Security enhanced Emergency Situation Detection System for Ambient Assisted Living

Placide Shabisha, Chamara Sandeepa, *Student Member, IEEE,* Charuka Moremada, *Student Member, IEEE,* Nadeeka Dissanayaka, *Student Member, IEEE,* Tharindu Gamage, *Member, IEEE,* An Braeken, Kris Steenhaut, *Member, IEEE,* and Madhusanka Liyanage, *Senior Member, IEEE,*

***Abstract*—Internet of Things (IoT) based wearable devices are used to continuously monitor the health conditions of elderly people and patients with chronic diseases. Typical wearable devices use a dedicated mobile phone as a relay node and fault in such relay nodes may lead to discontinuation of the communication link. To mitigate this challenge, we propose a novel security enhanced emergency situation detection system. The proposed system utilizes 3rd party unknown mobile relays instead of dedicated gateways. Bluetooth Low Energy (BLE) communication technology is used establish the connectivity between the wearable devices and relays. Due to the involvement of 3rd party unknown mobile relays, we also propose a key agreement and authentication scheme to ensure anonymity and untraceability for both wearable devices and relays. The proposed protocol solely relies on symmetric key based operations in order to function under resource constrained environments. Finally, a prototype of the system is developed by using commercial off-the-shelf devices to verify the validity of the proposed system and to evaluate the performance advantage over the existing systems. The proposed system's security is proved to be viable against the most common attacks.**

***Index Terms*—Internet of Things, Bluetooth Low Energy, Mobile Relays, Emergency Detection, Mobility, Symmetric key, Key agreement, Rubin-logic**

I. Introduction

Patients with chronic disease conditions and elderly people can get support from Bluetooth Low Energy (BLE) wearable sensors to keep track of their health [1]. These BLE sensors facilitate communication with the user's mobile phones. It is very important to continuously transmit the data from the sensor to the mobile, to keep track of their health conditions. But the elderly people may often forget to keep their mobile phones always charged. Also, during an emergency situation, the user's mobile phone might be turned off or might get damaged. Healthcare devices using dedicated gateways cannot transmit the data in such instance. Also, when the healthcare devices are out of range from the gateway, the continuous monitoring is halted due to the loss of connection. Further, the devices that communicate through generic BLE protocol cannot establish a connection and transmit the person's health data to a certain gateway in its proximity in case of a connection loss as this transmission would be unsecured.

As it is obvious with this kind of patient monitoring and data gathering system, the emerging concerns will be the safety and security of system users and data protection. Therefore, a specific security scheme has also been proposed and implemented along with an emergency situation detection system. Thanks to security integration in the system, mainly four features have been addressed namely confidentiality, mutual authentication, anonymity, and unlinkability. However, with the usage of the handheld devices and 3rd party mobile relays, there is a limited amount of resources available to build up such a security scheme associated with the system. Therefore, we have proposed a symmetric key-based security scheme that has been limited to the usage of XOR and one-way cryptographic hash functions as described in the upcoming sections.

Our Contribution

The paper contributes to the following:

- Proposes a protocol to successfully establish communication with a third party mobile relay and perform end-to-end secure communication with a cloud server.
- Implements a prototype of the emergency situation detection system with off-the-shelf BLE modules for the use case of heart rate monitoring by transmitting data to a remote server.
- Addresses the issues associated with real-time and stored data transmission from the IoT device.
- Discusses and experimentally analyzes the problems associated with the generic BLE communication channel implementation when transmitting data over a distance.
- Proposes a solution of handovering for relay mobiles to address the detected problems with BLE.
- Proposes a novel secure symmetric key agreement which can establish a shared common key between end devices based on freshly generated parameters.
- Security strengths of the proposed scheme are formally verified by Rubin logic and protection against security threats i.e node capturing attack, impersonation attack, man-in-the-middle attack, replay attack, online/offline dictionary attack is informally analyzed.

Placide Shabisha, An Braeken and Kris Steenhaut are with the Engineering Technology Department (INDI) and the department of Electronics and Informatics (ETRO), Vrije Universiteit Brussel (VUB), Belgium. e-mail: placide.shabisha@vub.be, an.braeken@vub.be, ksteenha@etrovub.be

Chamara Sandeepa, Charuka Moremada, Nadeeka Dissanayaka, Tharindu Gamage was with the Department of Electrical and Information Engineering, University of Ruhuna, Galle, Sri Lanka. E-mail: cpsandeepa@gmail.com, charuka.kavinda@yahoo.com, nadeekasumududissanayaka@gmail.com, tharindu@eie.ruh.ac.lk

Madhusanka Liyanage is with the Center for Wireless Communications, University of Oulu, Finland and the School of Computer Science, University College Dublin, Ireland. e-mail: madhusanka.liyanage@oulu.fi, madhusanka@ucd.ie,

The remainder of this paper is organized as follows: Section II provides the related work. Section III contains the background of our work including the associated technologies. Section IV provides a detailed description of the proposed architecture whereas information on the proposed communication protocol is explained in Section V, which acts as a core development within the overall project. Moreover, Section VI presents details on the developed security protocol and includes several sub chapters for the clarity of the content. The overall system implementation details are presented in Section VII and the performed experiments and their associated details are presented under Section VIII. A detailed security evaluation is included with informal and formal security evaluations under Section IX. The paper is completed with a discussion in Section X and a conclusion in Section XI.

II. Related Work

A. Mobile Relay Systems

The work presented in [2] [3] introduces a mobile-based relay assistance system for the establishment of a secure end-to-end (E2E) connection between low-power Internet of Things (IoT) sensors and cloud servers without using any specific and dedicated gateway. It proposes a basic prototype to accomplish the communication task and the E2E connection establishment is done through a secure AES-CCM encryption technique.

The solution in [4] contains the main server and several other sensing servers which are acting as gateways. It describes the IoT sensor network's middle-ware to perform sensor data translations. As the system is not a cost-effective solution and poorly scalable, it is not a very feasible solution for IoT applications.

The work, related with dedicated gateways presented in [5] describes the implementation of smart e-health gateways (named UT-GATE) at the edge of healthcare IoT in clinical environments. In addition to the cloud processing, they suggest local data processing through smart gateways. This step helps decreasing latency but it may be vulnerable to security problems such as the possibility of implementing malicious gateways that could eavesdrop on patient's data. Moreover, this system may not support mobility-related aspects due to cost and difficulty to provide universal connectivity (due to interoperability issues) in external environments with the proposed system.

In [6], the authors introduce an open BLE platform (custom-designed beacon platform nRF24Cheep) and open source development of the BLE physical and Medium Access Control (MAC) layer in order to provide the capabilities of changing the communication stack. The Contiki OS port is provided for the new platform.

In [7], authors propose an end-to-end security scheme for mobility enabled healthcare IoT. Their scheme has 3 main characteristics, (i) Secure and efficient end-user authentication and authorization architecture based on the certificate based DTLS handshake, (ii) Secure end-to-end communication developed on session resumption, and (iii) Robust mobility implemented through interconnected smart gateways.

In [8], a model named iConfig is proposed for managing IoT devices in smart cities. The system has an edge-driven platform that has addressed the three major issues in current IoT management being registration, configuration and maintenance.

In [9], a scheme named Collaborative on Demand Wi-Fi Sharing (COWS) is introduced, for which they propose a system to enable Wi-Fi roaming facilities for users. But this system is not fully compatible with resource-constrained devices such as those that have power limitations.

Besides [2], all the other related works use a dedicated gateway for data transmission. Moreover, the proposed solution in [2] supports a single relay for real-time data transmission only. This implies that it can fail when there is no relay nearby during an incident since critical data is not available on time or has been dropped. Therefore, a more reliable relay-based system is needed which is suggested in our work.

B. Security Schemes

Many mutual identity-based authentication schemes for a client server architecture in the context of IoT, offering anonymity have been proposed in literature. Often, the client represents a device with user interaction, being smart phone or smart card, enabling 2-factor and 3-factor authentication schemes. For the symmetric key based schemes, most of them offer anonymity and untraceability without protection against inside malicious nodes [10]. Recently in [11], another type of construction has been proposed for this aim and achieved the establishment of full anonymity and untraceability. They repaired the scheme of [12], which was not able to provide node anonymity. Their main idea is to achieve anonymity by dynamically updating the identities and to enable untraceability even if one of the nodes is captured. The main drawback of their scheme is that it is not resistant against offline dictionary attacks since guessing the identity of the sensor node can be verified by collecting the messages sent in the scheme, and if successful, results in a complete security failure of the node.

In the context of the so-called tripartite schemes, where three entities need to agree on a common key, we can also distinguish several identity-based mutual authentication schemes. Some of the schemes are based on symmetric key mechanisms, using a pre-shared common key [13]–[16]. In particular, [13], [16] study the minimum number of communication rounds and messages needed to establish mutual authentication among three different parties, taking into account different assumptions. The disadvantage in these schemes is that the session key is only constructed by the authentication server and the other two entities do not participate in the construction of it, making these schemes vulnerable for key control resilience attacks [17]. Moreover, none of the schemes provide anonymity, which was even considered impossible [18] to be reached by symmetric key-based schemes. Instead, we will provide in this paper a counterexample, showing that this statement is not correct. Our scheme relies on the basic structure of [19], but is extended to a tripartite architecture and corrects the weakness of the occurrence of an offline dictionary attack.

To further complete the state of the art with respect to authentication schemes for tripartite architecture, we also mention several public key based mechanisms for fog architectures. The role of the fog is in fact similar to the one of the mobile relay. In [20], an example of a key agreement scheme for a fog driven healthcare application is proposed in which anonymity of the end device is obtained. The scheme is an improvement of [21] in which the derived key was static and thus not able to establish perfect forward secrecy. However, it is limited to devices possessing a smart card-based entry and consists of compute intensive pairing operations. Another secure identity-based tripartite scheme is proposed in [22], which is in particular designed for mobile distributed computing environments. However, this scheme does not provide anonymity to participants and also consists of a pairing operation at the device's side. In addition, it is also not able to compute pairwise keys using the available key material at the end of the protocol. Finally, there is the scheme of [23], representing an identity-based, mutual authenticated key agreement protocol for this fog architecture, in which end device and fog are able to establish a secure communication without leakage of their identities. Only the cloud server is able to control the identities of device and fog. In addition, it has been formally proven that the session keys are also protected in the Canetti-Krawczyk security model, in which adversaries are considered to have access to session state specific information, previous session keys, or long-term private keys. The scheme is efficient compared to [20], [22] as it only utilises elliptic curve operations and basic symmetric key operations.

III. Background

This section offers background information related to Ambient Assisted Living (AAL) and Bluetooth Low Energy (BLE) technology.

A. Ambient Assisted Living (AAL)

The concept of Ambient Assisted Living (AAL) can be recognized as an area which consists of technical systems to support elderly people and people with special needs. The AAL based developments are capable of helping such people in their day-to-day routines by assuring their autonomy while improving the safety of their lives. In addition to that, the basic needs for the AAL implementations can be recognized as health, safety, peace of mind, independence, mobility and social contacts [24]. With the development of this AAL concept, several tools and technologies have been introduced to make things a reality. In particular, the following tools and technologies can be recognized as some of the leading areas of modern AAL implementations [25].

- Smart Home
- Mobile and Wearable Sensors
- Robotics

Moreover, a larger impact from these AAL based applications is expected in upcoming years because of the growing elderly population all around the world. The estimations say that the 16% of global population will be older than 65 years by the year 2050 [26]. Therefore, AAL based developments will become more critical in upcoming decades to facilitate the wellbeing of such a large population, especially in developed countries.

B. Bluetooth Low Energy (BLE)

BLE, also named Bluetooth Smart, is an emerging wireless technology and a low energy version of Bluetooth specified in version 4.0 [27]. It is a modern Bluetooth technology developed by Bluetooth Special Interest Group (SIG) and is intended to support short-range communications [28]. BLE is a single-hop solution in contrast to other low-power wireless solutions, such as ZigBee and Z-Wave. Nowadays BLE is a widely used technology, applicable in a variety of use cases such as healthcare, consumer electronics and smart energy and it is expected to be used in billions of devices in the near future [29].

BLE operates in the 2.4 GHz ISM band and it supports an outdoor maximum data rate of 1 Mb/s. The coverage range of BLE is typically over tens of meters [29]. Bluetooth SIG defines some of the standard services and characteristics. When establishing a connection, the server exposes its services and characteristics (represented by a 16-bit address format) to the client to define how the connection will be structured [30].

