Cybersecurity Assessment of the Polar Bluetooth Low Energy Heart-rate Sensor

S. Soderi[1]

IEEE Senior Member
soderi@ieee.com

Abstract. Wireless communications among wearable and implantable devices implement the information exchange around the human body. Wireless body area network (WBAN) technology enables non-invasive applications in our daily lives. Wireless connected devices improve the quality of many services, and they make procedures easier. On the other hand, they open up large attack surfaces and introduces potential security vulnerabilities. Bluetooth low energy (BLE) is a low-power protocol widely used in wireless personal area networks (WPANs). This paper analyzes the security vulnerabilities of a BLE heart-rate sensor. By observing the received signal strength indicator (RSSI) variations, it is possible to detect anomalies in the BLE connection. The case-study shows that an attacker can easily intercept and manipulate the data transmitted between the mobile app and the BLE device. With this research, the author would raise awareness about the security of the heart-rate information that we can receive from our wireless body sensors.

Keywords: Bluetooth · BLE · security · sensor · MitM · heart-rate · WBAN · privacy.

1 Introduction

In the last two decades, Bluetooth became very popular in the short-range communications. Every smartphone, tablet and personal computer embeds this technology. At the same time, many wireless sensors such as fitness sensors, smart watches, headsets, wireless medical devices (WMDs) rely on Bluetooth to exchange data with the user's smartphone or tablet. Today, wireless body area networks (WBANs) collect humans' information through Bluetooth low energy (BLE) sensors nodes. BLE is thus becoming de-facto a key wireless technology and users leave that interface always enabled on their devices. BLE was introduced as *Wibree* by Nokia in 2006 [21]. Today, BLE is the dominant technology to convey efficiently data in body networks using coin cell battery-powered devices.

Though the advantages offered by any WBAN are substantial, it makes one of the prime targets for security threats and more particularly for users' privacy. WBANs are commonly used to track health and fitness data and it raises the interest of cyber-criminals to this kind of information.

Fitness wearable devices collect human's information, and they are designed to be worn all day. The user reads these data through his smartphone, tablet or even by his smartwatch. Historically, these fitness trackers have numerous security vulnerabilities and it can happen that the wireless sensor or even the software application discloses users' data. It is clear that it might have important privacy implications [17, 18]. In the literature, there are several contributions that deals with security aspects in WBAN health-care applications. Indeed, the security weaknesses of WMDs can lead to a high risk for patient's safety [22]. The rapid proliferation of wireless implantable medical devices (WIMDs) coupled with their increasing features is raising the risk for patients [23].

The utilization of wireless technology makes the data prone to being eavesdropped, modified and injected. This increases concerns about the privacy of the information managed in WBANs. In this paper, author is considering man-in-the-middle (MitM) attack in a BLE WBAN fitness scenario. Observing the received signal strength indicator (RSSI) variations, this study proposes a mechanism to detect MitM attacks.

The rest of this paper is organized as follows. Section 2 overviews the BLE specifications. Section 3 describes the security vulnerabilities of a BLE heart-rate sensor and the results of a MitM attack. Then, the paper proposes security countermeasures. Finally, conclusions are presented in Section 4.

2 Bluetooth Low Energy (BLE)

2.1 BLE Core Specifications

Bluetooth is an open standard used for short-range communications. This wireless technology operates in the 2.4 GHz ISM band and it is primarily used for consumer, medical and personal devices [8]. With Bluetooth users are able to create personal ad-hoc networks to transfer any kind of data. Above 5 billions Bluetooth devices are expected to be shipped within 2022 [7]. At the time of writing, Bluetooth 5.0 is the most recent version and it is rapidly adopted in smartphones. Despite the Bluetooth Special Interest Group (SIG) releases newer versions, even older ones are currently in use and is common to find Bluetooth 4.1 and 4.2 in commercial devices [7, 9]. Bluetooth architecture specifies two forms: basic rate/enhanced data rate (BR/EDR) and low energy (LE) [4]. This paper refers to the BLE standard.

