# Ephemeral Elliptic Curve Diffie-Hellman to Secure Data Exchange in Internet of Medical Things


**Osman Salem and Ahmed Mehaoua**

## 1 Introduction

With the advances in information and communication technologies, the Internet of Medical Things (IoMT) becomes a promising solution for remote healthcare monitoring, where a set of wearable biosensors are used to collect the physiological data from the monitored patient, and to transmit the acquired measurements to a Local Processing Unit (LPU—such as Smartphone or tablet) for processing and alerting the healthcare professionals when an emergency is detected. Such monitoring systems are able to assist the healthcare professionals by analyzing the acquired physiological data in the edge of the network, and raising an alarm when an anomaly is detected by highlighting abnormal changes in monitored parameters. The use of IoMT for remote monitoring, and for the detection of chronic diseases gives impetus to the development and implementation of enriched and ubiquitous health services.

The use of IoMT devices provides a tool to improve the Quality of Life (QoL) by allowing the monitored patient to continue their Activity of Daily Living (ADL) while being monitored and followed-up. Their fast deployment has an impact on reducing the number of beds occupied by patients kept under monitoring. The COVID-19 pandemic has driven an exponential rise in IoMT, with quarantine and stay-at-home orders, which accelerated trends in telemedicine and telehealth.

However, the medical data involves stringent security requirements which are not available in sensors with restricted resources [1]. The collected sensitive medical data is transmitted to the LPU for processing using wireless technologies, and an attacker in vicinity can eavesdrop or modify the intercepted data [2] leading to false

O. Salem (✉) · A. Mehaoua
Borelli Research Center, University of Paris, Paris, France
e-mail: osman.salem@u-paris.fr; ahmed.mehaoua@u-paris.fr



K. Daimi et al. (eds.), *Emerging Trends in Cybersecurity Applications*,
https://doi.org/10.1007/978-3-031-09640-2_1

alarms, or can conduct a black hole attack by preventing information from being transmitted to the LPU, in order to prevent the system from raising alarms. The attacker may also exploit the vulnerabilities [3] in the software of IoMT device to increase the transmission rate and deplete the energy of sensors or to flood the LPU. Therefore, a security framework is required to ensure the integrity of the exchanged data.

Several mechanisms have been proposed and tested in the literature for securing the exchanged data between the sensors and the LPU [4]. The Bluetooth Low Energy (BLE) is widely implemented today in IoMT to transmit data from sensors to the LPU. The IoMT object requires a short range communication, low bandwidth, low delay, and reduced energy consumption. BLE exchanges less data than normal Bluetooth to reduce energy, and devices can stay in "sleep mode" until the next interaction. These advantages have led to this wireless technology being widely deployed in IoMT for remote monitoring of patients during long periods of time (months and even years) without charging or changing the battery.

Devices in BLE are classified into two types: central and peripheral. The central device (e.g., smartphone) has higher computational power and storage than peripherals and sends commands and collects data from peripherals. Conversely, the peripheral or the slave cannot initiate a connection and can only connect to a single master. It only executes received orders and sends packages to advertise its presence. The peripherals stop sending advertising packets when they receive a specific packet, indicating that they are connected to a central device. Peripherals are sensors that collect and send data to the central device for processing, such as the collection of blood pressure, SpO2 and body temperature, and other physiological parameters by sensors, as well as their transmission to a central processing unit (smartphone or tablet).

BLE operates using radio frequency on 2.4–2.8 GHz band within a distance of 10 m. It operates with 40 physical channels, against 80 for legacy Bluetooth, for frequency and time multiplexing thanks to the L2CAP layer. The difference between two channels is found to be 2 MHz. The devices in advertising mode send packets of 31 bytes at regular intervals. This task is conducted only on 3 of the channels: 37, 38, and 39. The other channels are reserved for data exchange between devices [5].

To establish a connection, the central device alternates between scanning for pairing requests and sending advertising packets. It scans to check if it can find a peripheral to begin the exchange with it. The scanning process is expensive, so the scan usually does not run indefinitely. The BLE devices exchange their services, their capabilities, their inputs (such as the presence of keyboard or not) and output resources, their names and their manufacturers' information, authentication method, etc. during the first phase of pairing, which is not encrypted. However, the second phase is for key exchange and needs to be secured.

In the second phase of pairing, one of the devices generates a Temporary Key (TK) which will be known from both devices. Confirmation of the key is made through the exchange of random values, encrypted and then decrypted. With the TK and random values, a Short-Term Key (STK) is derived by devices without traveling in

the network. The connection will be encrypted with this key at the link layer level. Eventually, a Long Term Key (LTK) can be exchanged for bonding.

Four pairing models are supported by BLE: "Just Works," Out of Band, Numerical Comparison, and Passkey. The BLE secures the communication using the Advanced Encryption Standard (AES) algorithm with a key length of 128 bits. However, when the object does not have I/O capabilities, the BLE "Just Works" pairing mode does not provide any protection against MitM (Man in the Middle) or eavesdropping. As the IoMT device does not have display or keyboard, the default value of pairing code $0x00$ is used as value for TK ($TK = 0$), which in turn is used to derive the STK and the LTK.

In other words, we can connect to any BLE device that uses the "Just Works" pairing mode and access the exchanged medical data. In fact, this pairing mode is deployed in several healthcare devices available in the market, and it does not provide any protection against MitM and must not be used in healthcare monitoring services. In the real world, sensors do not have I/O interfaces and this mode is currently deployed in healthcare products available in the market. The illegal access to medical data causes a huge violation to the privacy of the monitored patient, and the injection of faulty measurements may threaten the life of patient with a decision based on faulty measurements.

In this chapter, we implement the ECDHE with key renewal process to secure the communications and prevent MitM attacks while using the same security mechanisms in BLE for confidentiality and integrity. We use the Elliptic Curve Cryptography (ECC) with pre-distributed public keys used to derive the encryption key. The ECC has a small key size compared to RSA, where a 384 bits key is equivalent to 3072 bits in RSA [6]. Elliptic curve is more convenient for IoMT devices with constrained resources, where its usage is limited to derive a shared key using ECDH. The AES-CCM implemented in the BLE standard is used in our approach to provide encryption and integrity, and to prevent the MitM from conducting eavesdropping or injection attacks.