As specified in the Generic Access Profile (GAP) in BLE, there are two roles in a BLE device. They are:

- Broadcaster (Advertiser) periodically sends advertising packets to any device able to receive them.
- Observer (Scanner) continuously scans, at periodic intervals, if there are available advertising packets to receive from a broadcaster [30].

IV. Proposed Architecture

We design and implement an IoT based remote patient monitoring and caring system which offers maximum mobility and flexibility to its users. The proposed system is similar to a fog computing approach [31], with third party mobile relays. An overview of the system architecture is shown in Fig. 1.

As indicated in Fig. 1, the whole architecture consists of four main components. At one end of the architecture, the network consists of BLE based sensor nodes with low power consumption. This part of the system is responsible for gathering the required patient data. For the use case scenarios, we selected the heart-rate and fall detection.

The data generated from the sensors are forwarded into a 3rd party unknown mobile relay. In this case, the BLE sensor node selects a specific mobile relay that is in range and available before sending the information. The selection procedure is mentioned under the communication protocol section. Furthermore, each 3rd party mobile relay has a mobile application which enables and controls its connectivity with the network. At the mobile stations, there is no data processing or storing work, but the mobile can attach its location information with the data that is being transmitted, to get the approximate patient location.

In the next step, the mobile relay sends data to the cloud server via its internet connection accordingly. In this case, a secure socket communication can be established between the mobile relay and the cloud server.

The server performs data processing, data storing and emergency situations detection. After the detection of any emergency, the server sends notifications to the registered

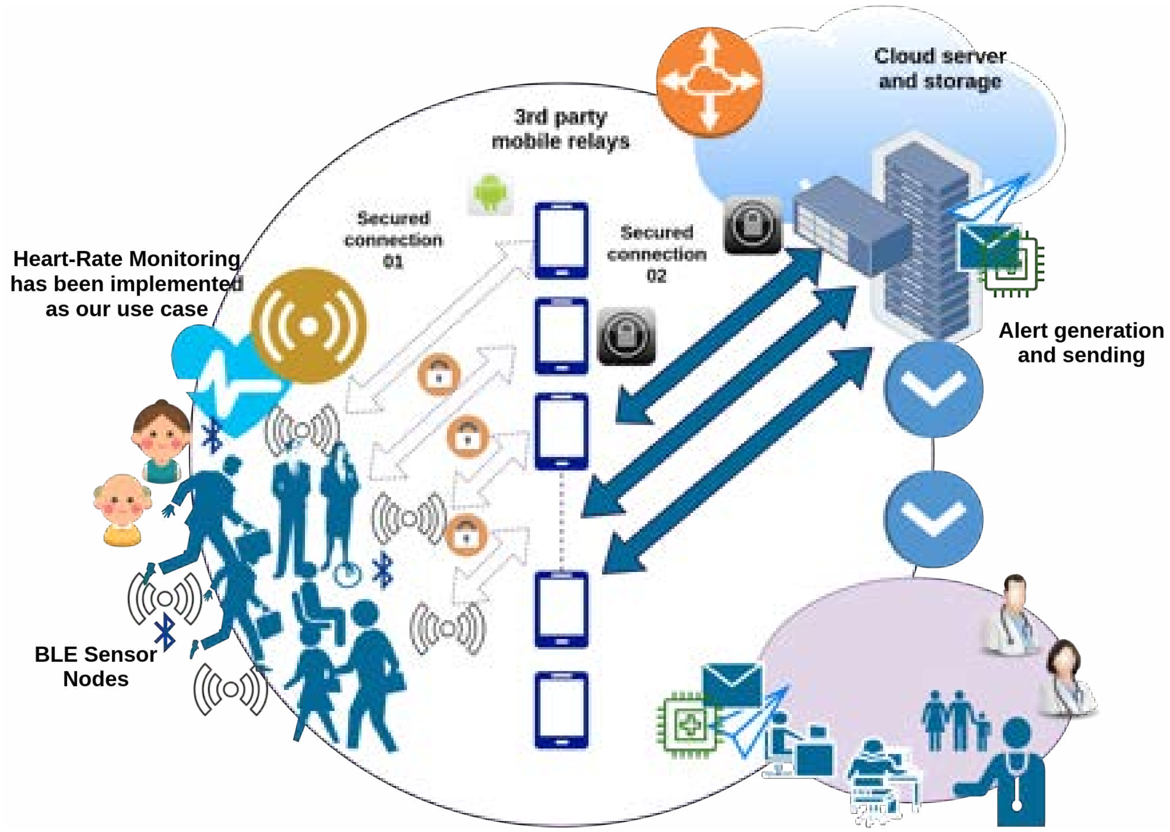


Fig. 1: Proposed system architecture

carers of the patients as SMS and emails. Other than the alerting functionality, the system also supports real-time data monitoring and location tracking services for the carers of the patient. Therefore, the patient's data is available for the carers to view via a web application.

The proposed architecture contains two key protocols, i.e. ***communication protocol*** and ***security protocol***. The communication protocol is designed to establish the communication between different entities in the proposed system. It has extended the existing solution with new capabilities such as multi-connect, automatic handovering, storage, forwarding data and load balancing. Further details are discussed in Section V. Moreover, the security protocol is designed to provide essential security features such as confidentiality, mutual authentication, unlinkability and anonymity. It also offers protection against several types of security threats i.e node capturing attack, impersonation attack, man-in-the-middle attack, replay attack and online/offline dictionary attack. Further details are discussed in Section VI.

V. Communication Protocol

A. Single mobile relay node BLE connection

The message flow of the protocol for a single mobile in the relay is depicted in Fig. 2.

1) ***Phase 1:*** The donor mobile relay node connects with the Cloud Server (CS) via a HTTPS connection request by the mobile app. In this case, the registration phase will be initiated and both BLE sensor node and the mobile relay have to be registered with the cloud server as described in Section VI-D. Moreover, a trusted mobile relay needs to be utilized in order to perform this initial registration, like the person's own smartphone. This registration consists of the establishment of the security key material. After successful registration with the CS, the IoT device can function with any other preregistered mobile relay as indicated in Fig. 2. On the other hand, a mobile can register itself as donor with the CS through an inbuilt process. Upon successful authentication, the CS issues a dynamic value a_f for the donor mobile, as discussed in Section VI-D. This value is considered as the advertisement ID and is used by the mobile to advertise its presence via BLE.
2) ***Phase 2:*** The mobile relay node starts advertising the received advertisement ID. Meanwhile, this mobile is also scanning for an advertisement from a wearable device advertising with the same ID. A wearable device which is scanning for mobile relays can get the advertisement ID from the mobile relay node and can start advertising itself with that same ID. The correctness and authenticity of this data will be evaluated after a successful run of the key agreement protocol. Therefore, a key agreement phase has to be carried out as further described in Section VI-E. In case of multiple mobile relays in proximity, the wearable device can select the mobile relay with the best Received Signal Strength Indicator (RSSI) value. Then, the mobile app can establish a

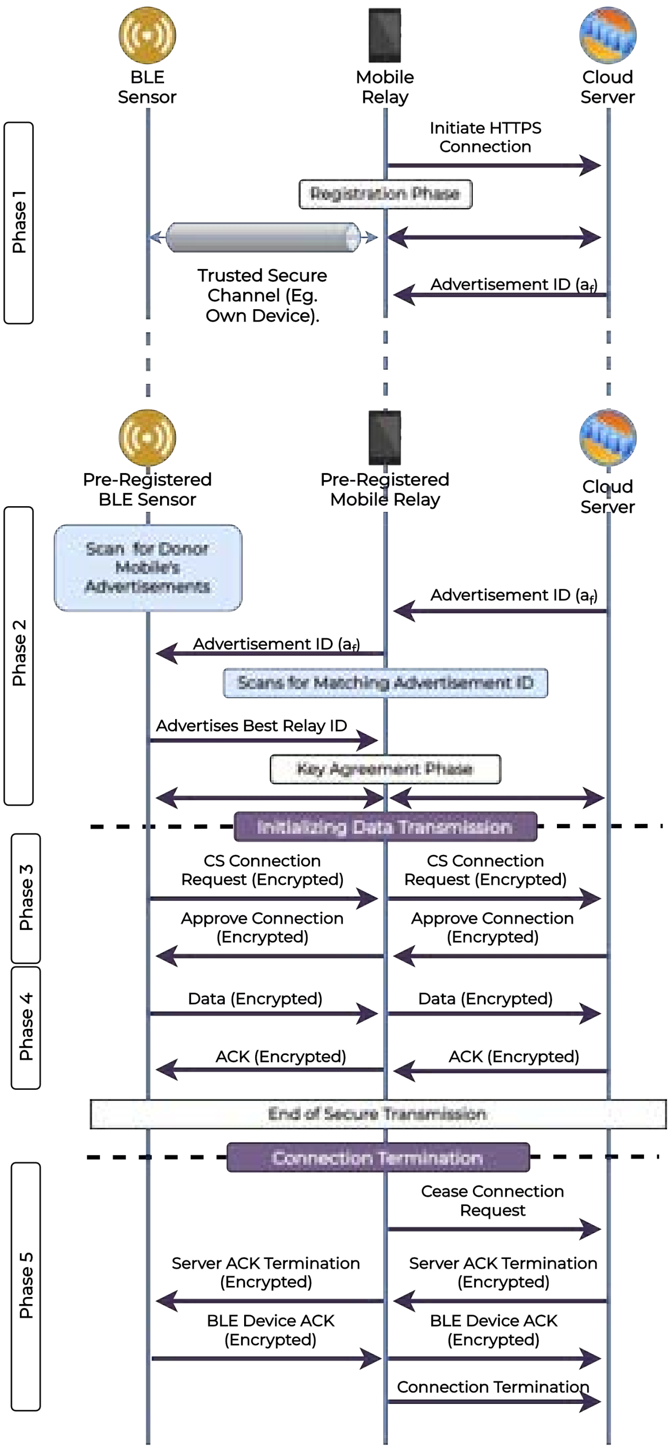

BLE Sensor
Mobile Relay
Cloud Server
Phase 1
Initiate HTTPS Connection
Registration Phase
Trusted Secure Channel (Eg. Own Device).
Advertisement ID (a_f)
Pre-Registered BLE Sensor
Pre-Registered Mobile Relay
Cloud Server
Phase 2
Scan for Donor Mobile's Advertisements
Advertisement ID (a_f)
Advertisement ID (a_f)
Scans for Matching Advertisement ID
Advertises Best Relay ID
Key Agreement Phase
Initializing Data Transmission
Phase 3
CS Connection Request (Encrypted)
CS Connection Request (Encrypted)
Approve Connection (Encrypted)
Approve Connection (Encrypted)
Phase 4
Data (Encrypted)
Data (Encrypted)
ACK (Encrypted)
ACK (Encrypted)
End of Secure Transmission
Connection Termination
Phase 5
Cease Connection Request
Server ACK Termination (Encrypted)
Server ACK Termination (Encrypted)
BLE Device ACK (Encrypted)
BLE Device ACK (Encrypted)
Connection Termination

Fig. 2: Message flow of the proposed protocol

connection with a wearable device if a match is found with its ID.

3) ***Phase 3:*** After the connection establishment with the mobile relay node, the wearable device can initiate a connection request with the CS. The mobile relay would forward this request to the CS. The server can then validate this request and approve the connection.

4) ***Phase 4:*** After the connection approval, the wearable device initiates the transmission of data to the mobile relay. These data packets may contain a timestamp and the data is encrypted so that the mobile relays cannot eavesdrop the user's data. Once a fixed amount of data is transferred, the wearable device expects an encrypted acknowledgment from the CS. If the acknowledgment is received, the data transmission is resumed. Otherwise, the wearable device terminates the connection with the relay and reports this session to the CS in the next successful data transmission.

5) ***Phase 5:*** The donor mobile can set the maximum threshold of data that a wearable device can consume, so if the threshold value is reached, it can request to cease the connection from the CS. The CS then sends the last acknowledgment message to the wearable device and the latter terminates the connection with the mobile relay. In this case, the wearable device discards the session information as this is a legitimate session termination. To send more data, the wearable device can restart from "Phase 2" and start scanning for other nearby mobile relays.