Table 1 shows a comparison of the lower layers between BLE and Bluetooth BR/EDR. This comparison indicates a different usage of the radio spectrum. BLE splits spectrum in 40 channels: 3 advertising channels to establish connections and 37 channels to transmit data. Furthermore, BLE devices are designed to send short bursts of data rather than a continuous data stream. It makes BLE ideal for sensor applications. BLE devices consume very low energy in comparison to other wireless technologies.

BLE is a full protocol stack. It is a combination of hardware parts and software layers [4]. As shown in Figure 1, BLE architecture is organized in three

Table 1. Lower layers comparison between BLE and Bluetooth BR/EDR [8]

Characteristic	**Bluetooth LE (BLE)**	**Bluetooth BR/EDR**
Frequency band	2.4 GHz	2.4 GHz
Channels	40 channels with 2 MHz spacing (3 advertising ch./37 data ch.)[1]	79 channels with 1 MHz spacing
Channel usage	FHSS	FHSS
Modulation	GFSK	GFSK, $\frac{\pi}{4}$ DQPSK, 8DPSK
Max data-rate	2 Mbps	3 Mbps
Max Tx power	100 mW	100mW
Power consumption	$(0.01 \div 0.05)\cdot(1)^2$	$(1)^2$
Network topologies	Point-to-Point,[3] Broadcast, Mesh	Point-to-Point[3]
Connection	Short burst data transmission	Continuous data stream
Typical range	30 m	50 m

[1] Advertising channels: ch. 37 (2402 MHz), ch. 38 (2426 MHz) and ch. 39 (2480 MHz);
[2] (1) is the reference value;
[3] Including piconet.

major blocks: applications, host and controller. The user *application* defines the interface with the Bluetooth stack. *Host* block consists of the upper layers, whereas *controller* includes lower layers. Host and controller communicate

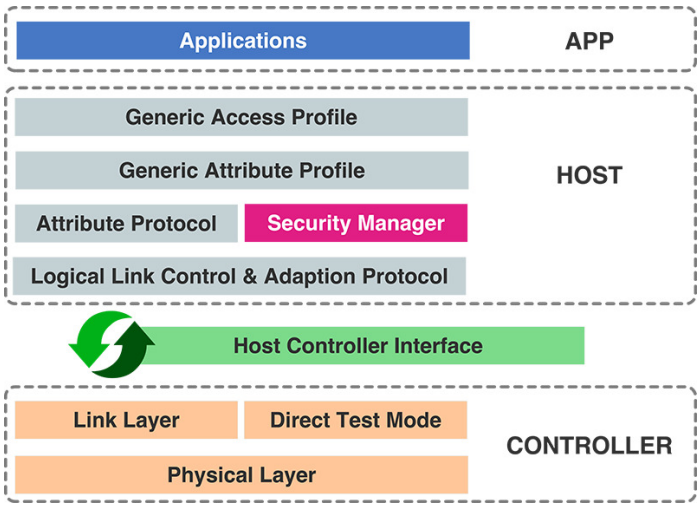

Fig. 1. Architecture of BLE.

through the host controller interface (HCI). This division makes possible to interface many hosts with a single controller by using the HCI.

The generic access profile (GAP) layer controls the role and connection of a BLE device. BLE specifications define GAP roles as follows [4]

- **Broadcaster**:a device that only sends advertising events;
- **Observer**: a device that only receives advertising events;
- **Peripheral**: a device that accepts the establishment of a LE physical link using the connection establishment procedure;
- **Central**:a device that initiates the establishment of a physical connection.

The logical link control and adaptation protocol (L2CAP) layer plays a central role in Bluetooth stack. It takes multiple protocols from the upper layers and encapsulates them into the standard BLE packet format and vice-versa. Actually, L2CAP layer is in charge or routing two main protocols: the attribute protocol (ATTP) and the security manager (SM).