The IoMT devices are susceptible to various exploits and an attacker can easily change the behavior of compromised devices to increase the transmission rate and flood the LPU. Such change increases the energy consumption of the compromised devices and the LPU and threatens the functioning of the network. There is a need of a suitable system to detect such intrusion and to alert the user. We applied the sequential change point detection algorithm PELT [7] and the box-and-whisker plot on the number of received packets by the LPU to detect such changes and raise a network alert for user.

The rest of this chapter is organized as follows. In Sect. 2, we review recent related work. Section 3 presents our proposed approach for securing the communication link between the devices and to detect anomaly in the physiological parameters and in the number of received packets. In Sect. 4, we present our experimental results from the application of our proposed framework on real physiological data. Finally, Sect. 5 concludes the chapter and presents our future work.

## 2 Related Work

Despite the security measures adopted in BLE, some attacks are still feasible up to date. They range from simple passive data interception to identity theft and Denial of Service (DoS). Pallavi et al. in [8] review feasible security attacks on IoT devices with BLE transmission technology. Sevier et al. in [9] highlighted BLE vulnerabilities and proved that TK is vulnerable and showed how to sniff and decrypt acquired BLE data. They used Ubertooth dongle to capture BLE packets and to obtain the signal strength of the different channel frequencies. As this dongle is able to capture exchanged packets in the handshake, the TK could be cracked using the Crackle software on the Ubertooth data capture. Therefore, the LTK can be derived from the TK [10], and Wireshark can be used to decrypt the BLE packets when the LTK is provided. As Ubertooth outputs PCAP file, the sniffer Wireshark can read it and decrypt the packets in an automatic manner.

Lounis et al. in [11] confirm the results of Sevier et al. in [9]. Using the "Just Works" pairing mode, they demonstrated its weakness by showing how to generate keys. Moreover, simple technologies have been used for conducting the sniffing attack. Data from smart deadbolt, bike lock, and a lightbulb have been captured and decrypted in their experiments. However, the "Just Works" pairing method is not secure enough to generate a TK.

Cominelli et al. in [12] presented an open-source sniffer based on a Software-Defined Radio framework to capture BLE data packets in a very simple manner. They used the Graphic Processor Unit (GPU) to process the traffic. Even though sniffing can be dangerous for sensitive medical data, the attacker can induce a Denial of Service (DoS) or even spoof a device.

Therefore, the IoMT are vulnerable to various attacks as the data is transmitted using BLE wireless technology from the sensor to the LPU [13]. An adversary can modify, eavesdrop, or delete the data [14]. The impact of such attacks has been highlighted on insulin pumps with over dosage to kill the patient, and on pacemaker [15] to threaten the patient's life.

The work of Lahmadi et al. in [16] demonstrated a MitM attack against BLE and showed the low security features and inherent vulnerabilities. Afterward, they compared two unsupervised learning techniques to detect suspicious data, followed by classification method to tag packets as normal or attack from suspicious measurements. Their work is very near in his spirit to our work, where they combined supervised and unsupervised techniques to detect anomaly. However, the supervised classification requires labelled training data, which is not easy to find or to build. It is interesting to propose a lightweight and reliable sequential and non-parametric approach to prevent passive and active attacks conducted by MitM.

Aghili et al. in [17] proposed a lightweight multi-factor authentication protocol for e-health systems in IoMT. Ayub et al. in [18] proposed a secure authenticated key agreement protocol using the concepts of Physically Unclonable Function (PUF). Other research work focused on authentication, encryption, integrity, and intrusion detection to secure the network of IoMT devices [2]. However, most of the proposed

solutions have higher computation complexity which prevents their deployment on the constrained resources in IoMT devices.

Gulen et al. in [19] implemented ECC on the MSP430 micro-controller, which is commonly used in wireless sensor devices to secure wireless transmissions. Their implementation combined number transformation and elliptic curves to reduce the processing complexity. However, the implementation of other elliptic curves with more efficient formulas for key derivation is required to evaluate the complexity of such techniques.

To overcome these problems, Ahmed et al. in [20] proposed an enhanced ECDH for securing the data exchange of IoT applications. Our approach is similar in the spirit to their approach, where we use the Ephemeral ECDH to derive a session key for securing the data exchange of IoMT devices in "Just Works" pairing mode. The use of ephemeral keys allows key renewal in every time period.

On the other hand, the IoMT raises an alarm when a healthcare emergency is detected. Change Point Detection (CPD) algorithms seek to detect abrupt changes in the monitored physiological parameters, such as detecting changes in SpO2 to identify severe hypoxia or patient with COVID-19, or detecting changes in Blood Pressure (BP) to subsequently identify hypertension after vaccine. These changes need to be identified automatically with the large amount of collected data.

Several approaches for identifying changes in monitored data have been proposed in the literature [21]. The most common methods are those based on segmentation. These methods identify one or more points in a dataset where the statistical properties (e.g., mean and variance), change over time, based on the likelihood of the data in the time series. Among the proposed segmentation methods [21–23], window-based change point detection, Binary Segmentation (BS), and Optimal Partitioning (OP).

BS [23] is a sequential approach with a computational complexity $O(nlogn)$ where $n$ is the number of samples in the segment. The principle of this method is to detect a change point in the time series, and to subdivide it into two parts, where the first is before the change and the second is after the change. The operation is repeated on the two resulting parts. BS is fast and seeks to identify the minimum number of change points.

Window-based change point detection is used to perform rapid segmentation of the signal. The algorithm uses two windows that slide along the data stream. The statistical properties of the signals in each window are compared to measure the deviation. Window Segmentation (WS) has low complexity $O(nw)$ where $n$ and $w$ are the number of sample and the size of window, respectively. However, it does not produce optimal segmentations [22]. The OP method has higher computational complexity $O(n^2)$ when compared to the previous two methods (BS & WS) but is able to find the exact global optimum.