B. Wearable device Multi connect

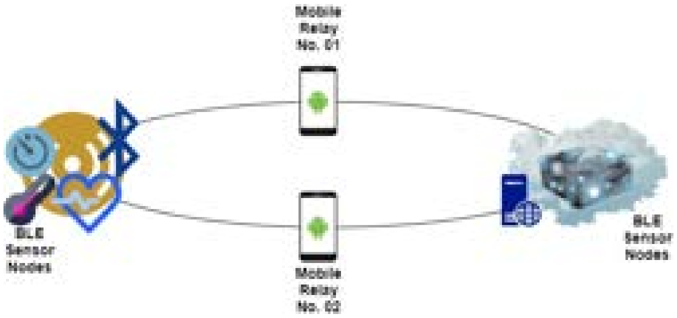


Fig. 3: Multiple mobile relay node connection

The transmission of real-time data and previously stored data together with a single mobile relay node may cause problems such as latency when sharing the same connection. Here, the stored data is the data that is generated by the wearable devices when the device is not connected to a nearby mobile relay. The paper proposes an expansion of the same protocol to transmit real-time and stored data separately by two mobile relays as indicated in Fig. 3. Note that the current prototype implementation does not include this concept since it would exceed the scope of the system implementation. However, this expansion will be considered in future work. Here, the wearable device follows a procedure such that one mobile relay is dedicated for transmitting the real-time data and another for the transmission of any stored data.

In order to achieve this, there should be at least two donor mobile devices. The real-time data has a higher priority and hence the selection of the mobile relay device for real-time data transmission is based on the best RSSI value. To transmit the stored data, the wearable device selects the second best mobile using the same criteria.

C. BLE Handover

With the implementation of this 3$^{\text{rd}}$ party relay based communication channel, we recognized the requirement of a proper handovering mechanism to keep the channel conditions at an acceptable level. Additionally, handovering will be a basic requirement to avoid unnecessary data losses or delays within a dynamic environment as well. In our system, mainly two scenarios were identified where the handovering mechanism should be implemented.

1) When the currently connected donor has decided to terminate the connection with the connected mobile relay.
2) When the patient is travelling away from the connected donor mobile (If the RSSI of the patient's device reaches a predefined threshold, the handovering should be performed).

Besides the above mentioned scenarios, there can be several other ways which lead to a connection termination. Therefore, the handovering is a must to keep a continuous data flow between the IoT device and the cloud server. It is also important to keep the real time data monitoring and emergency alert generation processes online.

With the currently developed handovering mechanism, if the ongoing connection terminates, the IoT device will scan for another mobile relay to be connected with. After detection of the mobile relay device that provides the best RSSI value, it establishes a new connection following the steps included within Sections V and VI.

VI. Security Protocol

The proposed security scheme consists of three main entities: BLE sensor nodes N, relay nodes F and cloud server CS as illustrated in Fig. 1. Both sensor nodes and relay nodes need to register with the CS and receive key material, which should be securely stored on the device. We assume that storage is tamper proof, which is currently common on state of the art devices. Also the server needs to store its master secret key in tamper resistant hardware.

If the sensor wants to communicate to the CS, it sends a request to the nearby relay, which further forwards the message to CS after having added additional information. Based on the received data, the CS verifies the authenticity of the request and if positive, it generates the required key material, allowing relay and sensor node to generate a common shared key with the server, hence all three can derive a common shared key. The relay node should not be able to derive the identity of the sensor in the whole process, and vice versa.

A. Required Security Features

To summarize, the following security features should be established.

- **Confidentiality:** Only the involved entities should be able to derive the key material.
- **Mutual authentication:** The common shared key should involve key material coming from all entities able to derive the corresponding key.
- **Anonymity:** No outsider and even not the relay node or other sensor nodes are able to derive the identity of the sensor node. Also the sensor node is not able to derive the identity of the relay node.
- **Unlinkability:** No outsider and even not the relay node or other sensor nodes are able to link messages coming from the same device. Also the sensor node is not able to link messages coming from the same relay node.

B. Attack model

In the attack model, we assume that the adversary is able to eavesdrop on the channel or actively manipulate the transmitted messages, i.e. insert, change, reply messages. These activities are typically applied when trying impersonation, man-in-the-middle (MITM) attacks, replay attacks and online/offline dictionary attacks. An attacker is also able to capture a node or relay node and to derive the key material stored in the tamper proof part of the memory. In this case, it is important to keep the impact of the attack local to the tampered device.

C. Proposed Scheme

To establish the above security features, we design the security scheme for the proposed architecture. The scheme consists of two phases: ***registration phase*** and ***key agreement phase***.

The operations in the scheme are limited to XOR $\oplus$ and a one-way cryptographic hash function H (eg. SHA2 or SHA3). Since the proposed scheme uses very low-cost cryptographic operations, it is efficient in terms of computation. Furthermore, the concatenation of two messages is denoted by $m_1 \| m_2$.

D. Registration phase

In the registration phase, both wearable devices and mobile relay need to register with the CS. During this operation, CS makes use of its master key k_m. The process for both is shown in Fig. 4.

The CS chooses a temporary key k_i and derives the following parameters for the entity with identity id_i. Here $i = n$ in case of the wearable device N and $i = f$ in case of the Relay F.

$$\begin{aligned} a_i &= id_i \oplus H(k_m \| k_i) \\ b_i &= a_i \oplus k_m \oplus k_i \\ c_i &= H(id_i \| k_m) \end{aligned}$$

The parameters (id_n, a_n, b_n, c_n) and (id_f, a_f, b_f, c_f) are sent over a secure channel to the wearable device and mobile relay respectively. Note that the temporary keys k_n and k_f are not stored, neither by the wearable devices, nor by the mobile relay, or by the CS. The parameters id_i, c_i, corresponding with the static identity, require secure storage in tamper resistant hardware at the wearable device and mobile relay as these are fixed parameters and do not vary over time. The parameters $\{a_i, b_i\}$ represent the dynamic identity, are publicly available as they appear in the message sent by the device, and are updated in each communication phase.

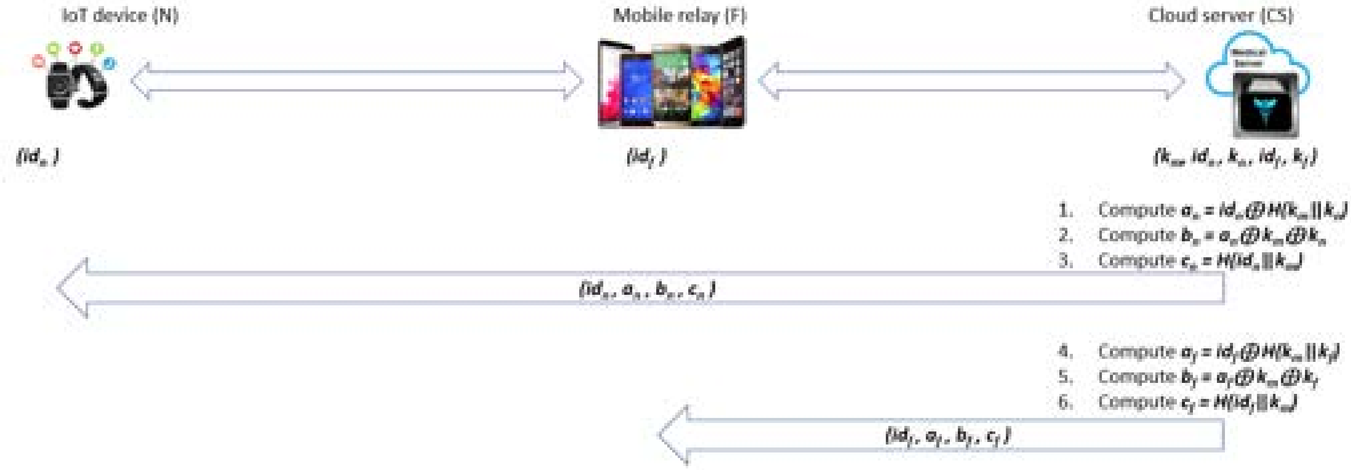

IoT device (N)
Mobile relay (F)
Cloud server (CS)
Medical Server
(id_n)
(id_f)
$(k_m, id_n, k_n, id_f, k_f)$
1. Compute $a_n = id_n \oplus H(k_m \| k_n)$
2. Compute $b_n = a_n \oplus k_m \oplus k_n$
3. Compute $c_n = H(id_n \| k_m)$
(id_n, a_n, b_n, c_n)
4. Compute $a_f = id_f \oplus H(k_m \| k_f)$
5. Compute $b_f = a_f \oplus k_m \oplus k_f$
6. Compute $c_f = H(id_f \| k_m)$
(id_f, a_f, b_f, c_f)

Fig. 4: The Registration phase

Note that at CS side, no specific storage is required for each entity, except the master key k_m. If the CS still wants to keep track of which devices are registered, it should not store the identities, but the hashed values of the identities. It is important that the identities are in no circumstance leaked, otherwise the security of the corresponding device is not guaranteed anymore, as the device is sufficiently weakened but not broken. The temporary keys k_n, k_f should not be stored and must be deleted by the CS.

E. Key agreement phase

In this phase, we distinguish five different steps, where the first four steps correspond with a message sent over the channel. Fig. 5 shows the key agreement phase in detail.

1) Wearable device request

In this step, the wearable device broadcasts a key agreement request. Therefore, it needs to choose a random value r_n and computes, using its stored parameters (id_n, a_n, b_n, c_n) and the current timestamp t_n, the following values:

$$\begin{array}{rcl} x_n & = & a_n \oplus id_n \\ y_n & = & x_n \oplus r_n \\ tid_n & = & H(id_n \| t_n \| c_n \| r_n) \end{array}$$

The message $m_1 = \{tid_n, y_n, a_n, b_n, t_n\}$ is sent by the wearable device.

2) Mobile relay request

The mobile relay, which picks up this message, also derives in the same way the following values using its stored parameters (id_f, a_f, b_f, c_f), the received timestamp t_n and a randomly chosen parameter r_f:

$$\begin{array}{rcl} x_f & = & a_f \oplus id_f \\ y_f & = & x_f \oplus r_f \\ tid_f & = & H(id_f \| tid_n \| t_n \| c_f \| r_f) \end{array}$$

Next, the message $m_2 = \{tid_n, y_n, a_n, b_n, t_n, tid_f, y_f, a_f, b_f\}$ is sent to CS.

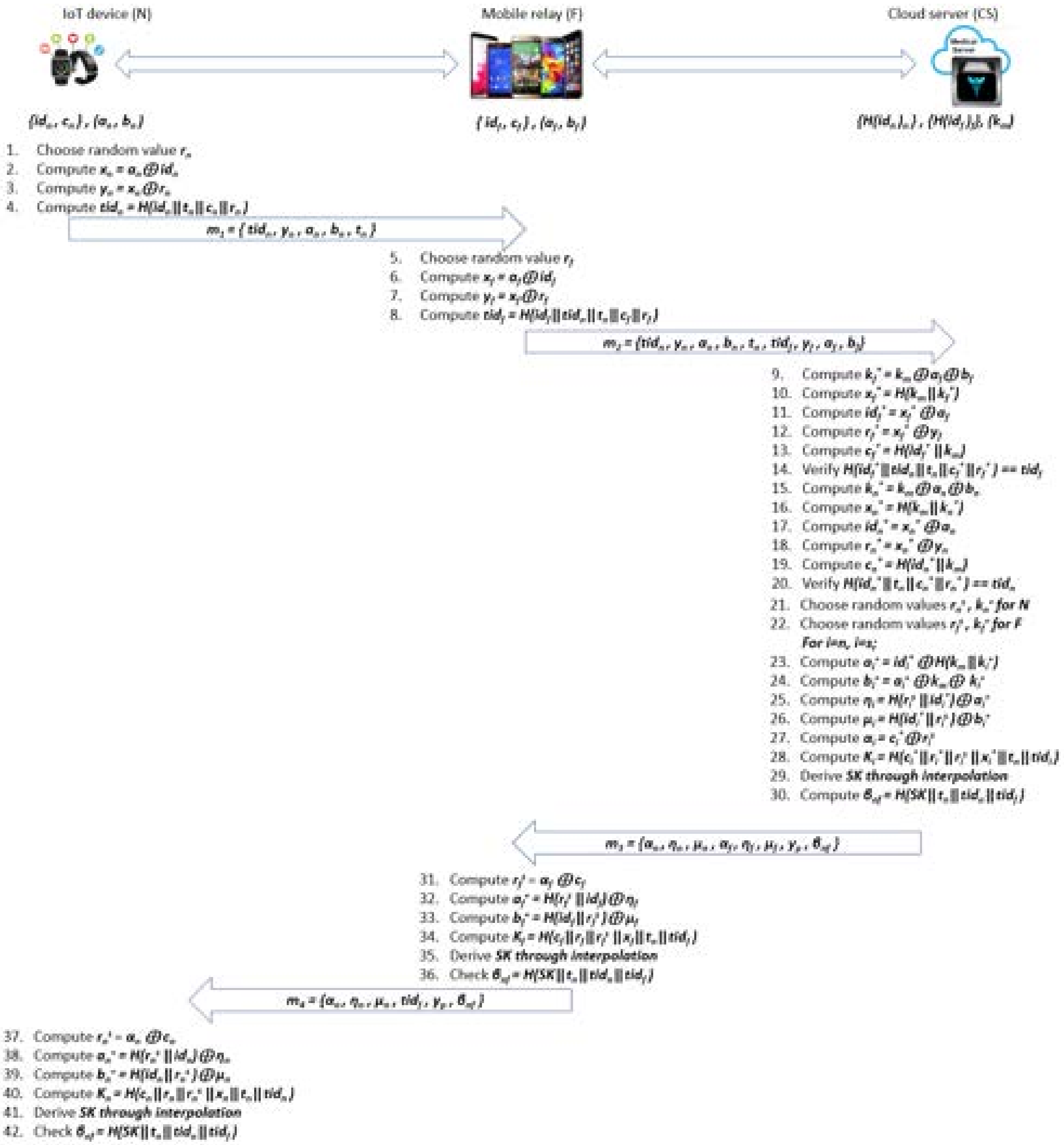
3) Cloud server check and key establishment

