acharya, Paul Joseph, Natarajan Kannathal, Choo Min Lim, and Jasjit Suri. Heart rate variability: a review. In *Medical and biological engineering and computing*, volume 44, pages 1031–1051. Springer, 2006.

[2] Regenscheid Andrew. *Cryptographic Standards and Guidelines Development*, volume 7977. NIST, 2016.

[3] Elaine Barker, William Barker, William Burr, William Polk, and Miles Smid. *Recommendation for key management part 1: General (revision 3)*, volume 800. NIST special publication, 2012.

[4] Larry Bassham, Çağdaş Çalık, Kerry McKay, Nicky Mouha, and Meltem Sönmez Turan. Profiles for the lightweight cryptography standardization process. *NIST draft*, 2017.

[5] Colin Boyd and Anish Mathuria. *Protocols for authentication and key establishment*. Springer Science & Business Media, 2013.

[6] Alejandro Calleja, Pedro Peris-Lopez, and Juan Tapiador. Electrical heart signals can be monitored from the moon: Security implications for ipi-based protocols. In *IFIP International Conference on Information Security Theory and Practice*, pages 36–51. Springer, 2015.

[7] Domenico Corrado, Antonio Pelliccia, Hein Heidbuchel, Sanjay Sharma, Mark Link, Cristina Basso, Alessandro Biffi, Gianfranco Buja, Pietro Delise, Ihor Gussac, et al. Recommendations for interpretation of 12-lead electrocardiogram in the athlete. *European heart journal*, 31(2):243–259, 2009.

[8] Tamara Denning, Kevin Fu, and Tadayoshi Kohno. Absence makes the heart grow fonder: New directions for implantable medical device security. In *HotSec*. USENIX, 2008.

[9] Tamara Denning, Alan Borning, Batya Friedman, Brian Gill, Tadayoshi Kohno, and William 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*, pages 917–926. ACM, 2010.

[10] Tracy S Diehl, Karen Kirk, and Jerry O'Shea. *ECG interpretation made incredibly easy*. Lippincott Williams & Wilkins, 2011.

[11] Tim Dierks and Eric Rescorla. Rfc 5246: The transport layer security (tls) protocol. *The Internet Engineering Task Force*, 2008.

[12] Whitfield Diffie, Paul C Oorschot, and Michael J Wiener. *Authentication and authenticated key exchanges*, volume 2. Springer, 1992.

[13] US Food and Drug Administation. Postmarket management of cybersecurity in medical devices. `https://www.fda.gov/downloads/medicaldevices/deviceregulationandguidance/guidancedocuments/ucm482022.pdf` Accessed May 2017. FDA, 2016.

[14] Kevin Fu. Inside risks reducing risks of implantable medical devices. *Communications of the ACM*, 52(6):25–27, 2009.

[15] A. Goldberger, L. Amaral, L. Glass, P. Hausdorff, J.and Ivanov, J. Mark, R.and Mietus, C. Moody, G.and Peng, and H. Stanley. PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals. *Circulation*, 101(23):e215–e220, 2000 (June 13). Circulation Electronic Pages: http://circ.ahajournals.org/content/101/23/e215.full PMID:1085218; doi: 10.1161/01.CIR.101.23.e215.

[16] Dieter Gollmann. What do we mean by entity authentication? In *Symposium on Security and Privacy*, pages 46–54. IEEE, 1996.

[17] Dieter Gollmann. Authentication-myths and misconceptions. In *Cryptography and Computational Number Theory*, pages 203–225. Springer, 2001.

[18] Solomon W Golomb. *SHIFT REGISTER SEQUENCES: Secure and Limited-Access Code Generators, Efficiency Code Generators, Prescribed Property Generators, Mathematical Models*. World Scientific, 1982.

[19] Paul Grassi, James Fenton, Elaine Newton, Ray Perlner, Andrew Regenscheid, William Burr, Justin Richer, Naomi Lefkovitz, Jamie Danker, and Choong. Digital identity guidelines, authentication and lifecycle management. *NIST Special Publication 800-63B*, 2017.

[20] Frank Gray. Pulse code communication, March 17 1953. US Patent 2,632,058.

[21] Daniel Halperin, Thomas Heydt-Benjamin, Benjamin Ransford, Shane Clark, Benessa Defend, Will Morgan, Kevin Fu, Tadayoshi Kohno, and William Maisel. Pacemakers and implantable cardiac defibrillators: Software radio attacks and zero-power defenses. In *Symposium on Security and Privacy*, pages 129–142. IEEE, 2008.

[22] Xiali Hei and Xiaojiang Du. Biometric-based two-level secure access control for implantable medical devices during emergencies. In *Proceedings INFOCOM*, pages 346–350. IEEE, 2011.

[23] ISO 9798-2. Information technology – Security techniques – Entity authentication – part 2. Standard, International Organization for Standardization, Geneva, CH, July 1999.

[24] Anil Jain, Ruud Bolle, and Sharath Pankanti. *Biometrics: personal identification in networked society*, volume 479. Springer Science & Business Media, 2006.

[25] Yeun-Ho Joung. Development of implantable medical devices: from an engineering perspective. In *International neurourology journal*, volume 17, pages 98–106. NCBI, 2013.

[26] Ari Juels and Martin Wattenberg. A fuzzy commitment scheme. In *Proceedings of the 6th ACM conference on Computer and communications security*, pages 28–36. ACM, 1999.

[27] Laurence Keselbrener, Michel Keselbrener, and Solange Akselrod. Nonlinear high pass filter for r-wave detection in ecg signal. In *Medical engineering & physics*, volume 19, pages 481–484. Elsevier, 1997.