Killick et al. in [24] improved the OP by proposing a new approach to search for change points. Their proposed approach is the PELT [7], which is an efficient approximate search method able to detect all change points with respect to the change of the mean or the variance, and regardless of the statistical distribution of the time series. Its basic idea is to divide the time series into several segments where the

average of each segment is significantly different from the previous and subsequent segments. The penalty is an adjustable parameter in PELT to control the number of detected change points.

The PELT has several advantages compared to other methods, especially in terms of linear computational complexity $O(n)$ as it uses dynamic programming and pruning [7]. Yeung et al. in [25] used the PELT method to analyze public feelings towards personal masks during the COVID-19 period using Twitter data. Valdez et al. in [26] exploited PELT to identify significant changes in the volume and feeling of tweets to obtain mental health information in the USA during COVID-19 pandemic. The detection of such changes has a significant implication to trigger mitigation efforts.

Several previous work [21, 22] devoted to the search for the most adequate strategy to segment the data and compare many CPD algorithms. Their results proved that PELT provides the best tradeoff between complexity and detection accuracy, where it has the lower complexity and memory requirements when compared to other methods. This is why we will use PELT in our approach for CPD in the measurements to detect healthcare emergency, and in the number of received packets to detect compromised sensors with a high transmission rate, which intends to flood the LPU and deplete the energy.

## 3 Proposed Approach

Most IoMT devices do not have I/O capabilities and the "Just works" with the default pin code is used. To secure the communication links between devices and the LPU and to prevent attacks conducted by MitM (as shown in Fig. 1), which is able to intercept and alter the data, our proposed approach is based on pre-distributed ECC keys before deployment. These small size pre-distributed keys are used to derive a shared session key to encrypt the communication between devices and LPU using the AES-CCM deployed in BLE.

The creation of asymmetric keys is based on modern public key ECC, which is based on mathematical elliptic curves known to produce a smaller key size than RSA. The reduced key size makes the encryption operation faster and reduces the processing complexity. Let $F$ be a field with $N$ elements, $E$ is an elliptic curve with


**Fig. 1** MitM attacks against IoMT

a set of points $(x, y)$, and $G$ is the identity or the neutral element of the curve. $E$ is a function known as the Weierstrass Equation (given in Eq. 1) defined over the field $F$:

$$(E) \; : \quad y^2 + a_1xy \; + \; a_3y = x^3 + \; a_2x^2 \; + \; a_4x \; + a_6 \tag{1}$$

The coefficients $a_1, a_2, a_3, a_4, a_6 \in F$ have real values. A curve of the Weierstrass equation is said to be smooth if the partial derivatives in $x$ and $y$ of the Eq. 2 do not cancel each other at the same time instant.

$$f(x, y) \; = y^2 + \; a_1xy \; + \; a_3y - x^3 - \; a_2x^2 \; - \; a_4x \; - a_6 \tag{2}$$

For their use in cryptography, a simplification of the Eq. 1 is given in Eq. 3:

$$y^2 = \; x^3 - \; ax \; + \; b \quad \text{with} \quad 4a^4 \; + \; 27b^2 \neq 0 \tag{3}$$

To create an asymmetric key pair $(P, K)$, we used openSSL with P-384 (secp384r1) to derive the 384-bit key pair, where $P_i$ is used to denote the public key, which results from ECC point multiplication of $G$ with the private key $(\eta_i)$:

$$P_i = \eta_i * G \tag{4}$$

The operator "*" is used to denote ECC point multiplication. With a pre-distributed key, the use of ECDH mechanism does not require any exchange between the two devices to derive the shared symmetric encryption key, as shown in Fig. 2 and in Eq. 5. In Fig. 2, $P_L$ denotes the public key of the LPU, $P_D$ denotes the public key of the IoMT device, and $S_k$ denotes the derived shared key.

$$S_k = \eta_i * P_j \qquad \text{with} \qquad i \neq j \tag{5}$$

where $S_k$ is the secret key used to guarantee the security of exchanged data, and $P_j$ is the public key of the other device. However, the derived secret key is always the same. To renew the key in our approach, the LPU starts by deriving an ephemeral ECC key pair $(\eta_E, P_E)$ for each IoMT device, and transmits the public key (digitally signed) to the device to derive the same ephemeral secret key (as shown in Fig. 3),

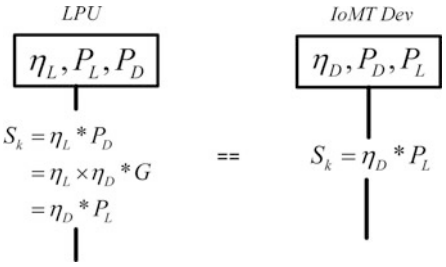


**Fig. 2** Elliptic curve Diffie-Hellman (ECDH)

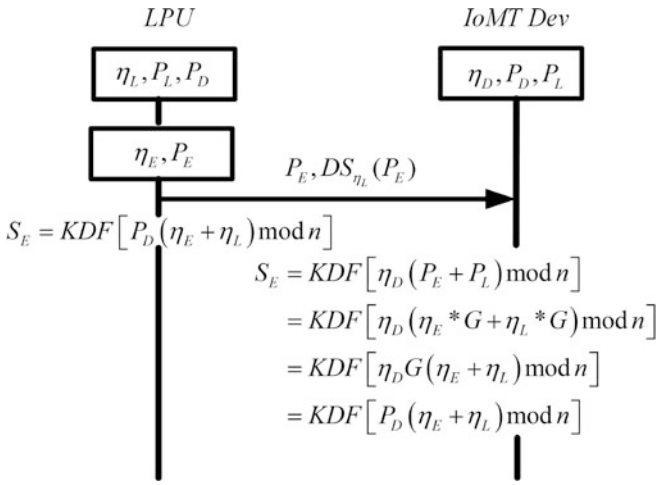


**Fig. 3** Ephemeral ECDH

which will change every period of time $T_k$. In Fig. 3, the Key Derivation Function is denoted by KDF, and the function $DS_{\eta_L}(P_E)$ is used to denote the Digital Signature (DS) of ephemeral key $P_E$.