Upon arrival of this message, CS verifies the correctness of the message by first deriving the identities of the wearable device and the relay, and then by checking if the message is well formed. This results in the following operations:

$$\begin{array}{rcl} k_f^* & = & k_m \oplus a_f \oplus b_f \\ x_f^* & = & H(k_m \| k_f^*) \\ id_f^* & = & x_f^* \oplus a_f \\ r_f^* & = & x_f^* \oplus y_f \\ c_f^* & = & H(id_f^* \| k_m) \\ \text{Verify:} & & H(id_f^* \| tid_n \| t_n \| c_f^* \| r_f^*) == tid_f \\ k_n^* & = & k_m \oplus a_n \oplus b_n \\ x_n^* & = & H(k_m \| k_n^*) \\ id_n^* & = & x_n^* \oplus a_n \\ r_n^* & = & x_n^* \oplus y_n \\ c_n^* & = & H(id_n^* \| k_m) \\ \text{Verify:} & & H(id_n^* \| t_n \| c_n^* \| r_n^*) == tid_n \end{array}$$

At this stage, the correctness of the identities of the mobile relay and the wearable device is verified. CS now creates new dynamic identities for the mobile relay and the wearable device and then derives a session key, using the random values chosen by the mobile relay and the wearable device. Therefore, it chooses two random values r_n^s, k_n^+ related to the wearable device and two random values r_f^s, k_f^+ related to the mobile relay. Next, it computes the new identity related material a_i^+, b_i^+ and session key K_i with $i = \{n, f\}$.

$$\begin{array}{rcl} a_i^+ & = & id_i^* \oplus H(k_m \| k_i^+) \\ b_i^+ & = & a_i^+ \oplus k_m \oplus k_i^+ \\ \eta_i & = & H(r_i^s \| id_i^*) \ \oplus a_i^+ \\ \mu_i & = & H(id_i^* \| r_i^s) \ \oplus b_i^+ \\ \alpha_i & = & c_i^* \oplus r_i^s \\ K_i & = & H(c_i^* \| r_i^* \| r_i^s \| x_i^* \| t_n \| tid_i) \end{array}$$


IoT device (N)
Mobile relay (F)
Cloud server (CS)
{id_n, c_n} , {a_n, b_n}
{ id_f, c_f} , {a_f, b_f}
{$H(id_n)_n$} , {$H(id_f)_f$}, {k_m}

1. Choose random value r_n
2. Compute $x_n = a_n \oplus id_n$
3. Compute $y_n = x_n \oplus r_n$
4. Compute $tid_n = H(id_n \| t_n \| c_n \| r_n)$

$m_1 = \{tid_n, y_n, a_n, b_n, t_n\}$

5. Choose random value r_f
6. Compute $x_f = a_f \oplus id_f$
7. Compute $y_f = x_f \oplus r_f$
8. Compute $tid_f = H(id_f \| tid_n \| t_n \| c_f \| r_f)$

$m_2 = \{tid_n, y_n, a_n, b_n, t_n, tid_f, y_f, a_f, b_f\}$

9. Compute $k_f^* = k_m \oplus a_f \oplus b_f$
10. Compute $x_f^* = H(k_m \| k_f^*)$
11. Compute $id_f^* = x_f^* \oplus a_f$
12. Compute $r_f^* = x_f^* \oplus y_f$
13. Compute $c_f^* = H(id_f^* \| k_m)$
14. Verify $H(id_f^* \| tid_n \| t_n \| c_f^* \| r_f^*) == tid_f$
15. Compute $k_n^* = k_m \oplus a_n \oplus b_n$
16. Compute $x_n^* = H(k_m \| k_n^*)$
17. Compute $id_n^* = x_n^* \oplus a_n$
18. Compute $r_n^* = x_n^* \oplus y_n$
19. Compute $c_n^* = H(id_n^* \| k_m)$
20. Verify $H(id_n^* \| t_n \| c_n^* \| r_n^*) == tid_n$
21. Choose random values r_n^s, k_n^+ *for N*
22. Choose random values r_f^s, k_f^+ *for F*

For i=n, i=s;

23. Compute $a_i^+ = id_i^* \oplus H(k_m \| k_i^+)$
24. Compute $b_i^+ = a_i^+ \oplus k_m \oplus k_i^+$
25. Compute $\eta_i = H(r_i^s \| id_i^*) \oplus a_i^+$
26. Compute $\mu_i = H(id_i^* \| r_i^s) \oplus b_i^+$
27. Compute $\alpha_i = c_i^* \oplus r_i^s$
28. Compute $K_i = H(c_i^* \| r_i^* \| r_i^s \| x_i^* \| t_n \| tid_i)$
29. Derive ***SK through interpolation***
30. Compute $\beta_{nf} = H(SK \| t_n \| tid_n \| tid_f)$

$m_3 = \{\alpha_n, \eta_n, \mu_n, \alpha_f, \eta_f, \mu_f, y_p, \beta_{nf}\}$

31. Compute $r_f^s = \alpha_f \oplus c_f$
32. Compute $a_f^+ = H(r_f^s \| id_f) \oplus \eta_f$
33. Compute $b_f^+ = H(id_f \| r_f^s) \oplus \mu_f$
34. Compute $K_f = H(c_f \| r_f \| r_f^s \| x_f \| t_n \| tid_f)$
35. Derive ***SK through interpolation***
36. Check $\beta_{nf} = H(SK \| t_n \| tid_n \| tid_f)$

$m_4 = \{\alpha_n, \eta_n, \mu_n, tid_f, y_p, \beta_{nf}\}$

37. Compute $r_n^s = \alpha_n \oplus c_n$
38. Compute $a_n^+ = H(r_n^s \| id_n) \oplus \eta_n$
39. Compute $b_n^+ = H(id_n \| r_n^s) \oplus \mu_n$
40. Compute $K_n = H(c_n \| r_n \| r_n^s \| x_n \| t_n \| tid_n)$
41. Derive ***SK through interpolation***
42. Check $\beta_{nf} = H(SK \| t_n \| tid_n \| tid_f)$

Fig. 5: Key Agreement phase

In order to derive a common shared key between all three, CS determines the line containing the points $(1, K_n), (2, K_f)$. The intersection with the Y-axis, i.e. $x = 0$ results in the secret key SK. The Cloud server also derives the point on the line for $x = 3$, resulting in $y = y_p$. In order to allow the mobile relay and the wearable device to derive the correctness of the message, the following parameter is also computed:

$$\beta_{nf} \quad = \quad H(SK \| t_n \| tid_n \| tid_f)$$

The message $m_3 = \{\alpha_n, \eta_n, \mu_n, \alpha_f, \eta_f, \mu_f, y_p, \beta_{nf}\}$ is sent to the mobile relay.

4) Mobile relay key derivation

The mobile relay considers the last 5 parts of the received message. Using α_f, it can find r_f^s. As a consequence, the new dynamic identity (a_f^+, b_f^+) is derived using this value r_f^s and the received parameters η_f, μ_f. Next, the common shared key K_f with CS can be computed. In order to also compute the common shared key between all three entities, it derives the line through the points $(3, y_p), (2, K_f)$ and computes the intersection with the Y-axis to find SK. Finally, the correctness is verified by checking the hash value to derive β_{nf}. If this last check is successful, the mobile relay forwards to the wearable device the message

$$m_4 = \{\alpha_n, \eta_n, \mu_n, tid_f, y_p, \beta_{nf}\}$$

5) Wearable device key derivation

The wearable device can perform similar steps as the mobile relay in order to derive r_n^s, its new dynamic identity (a_n^+, b_n^+) and its common shared key K_n with CS. In the same way, the common shared key SK is determined and its validity is checked by means of β_{nf}.

VII. Implementation

In order to implement the protocol, the selected use case scenario is the emergency situation detection system for people with heart conditions. This involves sensors that can detect heart problems, a BLE device that can transmit the data, a relay mobile that supports BLE and a cloud server that performs operations such as user registration, authentication, storage, detection of emergency situations and generation of alerts. The overall system was named as "The Healer" by the authors.

In order to perform some of the experiments mentioned in Section VIII, we prepared a specific setup as shown on Fig. 6.

A micro-controller unit named ESP32, which is powered with BLE 4.2, is used to establish the communication pathway. Table I provides details on the ESP32 unit. The mobile phones used for the implementation were Samsung Galaxy M20 and Samsung Galaxy A20 with Android 9 Pie system and OPPO A37 running Android 5.1 Lollipop. The Android Studio 3 libraries were used for the mobile application. For the server-side implementation, Java Spring Boot framework is used. The database was implemented with PostgreSQL and for the time-series data storage, TimescaleDB extension of the PostgreSQL is used. The heart rate is obtained from a heart rate sensor model named MAX30100.

During a communication failure or while waiting for the BLE sensor to establish the connection with the mobile relay, the data that is captured by the sensors are stored in an external flash memory connected to the ESP32. This flash memory device can also be used to store the data during the handovering process (time interval between the current connection termination and new connection establishment). Moreover, an external flash for the ESP32 was needed since ESP32 overall flash memory is 4 MB and a considerable amount of this space (3.5 MB) is allocated for the storage of the running application. The remaining amount of space would not be sufficient to store information generated over a long period of time.

TABLE I: Configuration Settings for ESP32

Attribute	Configured values
Transmission power	-21dBm
Number of BLE services	1
Number of BLE characteristics	2
Maximum packet size	244 bytes
Maximum app memory	3.5 MB

The ESP32 supports FreeRTOS and thus the parallel operation of multiple threads is possible from its dual-core CPU having Xtensa LX6 microprocessor. This can be used to establish the parallel operation of transferring real-time data from one thread and transferring stored data from another thread. It is more feasible than using a single thread as the real-time data should be given more priority over old stored data.

Before starting the data transmission, the device has to be registered by the user. For that, the registration phase described under Section VI-D has to be followed accordingly. Moreover, it is better to utilize a trusted mobile relay (e.g. personal mobile) for IoT device registration accordingly. In this case, the remote server issues an IoT device ID to the patient that should be saved within the BLE device's memory. The server is able to distinguish each device and the patient associated with it for all data sent when attached with this saved device ID.