[28] B Kohler, Carsten Hennig, and Reinhold Orglmeister. The principles of software qrs detection. In *Engineering in Medicine and Biology Magazine*, number 1, pages 42–57. IEEE, 2002.

[29] Fred Kusumoto and Nora Goldschlager. Remote monitoring of patients with implanted cardiac devices. In *Clinical cardiology*, volume 33, pages 10–17. Wiley Online Library, 2010.

[30] Jong-Min Lee, Dae-Jin Kim, In-Young Kim, Kwang-Suk Park, and Sun I Kim. Detrended fluctuation analysis of eeg in sleep apnea using mit/bih polysomnography data. *Computers in biology and medicine*, 32(1):37–47, 2002.

[31] Cuiwei Li, Chongxun Zheng, and Changfeng Tai. Detection of ecg characteristic points using wavelet transforms. *IEEE Transactions on biomedical Engineering*, 42(1):21–28, 1995.

[32] Eduard Marin, Dave Singelée, Flavio D Garcia, Tom Chothia, Rik Willems, and Bart Preneel. On the (in) security of the latest generation implantable cardiac defibrillators and how to secure them. In *Proceedings of the 32nd Annual Conference on Computer Security Applications*, pages 226–236. ACM, 2016.

[33] R Mark and G Moody. Mit-bih arrhythmia database directory. *Cambridge: Massachusetts Institute of Technology*, 1988.

[34] Kerry A McKay, Larry Bassham, Meltem Sönmez Turan, and Nicky Mouha. Report on lightweight cryptography. *NIST DRAFT NISTIR*, 8114, 2016.

[35] Nicky Mouha. Public comments on profiles for lightweight cryptography standarization process. *https://www.nist.gov/sites/default/files/documents/2017/06/20/public-comments-profiles-i-ii-june2017.pdf*, 2017.

[36] Conor Patrick and Patrick Schaumont. *The Role of Energy in the Lightweight Cryptographic Profile*. NIST, 2012.

[37] Jerome Radcliffe. Hacking medical devices for fun and insulin: Breaking the human scada system. In *Black Hat Conference presentation slides*, volume 2011, 2011.

[38] Masoud Rostami, Ari Juels, and Farinaz Koushanfar. Heart-to-heart (h2h): authentication for implanted medical devices. In *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*, pages 1099–1112. ACM, 2013.

[39] Andrew Rukhin, Juan Soto, James Nechvatal, Miles Smid, and Elaine Barker. A statistical test suite for random and pseudorandom number generators for cryptographic applications. Technical report, Booz-Allen and Hamilton Inc Mclean Va, 2001.

[40] Robert J Russo, Heather S Costa, Patricia D Silva, Jeffrey L Anderson, Aysha Arshad, Robert WW Biederman, Noel G Boyle, Jennifer V Frabizzio, Ulrika Birgersdotter-Green, Steven L Higgins, et al. Assessing the risks associated with mri in patients with a pacemaker or defibrillator. *New England Journal of Medicine*, 376(8):755–764, 2017.

[41] Jerome H Saltzer and Michael D Schroeder. The protection of information in computer systems. *Proceedings of the IEEE*, 63(9):1278–1308, 1975.

[42] Robert M Seepers, Jos H Weber, Zekeriya Erkin, Ioannis Sourdis, and Christos Strydis. Secure key-exchange protocol for implants using heartbeats. In *Proceedings of the ACM International Conference on Computing Frontiers*, pages 119–126. ACM, 2016.

[43] Robert M Seepers, Wenjin Wang, Gerard de Haan, Ioannis Sourdis, and Christos Strydis. Attacks on heartbeat-based security using remote photoplethysmography. *IEEE Journal of Biomedical and Health Informatics*, 2017.

[44] Claude E Shannon. A mathematical theory of communication. *ACM SIGMOBILE Mobile Computing and Communications Review*, 5(1):3–55, 2001.

[45] Adam Shostack. *Threat modeling: Designing for security*. John Wiley & Sons, 2014.

[46] Joshua Smith, Alanson Sample, Pauline Powledge, Sumit Roy, and Alexander Mamishev. A wirelessly-powered platform for sensing and computation. In *Ubiquitous Computing*, pages 495–506. Springer, 2006.

[47] Mika P Tarvainen, Perttu O Ranta-Aho, and Pasi A Karjalainen. An advanced detrending method with application to hrv analysis. *IEEE Transactions on Biomedical Engineering*, 49(2):172–175, 2002.

[48] Mika P Tarvainen, Juha-Pekka Niskanen, Jukka A Lipponen, Perttu O Ranta-Aho, and Pasi A Karjalainen. Kubios hrv–heart rate variability analysis software. *Computer methods and programs in biomedicine*, 113(1):210–220, 2014.

[49] European Union. Regulation 2016/676 of the european parliament and of the council. `http://eur-lex.europa.eu/legal-content/EN/TXT/?uri=uriserv:OJ.L_.2016.119.01.0001.01.ENG&toc=OJ:L:2016:119:TOC` Accessed April 2017. EU, 2016.

[50] Fengyuan Xu, Zhengrui Qin, Chiu C Tan, Baosheng Wang, and Qun Li. Imdguard: Securing implantable medical devices with the external wearable guardian. In *INFOCOM, 2011 Proceedings IEEE*, pages 1862–1870. IEEE, 2011.

[51] Teng Xu, James B Wendt, and Miodrag Potkonjak. Matched digital pufs for low power security in implantable medical devices. In *International Conference on Healthcare Informatics (ICHI)*, pages 33–38. IEEE, 2014.

[52] Yulong Zou, Jia Zhu, Xianbin Wang, and Lajos Hanzo. A survey on wireless security: Technical challenges, recent advances, and future trends. volume 104, pages 1727–1765. IEEE, 2016.