The confidentiality and integrity of the exchanged data are provided by AES-CCM to avoid the MitM from accessing the content or modifying the values of measurements. To prevent data suppression by the MitM, the transmission is reliable and must be acknowledged (ACK) in both directions to avoid black hole attack. In the case where the IoMT device does not receive an ACK after 3 retransmissions for $k$ consecutive packets, it raises a local alert (light or sound) to notify user with a network or security problem.

To detect anomaly in acquired vital signs, we start by preprocessing the data over a window of measurements. Let $y_{1:n}$ denote the set of measurements during the period of time $T$, where $y_{1:n} = (y_1, \ldots, y_n)$ is a set of $n$ physiological measurements with real values. The CPD algorithm is able to identify $m$ changes along with their positions $t_{1:m} = (t_1, \ldots, t_m)$. The position of the change point is an integer between 1 and $n$. The time series is supposed to be piecewise stationary, which means that some characteristics of the process suddenly change at unknown time instants $t_1 < t_2 < \ldots < t_m$. The data are normalized, and their values are between 0 and 1.

To detect change points, we applied the PELT method that aims to identify the instants of change in $y_{1:n}$. It is based on the OP and pruning method. The OP method aims to minimize cost:

$$\sum_{i=1}^{m+1} \left\{ C(y_{(t_{i-1}+1)}, \ldots, y_{t_i}) + \beta \right\} \tag{6}$$

where $C$ is a cost function for the $i$th segment, and $\beta$ is a penalty to prevent overfitting. Subsequently, PELT uses pruning to increase the efficiency of the OP method while ensuring that the method finds an overall minimum of the cost function. The optimal segmentation is $F(n)$:

$$F(n) = \min_{t} \left\{ \sum_{i=1}^{m+1} \left[ C\left( y_{(t_{i-1}+1)}, \ldots, y_t + \beta \right) \right] \right\} \tag{7}$$

The main idea behind the pruning is to remove these values of $t$ which can never be minima of the minimization performed in each iteration. The OP method applies recursive conditioning by starting with a first conditioning on the last change point and calculating the optimal segmentation of the data up to the change point:

$$F(n) = \min_{t} \left\{ \begin{array}{c} \min\limits_{t|t_m} \sum\limits_{i=1}^{m} \left[ C(y_{(t_{i-1}+1)}, \ldots, y_{t_i}) + \beta \right] \\ + C(y_{(t_m+1)}, \ldots, y_n) \end{array} \right\} \tag{8}$$

Using Eq. 6 to simplify the previous equation, the internal minimization is equal to $F(t_m)$ and the Eq. 8 can be re-written as:

$$F(n) = \min_{t_m} \left\{ F(t_m) + C(y_{(t_m+1)}, \ldots, y_n) \right\} \tag{9}$$

We applied the PELT on the received measurements and on the number of received packets. The CPD in the received measurements allows to detect emergency and to raise alarms for healthcare professionals, while the CPD in the total number of packets allows to detect compromised sensors with an increased transmission rate. However, the PELT method is sensitive to changes and identify all the change points with several false alarms. To increase the reliability of the system by reducing the False Alarm Rate (FAR), we apply the box-and-whiskers (boxplot) by comparing each identified change point by PELT with robust statistical parameters derived from a window of previous $w$ values in order to confirm its deviation.

Let $Y_i^w = \{y_{t-w,i}, \ldots, y_{t,i}\}$ represents the sliding window of the last $w$ values ($[DPC - w, DPC]$) for the $i$th monitored attribute. The first quartile $Q_1$ and the third quartile $Q_3$ of $Y_i^w$ are used to derive the interquartile range $\hat{\sigma} = IQR = Q_3 - Q_1$. A measurement is considered as abnormal (as shown in Fig. 4) if the following condition is satisfied:

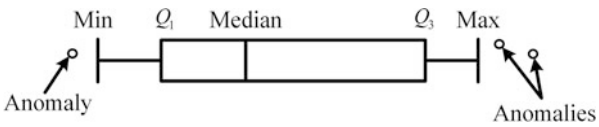


**Fig. 4** Box-and-whiskers

$$y_{t,i} \leq Q_1 - 1.5.(Q_3 - Q_1) \vee y_{t,i} \geq Q_3 + 1.5.(Q_3 - Q_1) \quad (10)$$

A medical alarm is raised if the deviation is detected only in the monitored biometric parameters and not in the number of received packets.

## 4 Experimental Results

To conduct an experiment and analyze the performance of our proposed approach, we used real physiological data collected from a patient with cardiovascular disease. The monitored patient is 68 years old, 1.75 m living independently in his apartment and kept under monitoring. The used dataset is private, collected using other prototype and stored inside a CSV file. We focus only on the chunk with changes in our experiments.

Five vital signs are available in the dataset: ABP Mean (Ambulatory BP), Heart Rate (HR), Pulse, SpO2, and Respiration Rate (RR). The measurement units are: mmHg for BP, beat per minute (bpm) for HR and Pulse, respiration per minute (rpm) for RR and % for SpO2. A value of SpO2 lower than 95% is symptomatic of asphyxia and requires ventilator and assistance. To simulate a real life scenario in Fig. 1, we used two Raspberry Pi 4B, with 8 GB of RAM and BLE as IoMT devices that read data from the CSV file and transmit records to the LPU (Android tablet) for processing. The first device transmits SpO2 and Pulse, while the second is used to transmit BP, HR, and RR.

We start our experiments by using AdaFruit USB stick (presented in Fig. 5) as BLE sniffer and Wireshark to prove the ability of MitM to access the data in the BLE pairing mode. The captured data by Wireshark sniffer in "Just works" mode is shown in Fig. 12, where the clear text value of the HR is 96 bpm. We refer to [16] and several tutorials available online to conduct such an attack using kali Linux [27].