After that, ESP32 continuously scans for the mobile relays within the close proximity of the sensor. Meanwhile, it saves the data generated from the heart rate sensors in flash memory. When a single mobile relay is discovered, it connects with the relay after following the key agreement phase of the security

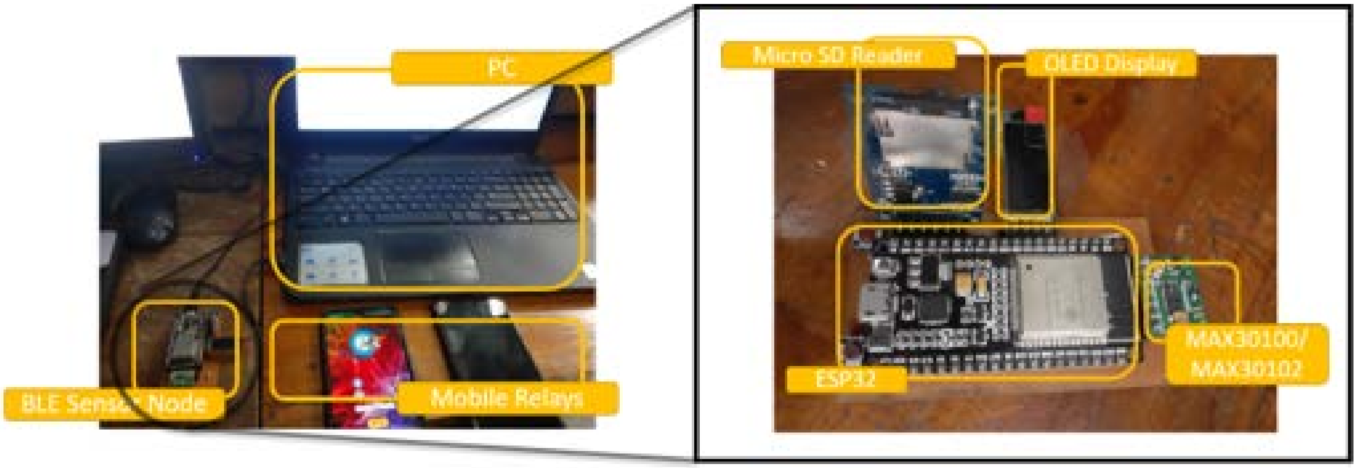


Fig. 6: Experimental Setup

protocol as indicated in Section VI-E and starts transferring data. In the presence of multiple mobile relay nodes, it selects the best mobile relay node based on the RSSI value and before connecting, it repeats the scanning multiple times to verify the availability of the best relay node.

The above-mentioned components and techniques have been put together into a single prototype-device as indicated on Fig.7. Besides the above-mentioned sensors and micro-controllers, we have integrated a body temperature sensor (MAX30205) with the prototype device with two basic purposes.

- The human body temperature can be used as an additional bio-medical measurement to detect abnormalities of the patient.
- If the body temperature becomes lower than usual for a certain duration of time and no heart rate data is generated, the carers of the patient can get an indication that the patient may not be wearing the device at the moment.

The device will be wearable near the patient's wrist and therefore, they can wear the device all the time without any interruption to their day-to-day activities.

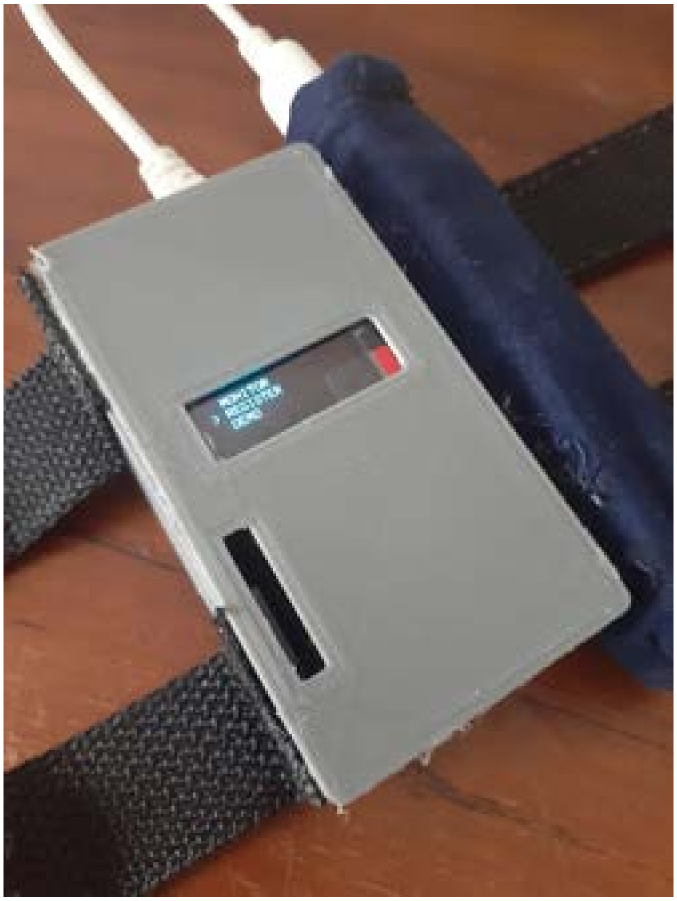


Fig. 7: Prototype IoT Device

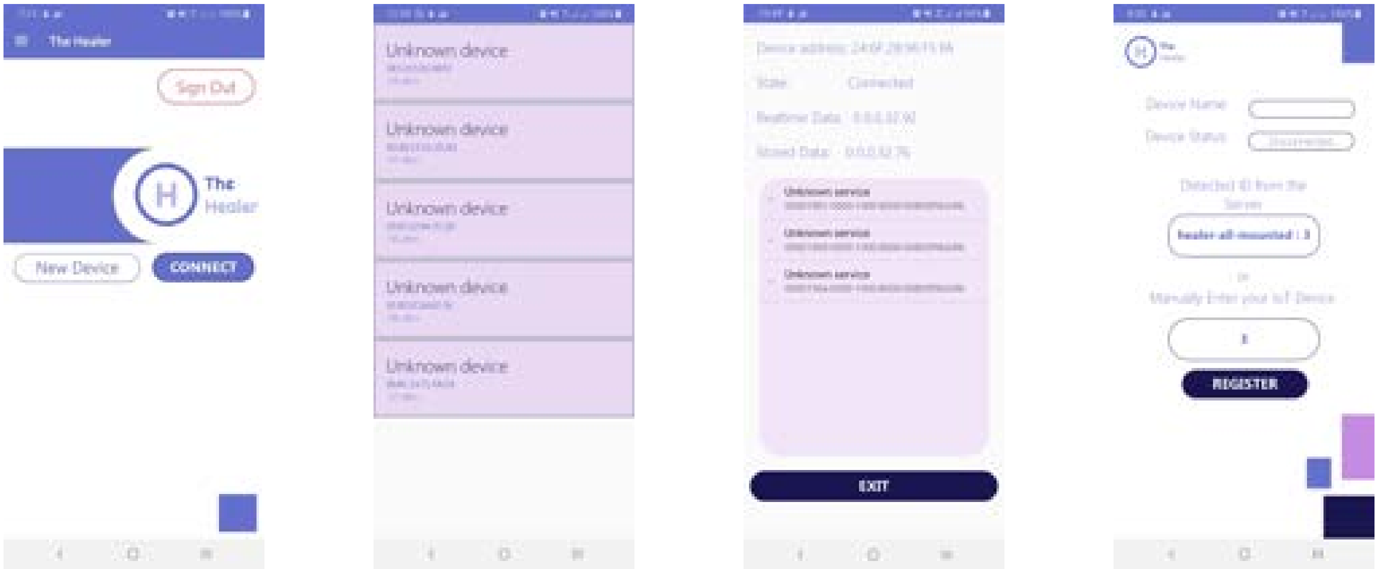
In order to register this handheld device with the server, a specific methodology has been introduced. This device is coming with two buttons built into it. The user has to keep pressing both of these buttons for 3 to 5 seconds to enter into the device's menu window. In that window, the user has to select the register option to enable the registration mode. The registration process can be carried out via any of the trusted mobile relays by following the aforementioned security protocol. During the registration process, the server will automatically grant a device ID for the specific device after having received a request from it (when registration mode is on and the trusted mobile is nearby, the request will automatically be sent by the IoT device). After the registration process, the IoT device can make connection with any other 3rd party mobile relay which is running the "The Healer" mobile application.

Once the Key exchange phase has been completed, the data transmission phase will be initiated as the next step of the communication protocol. During this phase, the mobile app installed within the relay directs these data into the server. Moreover the basic functionalities associated with the developed mobile application have been indicated with Fig. 8. We have also appended the location details of the mobile relay to the transmitted data stream (as an anonymous data field) to detect the approximate location of the patient instead of implementing the location tracing service within the IoT device of the patient. Authors have recognized that it is an energy-efficient strategy as more energy will be drained to operate a dedicated GPS sensor along with the BLE device.

For data visualization and user registration purposes, a specific web application has been implemented (Fig. 9). The front end web user interfaces were made using the Angular Framework for more sophisticated front end development and request handling. With this web application development, it is required to perform user registration prior to data transmission and is implemented as follows. A person should first undergo initial registration as a general user and they are able to select the role as a patient, donor or a carer. The donors are the third party mobile users who contribute as a mobile relay. The carer is someone who has the privilege to receive notifications about the patient's health status. Carers will be able to receive these notifications and to view real-time health information about the patient after the patient gives permission for that. A user can perform any of these roles including all three. The real-time data from the patient can be rendered in a graph format and viewed by both patient and the carers.

The remote server back end of our system is implemented using Java Spring Boot as a set of Representational State Transfer (REST) Application Program Interfaces (API).

We used web sockets to transfer data from the mobile app to the server. These data are continuously stored in the database. For each user session, the patient status is continuously monitored based on various criteria. The monitoring criterion can vary according to the patient's healthcare device sensors. First, we selected heart rate detection as the use-case for the implemented system and implemented a criterion based on upper and lower threshold heart Beats Per Minute (BPM) where the lower threshold for non-athlete individual is 60 BPM and the upper is 100 BPM [32]. To test another use case, we used a device with an accelerometer sensor ADXL345 to detect falls as well. New devices can be added with different sensors that need any other criteria, according



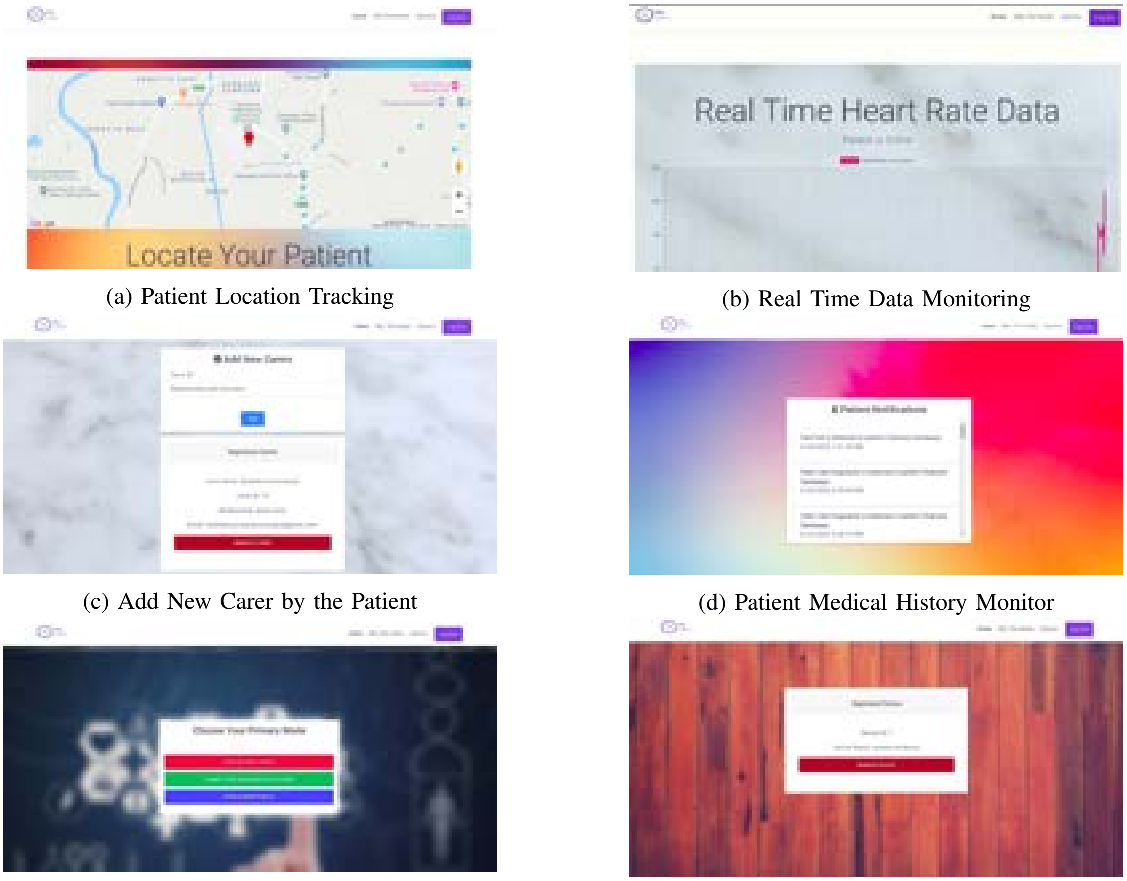
(a) Connection Establishment

(b) Scanning for an IoT Device

(c) Status After Connection Establishment

(d) IoT Device Registration

Fig. 8: Interfaces and Functionalities of Mobile Application



(a) Patient Location Tracking

(b) Real Time Data Monitoring

(c) Add New Carer by the Patient

(d) Patient Medical History Monitor

(e) Available Three Modes

(f) Add New Device

Fig. 9: Interfaces and Functionalities of Web Application

to the requirements of the patients or carers. If the emergency situation detection module detects any emergency, it sends a request to the notification API to forward an e-mail and SMS notifications to patients and pre-registered carers. All of the system notifications are controlled and triggered via the CS with aid of a few other online services. In this case, email notifications are sent to the registered carers of a patient. On the other hand, to send SMS notifications, we used a cloud communication service named Twilio along with the developed notification API within the CS.