Figure 2 shows the connections flow between master and slave BLE devices. In a fitness scenario, the central device, e.g. a smartphone, scans the frequencies for connectable advertising packets. The peripheral devices, e.g. the heart-rate sensor sends connectable advertising packets periodically and accepts incoming connections. Once a connection is established master and slave use generic attribute (GATT) profiles to exchange data. GATT is a simple structured list. Indeed, the data in GATT is organized in services and each service contains one or more characteristics. Each characteristic consists of a universally unique identifier (UUID), a value and a set of properties. Bluetooth SIG defined UUIDs to identify BLE manufacturers as well [3]. By reading UUIDs data, the hacker might gather useful information to plan his attack.

2.2 BLE Security

In the BLE architecture (Figure 1) SM is responsible for paring, integrity, authentication, and encryption [21,24]. SM distributes security keys between peers and provides cryptographic functionalities.

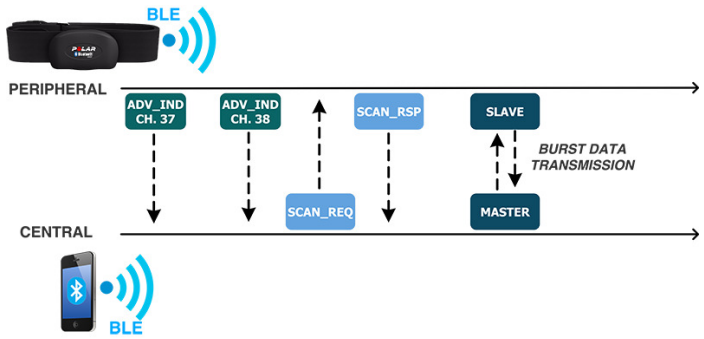

Fig. 2. BLE connection flow.

NIST 800-121-R2 details the security capabilities of Bluetooth and makes recommendations to effectively secure these devices. BLE security is different from Bluetooth BR/EDR. Since the introduction of the BLE 4.0, the protocol supports a 128-bit advanced encryption standard–counter with CBC-MAC (AES-CCM) [24]. Although AES is considered one of the most secure forms of encryption, the key exchange protocol is exploitable. Indeed, during the *pairing* process devices in BLE 4.0 and 4.1 versions exchange a temporary key (TK) and use it to create a short-term key (STK). These keys are used to encrypt the communication. In this case, an attacker can eavesdrop the keys and then decrypt the connection. This is not the case for BLE version 4.2 and beyond due to the introduction of a long-term key (LTK) which uses the elliptic curve Diffie Hellman (ECDH) key exchange, which is proved to be secure under this type of attack [19, 24].

BLE 4.0 and 4.1 devices use the secure simple pairing (SSP) model, in which devices based on their input/output (I/O) capabilities, choose one method from

- **Just Works**: TK is all zeros;
- **Passkey**: TK is a six-digit number combination inserted by the user;
- **Out-of-Band (OOB)**: TK is exchanged through a different medium.

The SM can protect the connection from MitM when the operating system (OS) selects Passkey or OOB as paring method. On the other hand, Just Works method does not provide any protection against MitM, that can be exploited by potential hackers.

3 Cybersecurity in a WBAN Fitness Scenario

Despite the Bluetooth SIG released the new Bluetooth 5.0, there is still a huge number of devices in use that utilizes older versions of Bluetooth, such as version 4.1 and 4.2 [7]. In fact, based on a recent estimation last year there were 4 billions BLE enabled devices using version 4.0 or 4.1 [19]. Based on this information, the paper addresses security issues in BLE 4.1 devices.

As shown in Figure 3, the scenario investigated in this paper, includes a Polar H7 heart-rate BLE sensor worn at appropriate positions on the body [12]. The device communicates a person's activity data through the BLE protocol to the smartphone. Due to their nature, WBAN might experience eavesdropping attacks. In this scenario, security and privacy are among major areas of concerns.