To prevent security attacks and leakage of sensitive data, we start by implementing our approach for ephemeral key derivation from ECDH, which is used to encrypt the data. We also configure the two devices to renew the key every 10 minutes to

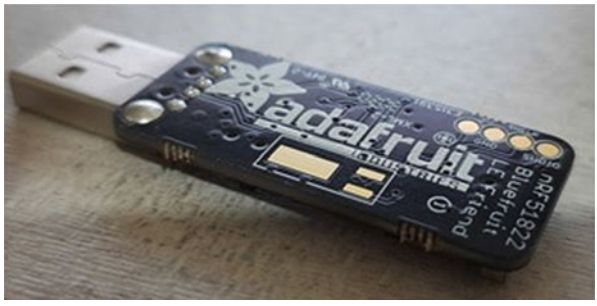


**Fig. 5** Sniffer BlueFruit

prevent off-line password guessing. The anomaly detection is implemented in the LPU and aims to identify changes in physiological and total number of received packets. The received data on the LPU from the two Raspberry devices are decrypted before processing.

The Continuous Noninvasive Atrial BP measurement (CNAP) is used to measure the BP continuously in real-time. Several CNAP monitors based on PhotoPlethysmoGraphy (PPG) are available in the market [28]. The variations of ABP Mean (denoted by BP) measurements are presented in Fig. 6, where the heavy change is visible around the time instant 18,000 sec and lasts until the end. The ABP Mean is derived from Systolic Blood Pressure (SBP) and Diastolic Blood Pressure (DBP) as given in Eq. 11:

$$ABPMean = \frac{1}{3}SBP + \frac{2}{3}DBP \tag{11}$$

Similarly, the variations of the HR and PULSE are shown in Figs. 7 and 8 where correlated changes occur at the same instant as the BP. The variations of the RR and SpO2 are presented in Figs. 9 and 10, respectively. The SpO2 falls down and becomes lower than 90% (asphyxia) at the same time instant 18,000 sec, and this explains the simultaneous increase in the number of RR and in the measurements of BP, HR, and PULSE. The patient tries to get more oxygen by increasing his respiration and making more effort. In fact, the patient needs oxygen assistant in this chunk of data.

The variations of whole physiological parameters (BP, HR, Pulse, Respiration, SpO2) are presented in Fig. 11, where we can identify a correlated change point around 18,000 sec for approximately whole parameters. The HR and PULSE superpose as they measure the same information (Fig. 12).

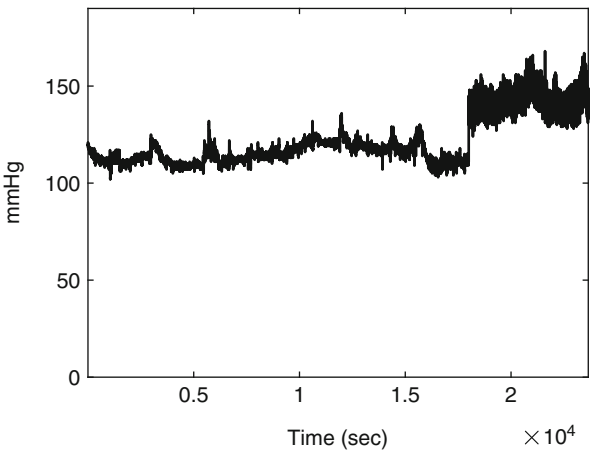


**Fig. 6** Blood pressure

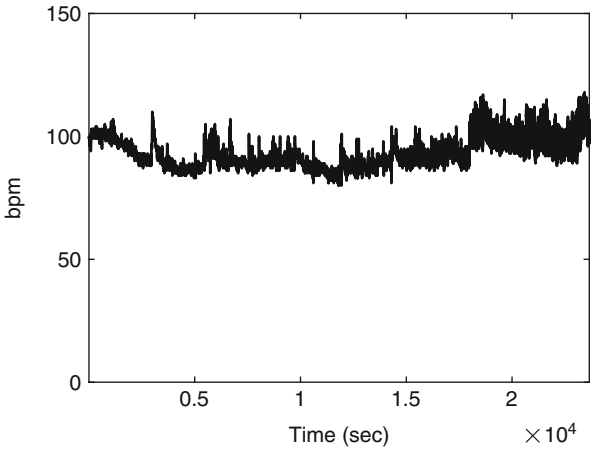


**Fig. 7** Heart rate

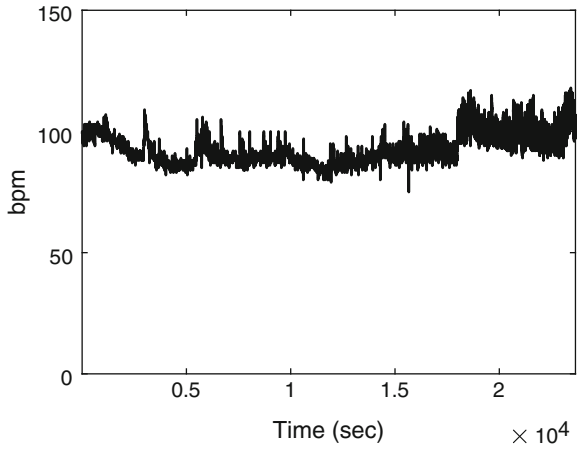


**Fig. 8** PULSE

In the second set of experiments, we start by conducting a MitM attack to capture and verify the encryption of the data. A screenshot of the captured data with Wireshark is presented in Fig. 13, where we can notice that encrypted data cannot be decoded by the sniffer. Afterward, we test the security of our approach by assuming the worst case scenario to simulate MitM attack, where an attacker successfully compromises both IoMT devices by exploiting software vulnerability. We start by increasing the transmission rate and the value of measurements for only one device in the beginning, followed by simultaneous increase in the rate of the second device (as shown in Fig. 14a) to deplete the energy of compromised sensors, and to flood the LPU with packets containing modified values. The measurements of HR in the beginning of attack can be distinguished from the Pulse as shown in Fig. 14b, where the variations are surrounded by an ellipse.