VIII. Experiments and Results

To address several issues associated with the IoT device and BLE while designing the system, we conducted two experiments and adapted the system based on the results.

A. *Data processing time under single core and dual core processing*

In this experiment, we observed data processing times under single and dual-core conditions of the ESP32 module. Under the single-core mode, both the stored data and real-time data are processed through a single-core and transmitted accordingly. In contrast, two cores split the real-time data processing and stored data processing under the dual-core mode. For the experiment, we have selected 10 generated data samples with sizes of 5 kB, 10 kB, 20 kB, 30 kB, 50 kB, 75 kB, 100 kB, 150 kB, 300 kB and 500 kB. Then each data sample is processed 30 times in the ESP32 module and the average processing times were calculated. Here, we observed a significant improvement in processing time when two cores are utilized for the data processing function in ESP32 as indicated in the following Fig. 10.

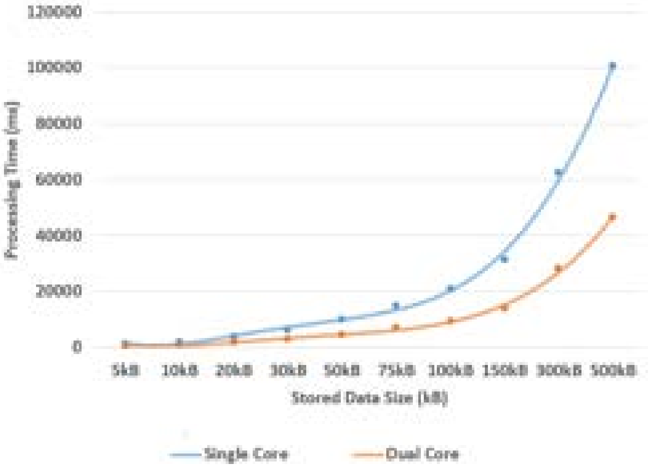


Fig. 10: Stored data size vs processing time of data in a single core and dual cores

Moreover the achieved improvement is summarized in Table II. As seen in the table, a decrease of more than 50% of transmission time delay can be achieved in each case through the utilization of two cores for the data processing ESP32.

TABLE II: Transmission Time Improvement under multi-connect

Sample Size	Transmission Time Improvement (%)
5kB	53.33 ± 2.36
10kB	55.93 ± 1.52
20kB	55.34 ± 1.77
30kB	54.91 ± 2.44
50kB	53.67 ± 3.02
75kB	53.01 ± 2.98
100kB	55.36 ± 1.27
150kB	55.68 ± 2.69
300kB	55.65 ± 6.42
500kB	53.59 ± 1.46

B. *Data transmission losses with distance*

This experiment intends to detect transmission losses related to transmission distance. First, data are sent to the mobile phone from the ESP32 continuously and for every 1000 points of data, ESP32 sends an acknowledgment to the mobile. Meanwhile the mobile also keeps track of the number of packets received. With higher inter-frame delays, the mobile phone was able to exactly synchronize this acknowledgment with its count. But when the delay is lower, we observed that the counts did not match all the time. The observed data is indicated in Fig. 11 and it is clear that the transmission disparity increases with increasing distance of transmission. It shows a drop of transmission success rate when the mobile is traveling away from the BLE sensor node ESP32. Moreover, we can clearly identify that more dissimilarities occur when the inter-frame delay is decreasing. Therefore, we can expect more reliable data transmission under sufficiently larger inter-frame delays.

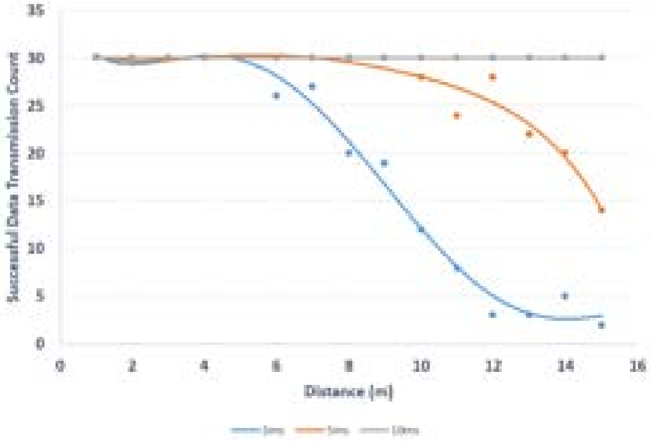


Fig. 11: Distance vs successful data transmission count

Thus, we implemented a system so that when there are multiple relays nearby, the best mobile donor should be selected, the one having the best RSSI. It shows the importance of implementing a handover mechanism to mitigate the issue of data transmission loss when the previously selected mobile is moving away from the BLE sensor. Data transmission loss can also be caused by multi path fade. In such a situation, the

IoT sensor scans and connects to preferred mobile as described in the protocol description.

The handover mechanism can be illustrated as in Fig. 12. The BLE sensor can perform the handover from one mobile to another if the current host donor mobile goes further away from the sensor than a threshold distance. In this case, it is halfway of 15 m distance that we have experimented. From Fig. 13, it can be observed that the successful data transmission possibility is significantly increased with the mobile handover because the distance between the BLE sensor and the mobile would not increase beyond the threshold distance when handovering is done.

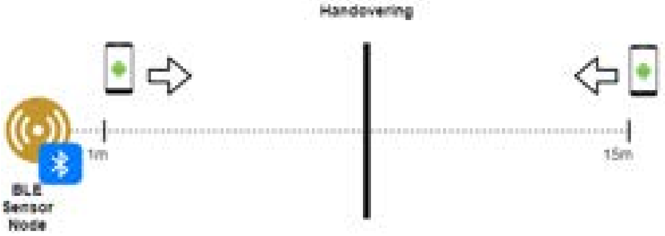


Fig. 12: Handover of mobile relays at a threshold distance

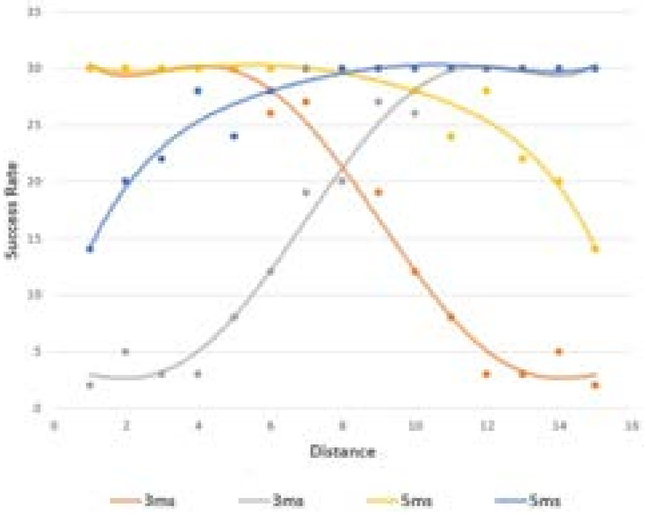


Fig. 13: Distance vs successful data transmission count when handovering under 3ms and 5ms inter frame gaps

1) Impact of Security

Several experiments were conducted to measure the impact of proposed security mechanism on the system performances. The proposed security features are implemented in the three entities, the cloud server, IoT device and the mobile relay device. Each experiment is conducted for 30 times each and obtained the average values.

C. Connection Establishment Delay

We measure the E2E connection establishment delay between the IoT node and Cloud and the results are shown in Table III.

From Table III, it can be observed that the implementation of the proposed security protocol does not consume much processing time, compared with the total time it takes to establish the communication without the security.

TABLE III: Time taken for the security protocol establishment

Task	**Average time taken (ms)**	
Connection Establishment without security	13671 ± 232	
	Time for 128 bits (ms)	Time for 256 bits (ms)
Security Protocol in IoT Device	1962 ± 1	1997 ± 1
Security Protocol in mobile relay device	500 ± 2	560 ± 2
Security Protocol in cloud server	286 ± 28	299 ± 29
Time taken for the security key agreement	2248 ± 28	2297 ± 29
Total connection establishment delay with Security	15919 ± 232	15968 ± 232
Percentage delay due to security protocol (%)	14.15 ± 0.25	14.40 ± 0.26

1) Impact on E2E Latency

In the next experiment, we measure the impact of security on E2E latency. To study the impact of security, we evaluated the time taken for AES (Advanced Encryption Standard) encryption by the IoT device, for the same file sizes we used in the Table II, ranging from 5 kB to 500 kB. The experiment results with a confidence interval of 95 % are shown in the Table IV.

TABLE IV: AES 128 bits Encryption time taken for multiple file sizes

File Size	Time taken without Encryption (ms)	Total Time taken with Encryption (AES 128bits) (ms)	Additional delay due to encryption in %
5 kB	1428 ± 21	1512 ± 21	5.58 ± 0.20
10 kB	2020± 19	2130 ± 19	5.19 ± 0.12
20 kB	3986 ± 43	4207 ± 43	5.26 ± 0.14
30 kB	6123 ± 92	6465 ± 92	5.30 ± 0.24
50 kB	10092 ± 188	10673 ± 188	5.46 ± 0.25
75 kB	14883 ± 277	15754 ± 277	5.54 ± 0.26
100 kB	20983 ± 163	22145 ± 163	5.25 ± 0.10
150 kB	31258 ± 508	32983 ± 508	5.24 ± 0.21
300 kB	62410 ± 933	66912 ± 933	5.24 ± 0.19
500 kB	100623 ± 911	106570 ± 911	5.58 ± 0.12

From the Table IV, it can be observed that the encryption only takes a small fraction compared to the time taken to send data without encryption. The delay rate is almost the same for all file sizes (around 5%) but the actual time taken to encrypt is low for small files (i.e 84 ms for 5kB) and high for bigger files (i.e 5947ms for 500kB). Therefore, the impact of encryption on the delay is lower for the small amounts of data transmitted by the IoT device.

2) Scalability (Server-Side)

In the next experiment, we measure the scalability of the proposed system by increasing the number of concurrent requests. The change of processing times with the number of server-side operations in the security protocol is shown by Fig.

14. This experiment was done with a Linux computer with 4 GB RAM, Intel Core i5-4200U CPU.

We measure the E2E connection establishment delay between the IoT node and Cloud. The results are shown in Table III.

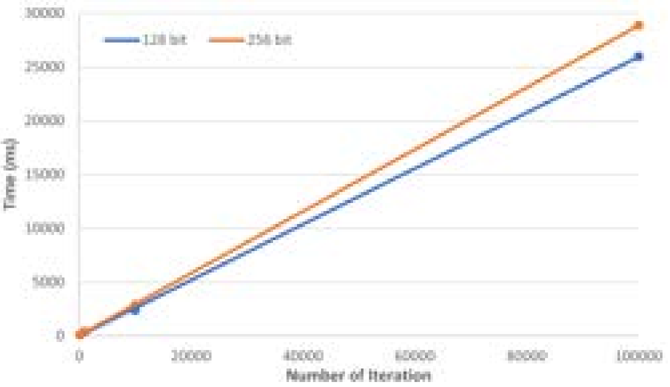


Fig. 14: Server Side data processing time with number of iterations for 128 and 256 bits master key length

In the case of the CS, the processing time increases linearly with the number of operations it handles. Therefore, the server can be scaled up according to the increasing demand from the number of users of the system.

IX. Analysis of Security Protocol

This section contains the security analysis of the proposed scheme. We use existing methods and tools to verify security properties of our protocol. We will provide both a formal security analysis as well as an informal security analysis.