Once the author described BLE protocol in the previous sections, the system analysis must be completed by describing the interfaces present in each device. Figure 4 shows the interfaces between the person, the Polar heart-rate sensor and the smartphone which runs the app to perform the synchronization. In particular, by using a SysML internal block diagram (IBD) [14], the author highlighted only those interfaces that might have a key role in this analysis.

The author has already discussed herein how the pairing procedure in BLE 4.1 makes these devices prone to eavesdropping attacks. The use of each association model is based on the I/O capabilities of the device. Considering that

Fig. 3. WBAN fitness scenario.

the smartphone, i.e. the *initiator*, is equipped with a display and a keyboard display that can be used for pairing by the user. Whereas, the heart-rate sensor, i.e. the *responder*, does not have any I/O capability. Thus, in the scenario under investigation and by analyzing the available interfaces, Table 2 represents all possible pairing options. Accordingly, with the SSP model in this scenario only the Just Works method can be used. The user is required to accept a connection without verifying TK value on both devices, so Just Works provides no MitM protection because it is an *unauthenticated* pairing.

The common type of attacks against BLE communications are

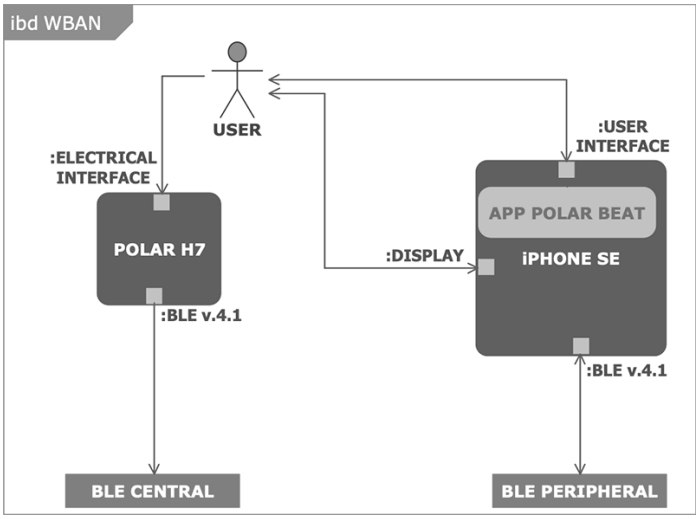

Fig. 4. WBAN fitness interfaces representation with SysML.

Table 2. Pairing procedure and I/O capabilities in the BLE fitness scenario

		Initiator[1]	
Responder[2]	**I/O Capabilities**	**Display Only**	**Keyboard Display**
	No Input No Output	Just Works (Unauthenticated)	Just Works (Unauthenticated)

[1] Smartphone;
[2] Heart-rate BLE sensor;

- **MitM** in which an attacker has the ability to both monitor and alter or inject messages into a communication channel;
- **Passive Eavesdropping** in which the attacker is secretly listening (by using a sniffing device) to the private communication of others without consent.

3.1 BLE 4.1 Assessment

This section describes the *cybersecurity assessment* of BLE 4.1 devices in a fitness scenario as shown in Figure 3. The assessment combined multiple methodologies to best fit the investigation needs. The author selected NIST 800 series to evaluate threats and vulnerabilities of BLE sensors [15,24]. Furthermore, the OWASP guideline is adopted to rate the risk associated to each vulnerability [16].

Tables 3 to 7 report found vulnerabilities for the scenario under investigation. In accordance with the methodology selected each table ties together concepts such as likelihood, technical impact, risk and threat events that could exploit the vulnerability. Moreover, for each vulnerability, the author provides a description

Table 3. BLE 4.1 vulnerability n.1

Vulnerability n.1	
Vulnerability	**Low energy legacy pairing provides no passive eavesdropping protection.**
Likelihood	High
Technical Impact	High
Risk	Critical
Threat Event	Passive Eavesdropping
Description	Eavesdroppers can capture secret keys (i.e., LTK) distributed during low energy pairing.
Mitigation	BLE devices should be paired by using an algorithm that provides a mechanism to exchange keys over an unsecured channel. For instance the ECDH.