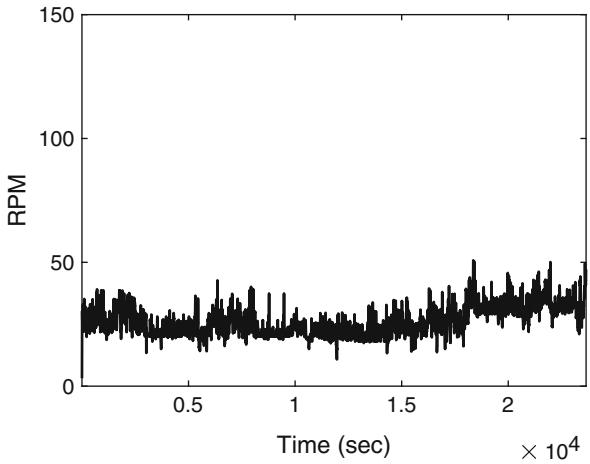


**Fig. 9** Respiration rate

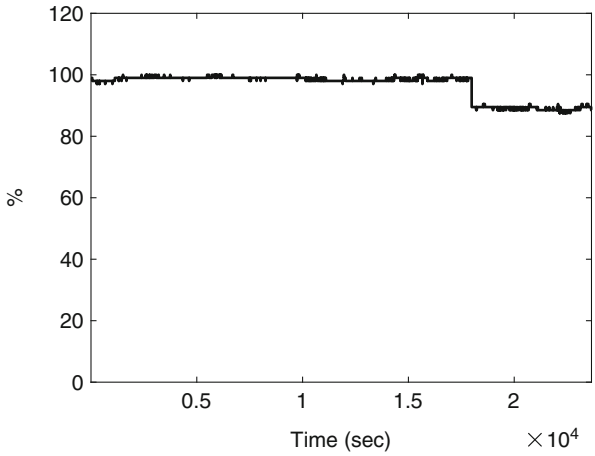


**Fig. 10** SpO2

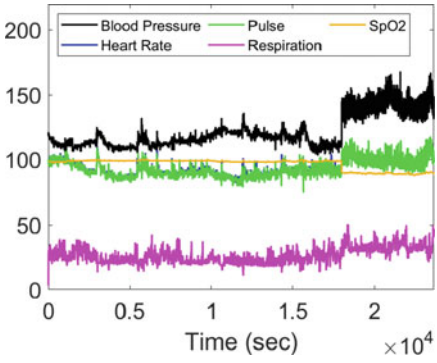


**Fig. 11** All parameters

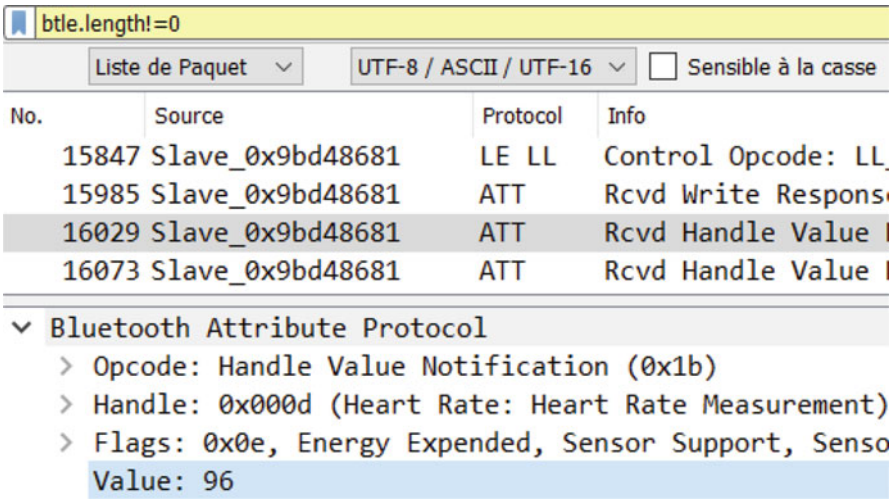


**Fig. 12** MitM: Wireshark with the value of HR

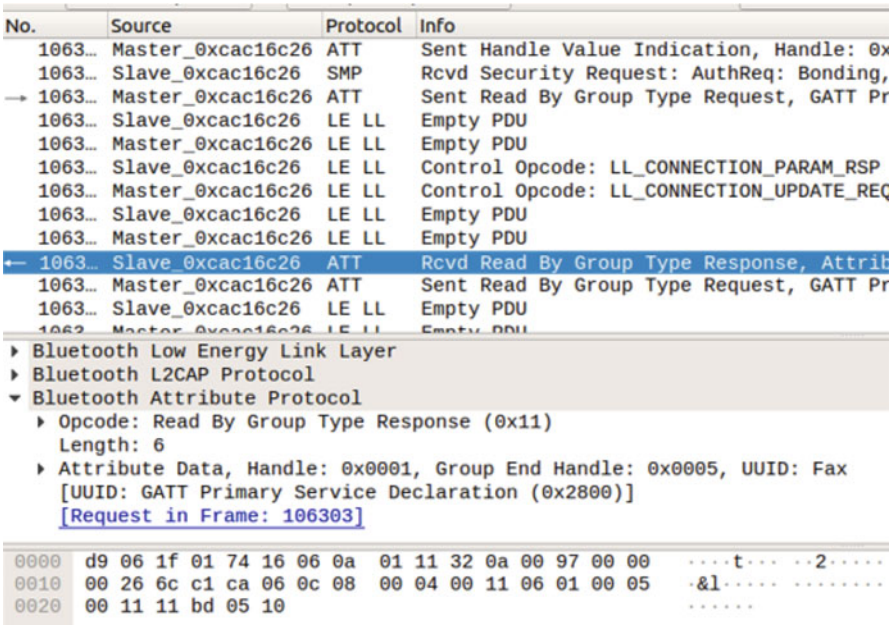


**Fig. 13** MitM: encrypted data

The average of received measurements in each second was derived and used in Fig. 14b. Our approach detects a change in the number of received packets for these variations and raises a local alert for user as a network connection alert. In such