A. Security Analysis

1) Formal Security Analysis

We choose to use Rubin-logic [33] to perform the verification of the protocol. Rubin-logic is a method which has been used to do security protocols analysis by several authentication and key agreement protocols [34]–[37] ; this method is based on well-defined specifications and uses the notions of Global sets, local sets, secret sets and actions.

The protocol specifications are detailed below. The proposed scheme is executed by a group comprised of wearable devices (N) controlled by a mobile relay (F) which is connected to a CS.

Global Set:

1) Principal Set: N,F,S. N is the initiator of the protocol.
2) Rule Set: The inference rules are defined as how to derive new statements from existing assertions.
3) Secret Set: $\{k_m, id_n, c_n, id_f, c_f\}$
4) Observer Set:
   Observer(k_m): {S}
   Observer(id_f, c_f): {F}
   Observer(id_n, c_n): {N}

Local Set: This set is defined for each principal, i.e., N, F, and S, respectively. As the key agreement process is being initiated by N, we start as follows:

- Principal N
  POSS(N): $\{id_n, a_n, b_n, c_n\}$
  BEL(N): $\{id_n, a_n, b_n, c_n\}$
  BL(N) =
  N1: Generate random value r_n
  N2: $x_n \leftarrow a_n \oplus id_n$
  N3: $y_n \leftarrow x_n \oplus r_n$
  N4: $tid_n \leftarrow H(id_n \| t_n \| c_n \| r_n)$
  N5: Send(F, $tid_n, y_n, a_n, b_n, t_n$)
  N6: Receive(F, $\alpha_n, \eta_n, \mu_n, tid_f, y_p, \beta_{nf}$)
  N7: $r_n^s \leftarrow \alpha_n \oplus c_n$
  N8: $a_n^+ \leftarrow H(r_n^s \| id_n) \oplus \eta_n$
  N9: $b_n^+ \leftarrow H(id_n \| r_n^s) \oplus \mu_n$
  N10: $K_n \leftarrow H(c_n \| r_n \| r_n^s \| x_n \| t_n \| tid_n)$
  N11: Derive through interpolation the shared key SK
  N12: Verify $(H(SK \| t_n \| tid_n \| tid_f), \beta_{nf})$
- Principal S
  POSS(S): $\{k_m, k_f, k_n, id_n, id_f\}$
  BEL(S): $\{k_f, k_n\}$
  BL(S) =
  S1: $k_f^* \leftarrow k_m \oplus a_f \oplus b_f$
  S2: $x_f^* \leftarrow H(k_m \| k_f^*)$
  S3: $id_f^* \leftarrow x_f^* \oplus a_f$
  S4: $r_f^* \leftarrow x_f^* \oplus y_f$
  S5: $c_f^* \leftarrow H(id_f^* \| k_m)$
  S6: Verify($H(id_f^* \| tid_n \| t_n \| c_f^* \| r_f^*)$, tid_f)
  S7: $k_n^* \leftarrow k_m \oplus a_n \oplus b_n$ S8: $x_n^* \leftarrow H(k_m \| k_n^*)$
  S9: $id_n^* \leftarrow x_n^* \oplus a_n$
  S10: $r_n^* \leftarrow x_n^* \oplus y_n$
  S11: $c_n^* \leftarrow H(id_n^* \| k_m)$
  S12: Verify($H(id_n^* \| t_n \| c_n^* \| r_n^*)$, tid_n)
  S13: Generate random values r_n^s, k_n^+ for N
  S14: Generate random values r_f^s, k_f^+ for F
  For i=f and i=n do the following
  S15: $a_i^+ \leftarrow id_i^* \oplus H(k_m \| k_i^+)$
  S16: $b_i^+ \leftarrow a_i^+ \oplus k_m \oplus k_i^+$
  S17: $\eta_i \leftarrow H(r_i^s \| id_i^*) \oplus a_i^+$
  S18: $\mu_i \leftarrow H(id_i^* \| r_i^s) \oplus b_i^+$
  S19: $\alpha_i \leftarrow c_i^* \oplus r_i^s$
  S20: $c_n^* \leftarrow H(id_n^* \| k_m)$
  S21: $K_i \leftarrow H(c_i^* \| r_i^* \| r_i^s \| x_i^* \| t_n \| tid_i)$
  S22: Derive through interpolation the shared key SK
  S23: $\beta_{nf} \leftarrow H(SK \| t_n \| tid_n \| tid_f)$
  S24: Send(F, $\alpha_n, \eta_n, \mu_n, \alpha_f, \eta_f, \mu_f, y_p, \beta_{nf}$)
- Principal F
  POSS(F): $\{id_f, a_f, b_f, c_f\}$
  BEL(F): $\{id_f, a_f, b_f, c_f\}$
  BL(F) =
  F1: Receive(N, $tid_n, y_n, a_n, b_n, t_n$)
  F2: Generate random value r_f
  F3: $x_f \leftarrow a_f \oplus id_f$
  F4: $y_f \leftarrow x_f \oplus r_f$
  F5: $tid_f \leftarrow H(id_f \| tid_n \| t_n \| c_f \| r_f)$
  F6: Send(S, $tid_n, y_n, a_n, b_n, t_n, tid_f, y_f, a_f, b_f$)
  F7: Receive(S, $\alpha_n, \eta_n, \mu_n, \alpha_f, \eta_f, \mu_f, y_p, \beta_{nf}$)
  F8: $r_f^s \leftarrow \alpha_f \oplus c_f$
  F9: $a_f^+ \leftarrow H(r_f^s \| id_f) \oplus \eta_f$

```
F10:    $b_f^+ \leftarrow H(id_f \| r_f^s) \oplus \mu_f$
F11:    $K_f \leftarrow H(c_f \| r_f \| r_f^s \| x_f \| t_n \| tid_f)$
F12:    Derive through interpolation the shared key $SK$
F13:    Verify $(H(SK \| t_n \| tid_n \| tid_f), \beta_{nf})$
F14:    Send(N, $\alpha_n, \eta_n, \mu_n, tid_f, y_p, \beta_{nf}$)
```

Below we proceed with the protocol verification. The verification process starts with execution of the actions in BL(N). In actions from N1 – N5, N computes x_n, y_n, tid_n and sends m_1 = $tid_n, y_n, a_n, b_n, t_n$ to F.

Hence, the local sets of N are changed as follows:

- POSS(N) = $\{id_n, a_n, b_n, c_n, tid_n, y_n, t_n, r_n, x_n\}$
- BEL(N) = $\{id_n, a_n, b_n, c_n, tid_n, y_n, t_n, r_n, x_n\ \}$

The global sets are updated as follows:

- Secret set: $\{r_n, x_n\}$
- Observer sets:
  Observer(r_n, x_n): {N}

Upon receiving m_1, F in actions (F2) – (F6), generates a random value r_f, computes x_f, y_f, tid_f and sends the message m_2 = $\{tid_n, y_n, a_n, b_n, t_n, tid_f, y_f, a_f, b_f\}$ to S.

Now, the local sets at F change as shown below

- POSS(F) = $\{id_n, a_n, b_n, tid_n, y_n, t_n, tid_f, y_f, a_f, b_f, x_f, r_f\}$
- BEL(F) = $\{id_n, a_n, b_n, tid_n, y_n, t_n, tid_f, y_f, a_f, b_f, x_f, r_f\}$

The global sets at F change as follows:

- Secret set: $\{r_f, x_f\}$
- Observer sets:
  Observer(r_f, x_f): {F}

After completing the action in (F6), the actions (S1)–(S12) in BL(S) are performed where S checks the received values from F. At the end of these steps S will have successfully verified tid_n and tid_f, the identities of N and F respectively. In case the verification fails, the protocol execution should be aborted. The actions in (S13) – (S24) are performed after which the shared key SK is derived through interpolation.

Then, the local sets of S are changed as follows.

- POSS(S) = $\{K_f, K_n, SK\}$
- BEL(S) = $\{K_f, K_n, SK\}$

Now the global sets of S are updated as follows:

- Secret set: $\{K_f, K_n, SK\}$
- Observer sets:
  Observer(K_f): {F,S}
  Observer(K_n): {N,S}
  Observer(SK): {S}

After receiving message m_3 in F7, F executes actions in (F8) – (F14) thus deriving the shared key SK and the session key K_f. F then sends message m_4 = $\alpha_n, \eta_n, \mu_n, tid_f, y_p, \beta_{nf}$ to N. The local sets of F are finally changed as follows.

- POSS(F) = $\{SK, K_f\}$
- BEL(F) = $\{SK, K_f\}$

The global sets are updated as follows:

- Secret set: $\{SK, K_f\}$
- Observer sets:
  Observer(SK, K_f): {F,S}

Upon receiving the message m_4 in N6, N executes the actions in (N7) – (N12) and derives the shared key SK and the session key K_n. The local sets of N are finally changed as follows.

- POSS(N) = $\{SK, K_n\}$
- BEL(N) = $\{SK, K_n\}$

The global sets are updated as follows:

- Secret set: $\{SK, K_n\}$
- Observer sets:
  Observer(SK, K_n): {N,S}

This result implies that:

- r_f, r_n as well as k_f, k_n are fresh for each session, and are known by the legitimate F, N and S.
- Only the legitimate entities N, F and S are able to derive the common shared key SK.
- S is able to verify the identities of N and F.

This result proves that our scheme resists against the attack model described in Section VI-B.

2) *Informal Security Analysis*

Let us now discuss the strength of the proposed protocol with respect to the obtained security features and the resistance against the attacks considered in the required security features in Section VI-A.

- **Confidentiality:** The transmitted messages m_1, m_2, m_3, m_4 have no meaning without knowledge of the secret key material.
- **Mutual authentication:** The common shared keys between CS and the mobile relay K_f on the one hand and CS and the wearable device K_n on the other hand contain a random variable derived by each of the involved entities. As the common key SK is based on these keys K_f, K_n, each entity has contributed in the construction of it.
- **Anonymity:** This feature is guaranteed by the fact that the dynamic identity of the wearable device (a_n, b_n) and the mobile relay (a_f, b_f), sent in messages m_1, m_2 is updated independently in each request. Without knowledge of the key material, stored in the tamper proof part of the memory, it is impossible to derive the relation with the real identity.
- **Unlinkability:** In order to derive the updated dynamic identity, knowledge of the parameters c_i, id_i is required; these parameters are securely stored by the wearable device $i = n$ and mobile relay $i = f$.
- **Node capture attack:** Suppose the node is captured and even the key material stored in the tamper resistant memory is leaked, the security with respect to that node is completely broken. However, the attack remains local as it has no impact on other devices in the system since each device has its own specific construction and the master key cannot be retrieved from the stored values without being able to break the hash function.
- **Impersonation and MITM attacks:** These attacks are not possible due to the mutual authentication feature. However, special care should be given to a malicious mobile relay node, willing to take over the request sent to another mobile relay. For the computation of K_n, the

TABLE V: The cryptographic operations that device, relay node and CS need to perform for a key agreement scheme. Note that T_b = time for bilinear pairing, T_{mp} = time for point multiplication, T_{ap} = time for point addition, T_s = time for symmetric encryption/decryption, T_H = time for map to point, T_h = time for hash operation, T_x = time for XOR operation.

Scheme	Cost for wearable device	Cost for mobile relay	Cost for CS
[20]	$2T_{mp} + 1T_b + 6T_h$	$2T_{mp} + 1T_b + 4T_h$	$3T_{mp} + 1T_b + 11T_h$
[22]	$T_H + 5T_{mp} + 1T_b + 3T_{ap} + 4T_h$	$4T_H + 13T_{mp} + 7T_b + 7T_{ap} + 8T_h$	$T_H + 6T_{mp} + 3T_b + 4T_{ap} + 5T_h$
[23]	$7T_{mp} + 2T_{ap} + 12T_h + 2T_s$	$7T_{mp} + 2T_{ap} + 13T_h + 2T_s$	$9T_{mp} + 4T_{ap} + 13T_h + 4T_s$
Ours	$5T_x + 5T_h$	$5T_x + 5T_h$	$14T_x + 11T_h$

server also includes the temporary identity of the mobile relay tid_f in the hash, which is also added in clear text by the mobile relay to the message m_4. As a consequence, it is not possible for another mobile relay to change the identity of the mobile relay, without CS being aware of it.

The verification parameter β_{nf} involves the temporary identities of the entities, both mobile relay and wearable device, and therefore, no impersonation attacks can be performed. In addition, mobile relay and wearable device are ensured in this way that CS has legitimated them.