Table 4. BLE 4.1 vulnerability n.2

Vulnerability n.2	
Vulnerability	**The Just Works pairing method provides no MITM protection.**
Likelihood	High
Technical Impact	High
Risk	Critical
Threat Event	MitM attack
Description	MITM attackers can capture and manipulate data transmitted between trusted devices.
Mitigation	Low energy devices should be paired in a secure environment to minimize the risk of eavesdropping and MITM attacks. Just Works pairing should not be used for low energy.

Table 5. BLE 4.1 vulnerability n.3

Vulnerability n.3	
Vulnerability	**No user authentication exists.**
Likelihood	Medium
Technical Impact	High
Risk	High
Threat Event	Pairing Eavesdropping
Description	Only device authentication is provided by the specification.
Mitigation	Application-level security, including user authentication, can be added via overlay by the application developer.

and possible mitigation. Since these vulnerabilities have a risk mainly rated between *high* and *critical*, it should raise some concerns about the security of the heart-rate information transmitted by the sensor.

3.2 Experiment with MitM Attack

Based on the cybersecurity assessment, in this section, the author proposes an active MitM attack for testing the BLE WBAN in a fitness scenario.

MitM usually involves three actors: Alice, Bob and Eve. In BLE networks this attacks changes its architecture. Indeed, the attacker, i.e. Eve, cannot act simultaneously as a sensor and as mobile app. Therefore, a BLE MitM needs to make use of two BLE components capable of acting together: one connects to the mobile app acting as the smartphone, while the other connects to the smartphone acting as the mobile app [21].

The WBAN experiment setup consisted of a Polar H7 heart-rate sensor, i.e. Alice, and an Apple iPhone SE [2], i.e. Bob. The synchronization between the

Table 6. BLE 4.1 vulnerability n.4

Vulnerability n.4	
Vulnerability	**End-to-end security is not performed.**
Likelihood	Medium
Technical Impact	Medium
Risk	Medium
Threat Event	MitM attack
Description	Only individual links are encrypted and authenticated. Data is decrypted at intermediate points.
Mitigation	End-to-end security on top of the Bluetooth stack can be provided by use of additional security controls.

Table 7. BLE 4.1 vulnerability n.5

Vulnerability n.5	
Vulnerability	**Discoverable and/or connectable devices are prone to attack.**
Likelihood	Medium
Technical Impact	High
Risk	High
Threat Event	Passive Eavesdropping, MitM attack
Description	A hacker can try to take over any discoverable and/or connectable BLE device, and then he can get access to all the information.
Mitigation	Any device that must go into discoverable or connectable mode to pair or connect should only do so for a minimal amount of time. A device should not be in discoverable or connectable mode all the time.

sensor and the smartphone is performed over BLE 4.1. The smartphone ran the Polar Beat mobile app for real-time heart-rate monitoring [13]. As shown in Figure 5, Eve consists of a laptop that runs Linux Ubuntu 18.10 and two CSR 8510-based USB dongles that support BLE 4.1. These two dongles are connected to the laptop and communicate with each other using the *BtleJuice* web-based software [11].

BtleJuice is a framework to perform MitM attacks on BLE devices. This framework consists of two parts which run on two virtual machines hosted by the same laptop. These parts named *interception core* and *proxy*, and they implement the MitM architecture shown in Figure 5. And to do this, each virtual machine manages one USB dongle.