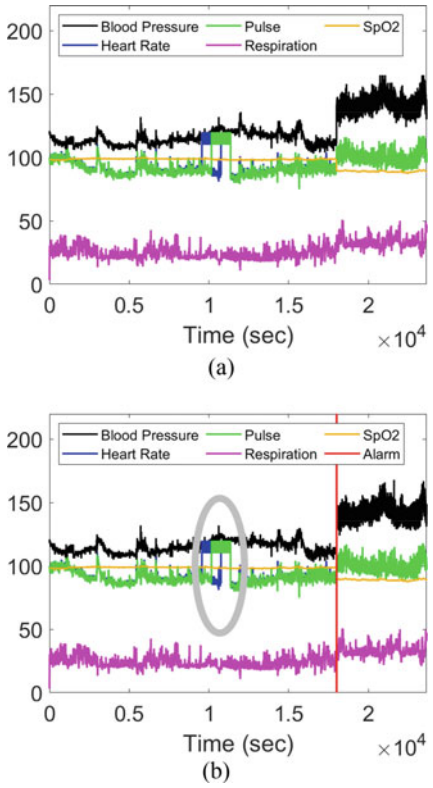


**Fig. 14** Injected measurements. (**a**) Injected values. (**b**) Normal and alarm

situation, the user must re-initialize the system to force the change of the encryption key.

The raised medical alert is represented by vertical red line in Fig. 14b and triggered only if there is no change point in the number of received packets.

In the third set of experiments, we conduct a performance comparison using the Receiver Operating Characteristic (ROC) to study the impact of the threshold on the accuracy of the system in terms of True Positive Rate (TPR) and False Alarm Rate (FAR). The TPR and FAR are given in the following equations:

$$\text{TPR} = \frac{TP}{TP + FN} \tag{12}$$

$$\text{FAR} = \frac{FP}{FP + TN} \tag{13}$$

where TP is the number of True Positives, FP is the number of False Positives, FN is the number of False Negatives, and TN is the number of True Negatives. The ROC represents the variation of TPR with respect to FAR when changing the value of the score. A value of TPR closer to 100% indicates a high detection accuracy, while a lower value of FAR is desirable to achieve to enhance the reliability of the system. However, increasing the value of TPR induces an increase of FAR, and decreasing

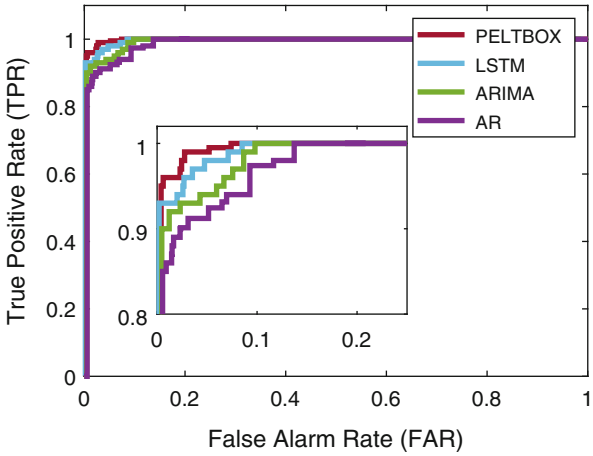


**Fig. 15** ROC

the FAR induces a reduction in TPR. Therefore, a tradeoff between TPR and FAR is required by changing the value of the decision threshold.

The ROC curve presented in Fig. 15 shows the relationship between the TPR and FAR for our proposed approach. To prove the effectiveness of our approach, we also conduct a performance comparison with existing works [29] which are based on the difference between predicted and measured values to identify changes in time series. The prediction of the current measurement was achieved using Long Short-Term Memory (LSTM), AutoRegressive Integrated Moving Average $ARIMA(p, d, q)$, and Auto Regressive $AR(p)$, with $p = 4$, $d = 1$, and $q = 2$.

The obtained ROC is presented in Fig. 15 where for a TPR of 99%, our approach has a FAR of 6%, followed by LSTM with 8%, ARIMA with 9% and AR with 12%. In fact, the use of our approach slightly outperforms the LSTM in term of FAR. On the other hand, even if the four methods have a linear computational complexity $O(n)$, our method has less execution time for processing one record than LSTM, where the decision delay of our method is 25.56 sec while the delay for LSTM is 39.63 sec, followed by ARIMA with 20.61 sec and AR with 18.48 sec.

## 5 Conclusion

In this chapter, we proposed a framework to secure the exchange of medical data in IoMT and to detect anomaly in the number of received packets and in the acquired vital signs from monitored patient. We used the ECDHE to exchange the session key in "Just Works" pairing mode, while keeping the same mechanisms used in BLE to ensure confidentiality and integrity. To detect healthcare emergency, we applied the PELT algorithm followed by boxplot to detect changes in the monitored physiological parameters with reduced FAR and low computational complexity. Furthermore, to detect attacks aiming to deplete the energy of sensors or to flood LPU, we applied the

same change point detection algorithm on the number of received packets in LPU to raise network alarms.

We conducted several experiments on data from different subjects for performance analysis and we compare the performance of our approach with previous works. Our experimental results on real physiological data showed that our approach is effective and able to achieve a good detection accuracy with a FAR of 6%. The comparison results showed that our system slightly outperforms LSTM and regression based systems. Our future work will focus on anomaly detection in the amount of energy consumed by compromised IoMT device.