BtleJuice acts as a proxy between the mobile app and the BLE heart-rate sensor. Any command sent to the sensor is captured by BtleJuice and relayed to

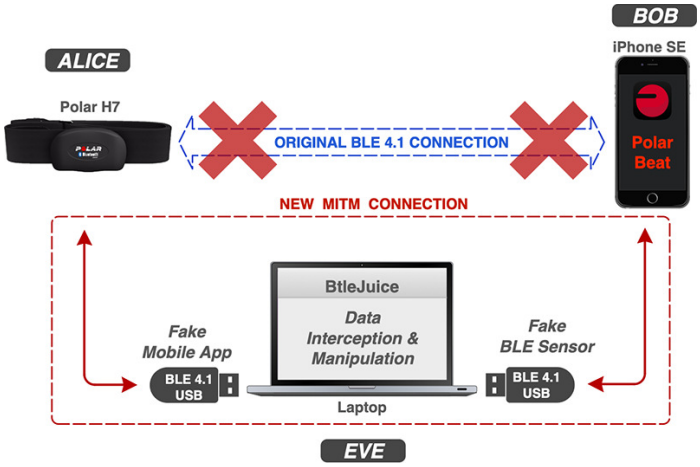

Fig. 5. Active MitM architecture for BLE fitness scenario.

the sensor. In particular, the interception proxy interacts with BLE peripherals and the interception core generates the fake devices with a fake BLE address. Then, the attacker from the web user interface (UI) can control the interception core. He can select the BLE target and intercept GATT operations. From the UI, it is possible to replay any GATT operation, but it allows also on-the-fly (OTF) data modification.

In WBAN BLE fitness scenario, Polar H7 heart-rate sensor communicates with the Polar Beat. Since the pairing process is completed, the author started

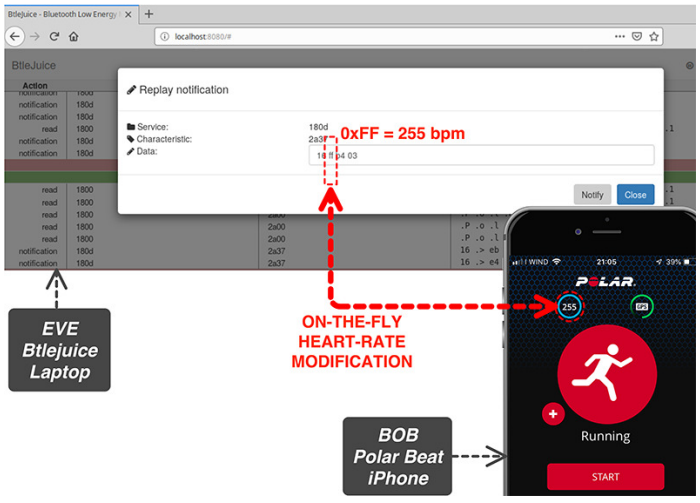

Fig. 6. Heart-rate on-the-fly modification by using replay feature in Btlejuice.

the experiment by attempting to actively sniff BLE traffic. Figure 6 shows how Eve, by using BtleJuice, is able to intercept the BLE information exchanged between the peripheral and the mobile app and manipulate data OTF.

Once Btlejuice is initialized, the UI allows Eve the selection and the connection of the Polar H7 BLE sensor. In this way, Bob rather than connect to its peripheral, he connects his mobile app to the fake device. As shown in Figure 6, the attacker by using the replay function in Btlejuice is able to modify the heart-rate measurement, i.e. characteristics 0x2$A37$ [5], inside the heart-rate service, i.e. 0x180D [6]. As proof of concept (POC), Eve pushes 255 beats per minute (bpm) in the Polar Beat app.

3.3 Discussion about Security Countermeasures

As pointed out during the cybersecurity assessment (Section 3.1), BLE specifications do not offer defenses against MitM attacks. Although the experiment is limited to the WBAN fitness scenario described in this paper, these security leaks can give an advantage to the attacker.

On the L2CAP layer, there is the possibility to request an echo from the BLE sensor to measure round trip time (RTT) on the established link by using the *l2ping* command. It is included in the BlueZ utils [10]. Author assumes that the RTT of the MitM connection in Figure 5 is greater than the one in the BLE original connection. The evaluation of the RTT might offer a mechanism to detect the MitM. Unfortunately, the *l2ping* command is not supported in most of the BLE peripherals.