## References


1. J. Fiaidhi, S. Mohammed, Security and vulnerability of extreme automation systems: the IoMT and IoA case studies. IT Professional **21**(4), 48–55 (2019)
2. G. Thamilarasu, A. Odesile, A. Hoang, An intrusion detection system for internet of medical things. IEEE Access **8**, 181560–181576 (2020)
3. G. Hatzivasilis, O. Soultatos, S. Ioannidis, C. Verikoukis, G. Demetriou, C. Tsatsoulis, Review of security and privacy for the internet of medical things (IoMT), in *15th International Conference on Distributed Computing in Sensor Systems (DCOSS)* (2019), pp. 457–464
4. D. Koutras, G. Stergiopoulos, T. Dasaklis, P. Kotzanikolaou, D. Glynos, C. Douligeris, Security in IoMT communications: a survey. Sensors **20**(17), 4828 (2020)
5. Bluetooth SIG. Bluetooth Radio Versions. https://www.bluetooth.com/learn-about-bluetooth/radio-versions/, Last visited: February 2022
6. Australian Government Autralian Cyber Security Center. Information Security Manual. https://www.cyber.gov.au/sites/default/files/2022-03/22.%20ISM%20-%20Guidelines%20for%20Cryptography%20%28March%202022%29.pdf, March 2022
7. R. Killick, I. Eckley, changepoint: an R package for changepoint analysis. J. Statist. Softw. **58**(3), 1–19 (2014)
8. S. Pallavi, V.A. Narayanan, An overview of practical attacks on BLE based IOT devices and their security, in *5th International Conference on Advanced Computing Communication Systems (ICACCS'19)* (2019), pp. 694–698
9. S. Sevier, A. Tekeoglu, Analyzing the security of bluetooth low energy, in *International Conference on Electronics, Information, and Communication (ICEIC'19)* (2019), pp. 1–5
10. K. Ren, Bluetooth Pairing Part 3 – Low Energy Legacy Pairing Passkey Entry (2016). https://www.bluetooth.com/blog/bluetooth-pairing-passkey-entry/
11. K. Lounis, M. Zulkernine, Bluetooth low energy makes "Just Works" Not Work, in *3rd Cyber Security in Networking Conference (CSNet'19)* (2019), pp. 99–106
12. M. Cominelli, P. Patras, F. Gringoli, One GPU to snoop them all: a full-band bluetooth low energy sniffer, in *Mediterranean Communication and Computer Networking Conference (MedComNet'20)* (2020), pp. 1–4
13. Wencheng Sun, Zhiping Cai, Yangyang Li, Fang Liu, Shengqun Fang, Guoyan Wang, "Security and Privacy in the Medical Internet of Things: A Review", Security and Communication Networks, vol. 2018, Article ID 5978636, 9 pages, 2018. https://doi.org/10.1155/2018/5978636
14. T. Yaqoob, H. Abbas, M. Atiquzzaman, Security vulnerabilities, attacks, countermeasures, and regulations of networked medical devices – a review. IEEE Commun. Surv. Tutor. **21**(4), 3723–3768 (2019)
15. H.A.M. Puat, N.A. Abd Rahman, IoMT: a review of pacemaker vulnerabilities and security strategy. J. Phys. Conf. Ser. **1712**(1), 012009 (2020)

16. A. Lahmadi, A. Duque, N. Heraief, J. Francq, MitM attack detection in BLE networks using reconstruction and classification machine learning techniques, in *2nd Workshop on Machine Learning for Cybersecurity (MLCS'20)* (2020), pp. 1–16
17. S.F. Aghili, H. Mala, M. Shojafar, P. Peris-Lopez, LACO: lightweight three-factor authentication, access control and ownership transfer scheme for e-health systems in IoT. Future Gener. Comput. Syst. **96**, 410–424 (2019)
18. M.F. Ayub, M.A. Saleem, I. Altaf, K. Mahmood, S. Kumari, Fuzzy extraction and PUF based three party authentication protocol using USB as mass storage device. J. Inf. Secur. Appl. **55**, 102585 (2020)
19. U. Gulen, S. Baktir, Elliptic curve cryptography for wireless sensor networks using the number theoretic transform. Sensors **20**(5), 1507 (2020)
20. M.I. Ahmed, G. Kannan, Secure end to end communications and data analytics in IoT integrated application using IBM Watson IoT platform. Wirel. Personal Commun. **120**, 1–16 (2021)
21. C. Truong, L. Oudre, N. Vayatis, Selective review of offline change point detection methods. Signal Process. **167**, 107299 (2020)
22. G.J.J. van den Burg, C.K.I. Williams, An evaluation of change point detection algorithms. arXiv, abs/2003.06222 (2020)
23. S. Kovács, H. Li, P. Bühlmann, A. Munk, Seeded binary segmentation: A general methodology for fast and optimal change point detection (2020). Preprint arXiv:2002.06633
24. R. Killick, P. Fearnhead, I.A. Eckley, Optimal detection of changepoints with a linear computational cost. J. Amer. Statist. Assoc. **107**(500), 1590–1598 (2012)
25. N. Yeung, J. Lai, J. Luo, Face off: Polarized public opinions on personal face mask usage during the covid-19 pandemic, in *IEEE International Conference on Big Data (Big Data)* (2020), pp. 4802–4810
26. D. Valdez, M. Ten Thij, K. Bathina, L.A. Rutter, J. Bollen, et al., Social media insights into us mental health during the covid-19 pandemic: longitudinal analysis of twitter data. J. Med. Int. Res. **22**(12), e21418 (2020)
27. B. Hills, Machine in the Middle (MitM) BLE Attack (2020). https://www.blackhillsinfosec.com/machine-in-the-middle-mitm-ble-attack/
28. A. Paviglianiti, V. Randazzo, S. Villata, et al. A Comparison of Deep Learning Techniques for Arterial Blood Pressure Prediction. Cognitive computation (2021). https://doi.org/10.1007/s12559-021-09910-0, DOI: 10.1007/s12559-021-09910-0, (EPUB). https://link.springer.com/content/pdf/10.1007/s12559-021-09910-0.pdf Open access paper.
29. A. Khamparia, R.H. Mondal, P. Podder, B. Bhushan, V.H.C. de Albuquerque, S. Kumar, *Computational Intelligence for Managing Pandemics*, vol. 5. (Walter de Gruyter GmbH & Co KG, Berlin, 2021)