Alternatively, the evaluation of the RSSI, might offer another way to detect MitM attack. In the literature, there are several contributions to the relationship between the RSSI and the distance. In [20, 25], RSSI is described as follows

$$RSSI = -10 \cdot N \cdot log(d) + a, \tag{1}$$

where N is a constant assumed 1, d is the distance in meters and a is the transmitted power at 1-meter distance.

The author measured the RSSI by using the iPhone and the Bluefruit mobile app [1]. Table 8 shows the average value and the standard deviation of the RSSI over 10 measurements for each distance. These measurements validate the RSSI model in (1). Although indoor environment and layout settings have a

Table 8. RSSI measurements

RSSI	**Distance [m]**			
	0	0.5	1	3
Mean [dBm]	-26.4	-52.8	-60.8	-66
Std. Dev. [dB]	1.2	3.3	2.6	3

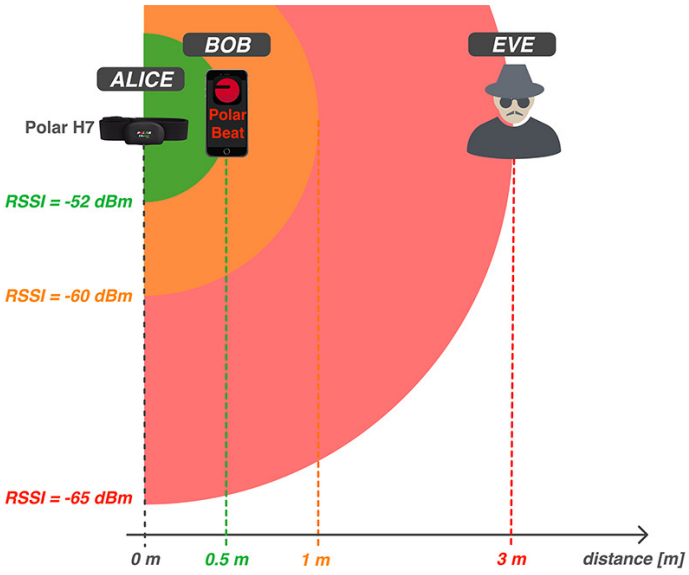

Fig. 7. RSSI .

direct effect on the RSSI variability, the author assumes that Eve's RSSI can be greater than Bob's RSSI. Figure 7 and Table 8 confirm the assumption. In this scenario, by monitoring the RSSI value Bob can have a mechanism to detect the attack. Indeed, if the RSSI increased unexpectedly then the mobile app might alert the user about a possible attack.

4 Conclusions

This paper aims to raise a concern about the need to be aware on the use of BLE devices. The author analyzed the security issues of BLE 4.1. based sensors. Using the NIST classified threats, the author has identified a list of attacks which are applicable to BLE devices.

The WBAN scenario under test consists of a Polar H7 heart-rate sensor that communicates with the Polar Beat mobile app using the BLE technology. Being Polar Electro a top player in the wearable smart health devices, its BLE-based sensors are spread worldwide. With this research, the author would raise awareness about the security of the heart-rate information that we can receive from our wireless body sensors.

The case study shows that an attacker can easily intercept and manipulate the data transmitted between the mobile app and the BLE device. Btlejuice was used to implement an active MitM attack, an operation that can result in the OTF modification of the data. The author remarks the importance to detect this kind of attack that might modify sensitive information such as the heart-rate.

References

1. Adafruit Bluefruit LE Connect, https://itunes.apple.com/it/app/adafruit-bluefruit-le-connect/id830125974?mt=8
2. Apple iPhone SE - Technical Specifications, https://support.apple.com/kb/sp738?locale=en_GB
3. Bluetooth 16 Bit UUIDs For Members, https://www.bluetooth.com/specifications/assigned-numbers/16-bit-uuids-for-members
4. Bluetooth Core Specifications, https://www.bluetooth.com/specifications/bluetooth-core-specification
5. Bluetooth GATT Characteristics , https://www.bluetooth.com/specifications/gatt/characteristics/
6. Bluetooth GATT Services, https://www.bluetooth.com/specifications/gatt/services/
7. Bluetooth Market Update 2018, https://www.bluetooth.com/markets/market-report
8. Bluetooth Radio Versions, https://www.bluetooth.com/bluetooth-technology/radio-versions
9. Bluetooth SIG, https://www.bluetooth.com
10. BlueZ, An Official Linux Bluetooth protocol stack , http://www.bluez.org
11. BtleJuice Bluetooth Smart (LE) Man-in-the-Middle framework, https://github.com/DigitalSecurity/BtleJuice
12. Polar, https://www.polar.com/en
13. Polar Beat Free Fitness and Training App, https://www.polar.com/en/products/polar_beat
14. SysML Open Source Project - What is SysML?, https://sysml.org
15. NIST 800-30. Guide for Conducting Risk Assessments Revision 1 (2012)
16. OWASP Testing Guide v4 (2014), https://www.owasp.org/index.php/OWASP_Testing_Project
17. Cyr, B.S., Horn, W., Miao, D., Specter, M.: Security analysis of wearable fitness devices (fitbit) (2014), https://pdfs.semanticscholar.org/f4ab/ebef4e39791f358618294cd8d040d7024399.pdf
18. Das, A.K., Pathak, P.H., Chuah, C.N., Mohapatra, P.: Uncovering Privacy Leakage in BLE Network Traffic of Wearable Fitness Trackers. In: Proceedings of the 17th International Workshop on Mobile Computing Systems and Applications. pp. 99–104. HotMobile '16, ACM, New York, NY, USA (2016). https://doi.org/10.1145/2873587.2873594, http://doi.acm.org/10.1145/2873587.2873594
19. Filizzola, D., Fraser, S., Samsonau, N.: Security analysis of bluetooth technology (2018), https://courses.csail.mit.edu/6.857/2018/project/Filizzola-Fraser-Samsonau-Bluetooth.pdf
20. Karani, R., Dhote, S., Khanduri, N., Srinivasan, A., Sawant, R., Gore, G., Joshi, J.: Implementation and design issues for using Bluetooth low energy in passive keyless entry systems. In: 2016 IEEE Annual India Conference (INDICON). pp. 1–6 (Dec 2016). https://doi.org/10.1109/INDICON.2016.7838978
21. Melamed, T.: An active man-in-the-middle attack on bluetooth smart devices. International Journal of Safety and Security Engineering **8**, 200–211 (02 2018). https://doi.org/10.2495/SAFE-V8-N2-200-211
22. Partala, J., Keränen, N., Särestöniemi, M., Hämäläinen, M., Iinatti, J., Jämsä, T., Reponen, J., Seppänen, T.: Security threats against the transmission chain of a

medical health monitoring system. In: 2013 IEEE 15th International Conference on e-Health Networking, Applications Services (Healthcom). pp. 243–248 (Oct 2013), https://doi.org/10.1109/HealthCom.2013.6720675
23. Pycroft, L., Aziz, T.Z.: Security of implantable medical devices with wireless connections: The dangers of cyber-attacks. Expert Review of Medical Devices **15**(6), 403–406 (2018). https://doi.org/10.1080/17434440.2018.1483235, https://doi.org/10.1080/17434440.2018.1483235, pMID: 29860880
24. Scarfone, K.A., Padgette, J.: NIST SP 800-121. Guide to Bluetooth Security (2008)
25. Tosi, J., Taffoni, F., Santacatterina, M., Sannino, R., Formica, D.: Performance Evaluation of Bluetooth Low Energy: A Systematic Review. Sensors **17**, 2898 (12 2017). https://doi.org/10.3390/s17122898