- **Replay attacks:** are avoided by using both timestamps and random values in the protocol.
- **Online/offline dictionary attack:** Guessing the master key is useless as another temporary key needs to be determined as well. This key changes in each communication request.

  Guessing the identity of the node, results in deriving x_n, r_n, but will not lead to finding the random value r_n^s chosen by CS as that requires knowledge of c_n. Both values need to be known to derive the key material with the wearable device. Moreover, to check the validity by using the hash value β_{nf}, and the identity material of the relay id_f, c_f should be guessed. Consequently, in order to make a successful dictionary attack, four different parameters id_f, c_f, id_n, c_n should be guessed.

B. Performance Analysis

The proposed scheme is very efficient in terms of computation and communication. For the constrained devices, only 5 XOR and 5 hash operations are executed. Table V shows a comparison between our scheme and related works in the literature. By using hash functions and negligible XOR operations, we avoid compute intensive paring operations as observed in many cryptographic algorithms [20], [22] thus offering secure exchange of data at a very low cost.

X. Discussion

A. Comparison with Existing Work

Table VI shows the added features of our proposed architecture compared with existing solutions. From the table, it is clear that the proposed system is a unique solution and addresses many problems existing with similar proposals and implementations.

Our system has proposed and implemented a mobile relay based system and it provides anonymity and unlinkability with our security protocol. It also ensures the confidentiality of all the parties involved in establishing the communication. Comparing with similar works, we can observe that most of the solutions do not provide those features. In addition, our solution proposes multi-connect, load balancing and we implemented the automatic relay handover, which are not available in many of the similar work.

TABLE VI: Comparison of proposed solution with the existing pertinent works

Characteristic	Ref. **[5]**	Ref. **[6]**	Ref. **[7]**	Ref. **[8]**	Ref. **[2]**	**Ours**
BLE Support	–	✓	–	✓	✓	✓
Support for Third party Relay	–	–	–	✓	✓	✓
E2E Encryption	✓	✓	✓	✓	✓	✓
Automatic Relay Handover	✓	–	✓	–	–	✓
Multi Connect	–	✓	–	–	–	✓
Transmission of Real-time data	✓	✓	✓	✓	✓	✓
Store and Forwarding of data	✓	✓	✓	–	–	✓
Support for Load Balancing	✓	–	–	–	–	✓
Confidentiality Protection	✓	✓	✓	–	✓	✓
Mutual authentication	✓	✓	✓	✓	✓	✓
Anonymity	–	–	–	–	–	✓
Unlinkability	–	–	–	–	–	✓

B. Limitations

1) Limitations in Communication Protocol

Even though we have proposed and implemented a highly sophisticated communication protocol in this research, there are a few limitations associated with this development as well. In this case, it is clear that, the protocol is limited to the usage of smartphones which support BLE. That means it is supporting the Android versions from Android 4.3 (API Level 18) [38] and it is not working with the legacy Bluetooth versions. In addition to that, the relay mobile should be powered with both BLE transmission and reception capabilities. Moreover, as indicated from Section VIII-B, there can be issues associated with small inter-frame gaps when the distance between the mobile relay and the IoT device increases. Therefore, it is obvious that we have to utilize somewhat higher inter-frame gaps such as 10ms accordingly. Furthermore, the associated implementations of the proposed protocol have been limited for the connection between an IoT handheld device to a single mobile relay; however, it has the capability to expand multiple connections as described in Section V-B.

2) Limitations in Security

The proposed protocol has also a few limitations related to security. First, a secure key storage is required in IoT node and mobile relay. Also, the centralized CS is a single point of attack since the whole system security relies on master key $\{k_m\}$.

C. Usefulness of extra session key

The proposed key exchange mechanism is distributing different keys between the entities. In the current version of the proposed architecture, the mobile relay is just acting as a relay and forwards the received data from the sensor nodes. Thus, the wearable devices currently only use the key K_n for encryption since we support E2E encryption. However, our key agreement phase supports another session key (i.e. SK), which is shared between all three entities. If the mobile relay can do some kind of prepossessing for data at the relay itself, wearable devices can use the key SK for encryption. That way, the mobile relay is able to decrypt and process the data. The prepossessing of data at the mobile relay will be carried out in future research.

XI. Conclusion

This paper proposed a secure BLE relay-based emergency situation detection system for AAL. The proposed solution extended the features of existing solutions by adding new capabilities such as multi-connect, automatic handovering, storage, forwarding data and load balancing as well as security features confidentiality, mutual authentication, unlinkability and anonymity. A prototype of the proposed solution was developed and performed several experiments to get insights on the performance of the proposed system. It was detected that multi-core processing of real-time and stored data separately is a better solution than processing both together sequentially. The experiments revealed that the distance of the mobile from the sensor is important due to the increase in reliability of data with closer proximity. Therefore, it can be concluded that a data transmission handover mechanism in the BLE sensor node from a distant mobile relay to a closer one is important when implementing such systems.

To enhance the security of the system, we also proposed a novel secure symmetric key agreement scheme. The proposed scheme is secure as the capacity to build a shared common key is based on freshly generated parameters. Our scheme is efficient in terms of computation since it uses very low cost cryptographic operations (i.e concatenation, XOR and hash functions). We have used Rubin logic to provide this scheme's security strengths and we have also done an informal analysis of the scheme for several types of security threats i.e node capturing attack, impersonation attack, man-in-the-middle attack, replay attack, online/offline dictionary attack.

For the future work, we intend to extend the work to the development of machine learning algorithms to detect emergencies and anomalies in both health and security related data.

Acknowledgement

This work is partly supported by Academy of Finland in 6Genesis (grant no. 318927) and VLIR-UOS in IUC 2017 UB Phase3 project.

References

[1] B. Farahani, F. Firouzi, and K. Chakrabarty, "Healthcare IoT," in Intelligent Internet of Things. Springer, 2020, pp. 515–545.

[2] P. Porambage, A. Manzoor, M. Liyanage, A. Gurtov, and M. Ylianttila, "Managing Mobile Relays for Secure E2E Connectivity of Low-Power IoT Devices," in 2019 16th IEEE Annual Consumer Communications & Networking Conference (CCNC). IEEE, 2019, pp. 1–7.

[3] A. Manzoor, P. Porambage, M. Liyanage, M. Ylianttila, and A. Gurtov, "Mobile Relay Architecture for Low-power IoT Devices," in 2018 IEEE 19th International Symposium on" A World of Wireless, Mobile and Multimedia Networks"(WoWMoM). IEEE, 2018, pp. 14–16.

[4] J.-W. Yoon, Y.-k. Ku, C.-S. Nam, and D.-R. Shin, "Sensor network middleware for distributed and heterogeneous environments," in 2009 International Conference on New Trends in Information and Service Science. IEEE, 2009, pp. 979–982.

[5] A. M. Rahmani, T. N. Gia, B. Negash, A. Anzanpour, I. Azimi, M. Jiang, and P. Liljeberg, "Exploiting Smart E-Health Gateways at the Edge of Healthcare Internet-of-Things: A fog computing approach," Future Generation Computer Systems, vol. 78, pp. 641–658, 2018.

[6] S. Raza, P. Misra, Z. He, and T. Voigt, "Building the Internet of Things with Bluetooth Smart," Ad Hoc Networks, vol. 57, pp. 19–31, 2017.

[7] S. R. Moosavi, T. N. Gia, E. Nigussie, A. M. Rahmani, S. Virtanen, H. Tenhunen, and J. Isoaho, "End-to-End Security Scheme for Mobility Enabled Healthcare Internet of Things," Future Generation Computer Systems, vol. 64, pp. 108–124, 2016.

[8] M. Haus, A. Y. Ding, and J. Ott, "Managing Iot at the Edge: The Case for BLE Beacons," in Proceedings of the 3rd Workshop on Experiences with the Design and Implementation of Smart Objects. ACM, 2017, pp. 41–46.

[9] H. Wirtz, T. Zimmermann, M. Serror, and K. Wehrle, "Collaborative on-demand wi-fi sharing," in 2015 IEEE 40th Conference on Local Computer Networks (LCN). IEEE, 2015, pp. 19–27.

[10] P. Kumar, A. Braeken, A. Gurtov, J. Iinatti, and P. Ha, "Anonymous secure framework in connected smart home environments," IEEE Transactions on Information Forensics and Security, vol. 12, no. 4, pp. 968–979, 2017.

[11] C.-M. Chen, B. Xiang, T.-Y. Wu, and K.-H. Wang, "An anonymous mutual authenticated key agreement scheme for wearable sensors in wireless body area networks," Applied Sciences, vol. 8, no. 1074, pp. 1–15, 2018.

[12] X. Li, M. Ibrahim, S. Kumari, A. Sangaiah, V. Gupta, and K. Choo, "Anonymous mutual authentication and key agreement scheme for wearable sensors in wireless body area networks," Computer Networks, vol. 129, no. 2, pp. 429–443, 2017.

[13] L. Gong, "Lower bounds on messages and rounds for network authentication protocols," Proceedings of the 1st ACM Conference on Computer and Communications Security, pp. 26–37, 1993.

[14] C. Lee, S. Chen, and C. Chen, "A computation-efficient three-party encrypted key exchange protocol," Applied Mathematics & Information Sciences, vol. 6, no. 3, pp. 573–579, 2012.

[15] X. Li, J. Niu, S. Kumari, M. Khan, L. Liao, and W. Liang, "Design and analysis of a chaotic maps-based three-party authenticated key agreement protocol," Nonlinear Dynamics, vol. 80, no. 3, pp. 1209–1220, 2015.

[16] T. Lee and T. Hwang, "Three-party authenticated key agreements for optimal communication," Journal of SomeThing, vol. 12, no. 3, pp. 1–25, 2017.

[17] L. Ni, G. Chen, and J. Li, "Escrowable identity-based authenticated key agreement protocol with strong security," Comput. Math. Appl, vol. 65, no. 9, pp. 1339–1349, 2013.

[18] D. Wang and P. Wang, "On the anonymity of two-factor authentication schemes for wireless sensor networks: Attacks, principle and solutions," Computer Networks, vol. 73, pp. 41–57, 2014.

[19] Y. Chen, J. Martinez, P. Catellejo, and L. Lopez, "An anonymous authentication and key establish scheme for smart grid: Fauth," Energies, vol. 10, no. 1345, pp. 1–23, 2017.

[20] X. Jia, D. He, N. Kumar, and K. Choo, "Authenticated key agreement scheme for fog-driven iot healthcare system," Wireless Networks, Springer, vol. 25, no. 8, pp. 4737–4750, 2019.

[21] H. Hamid, S. Rahman, M. Hossain, A. Almogren, and A. Alamri, "A security model for preserving the privacy of medical big data in a healthcare cloud using a fog computing facility with pairing-based cryptography," IEEE Access, vol. 5, pp. 22 313–22 328, 2017.

[22] C. Liu, W. Tsai, T. Chang, and T. Liu, "Ephemeral-secret-leakage secure id-based three party authenticated key agreement protocol for mobile distributed computing environments," Symmetry, vol. 10, no. 4, p. 84, 2018.

[23] S. Patonico, A. Braeken, and K. Steenhaut, "Identity-based and anonymous key agreement protocol for fog computing resistant in the canetti-krawczyk security model," Wireless Networks, 2019.

[24] A. Dohr, R. Modre-Opsrian, M. Drobics, D. Hayn, and G. Schreier, "The internet of things for ambient assisted living," in 2010 seventh international conference on information technology: new generations. Ieee, 2010, pp. 804–809.

[25] P. Rashidi and A. Mihailidis, "A survey on ambient-assisted living tools for older adults," IEEE journal of biomedical and health informatics, vol. 17, no. 3, pp. 579–590, 2012.

[26] D. of Economic and S. A. P. Division, World population ageing, 2019 highlights. United Nations, 2019.

[27] "Bluetooth archived specifications : Bluetooth low energy version 4.0 cs – core specifications." [Online]. Available: https://www.bluetooth.com/specifications/archived-specifications

[28] K. Townsend, C. Cufí, R. Davidson et al., Getting started with Bluetooth low energy: tools and techniques for low-power networking. " O'Reilly Media, Inc.", 2014.

[29] C. Gomez, J. Oller, and J. Paradells, "Overview and evaluation of bluetooth low energy: An emerging low-power wireless technology," Sensors, vol. 12, no. 9, pp. 11 734–11 753, 2012.

[30] J. Tosi, F. Taffoni, M. Santacatterina, R. Sannino, and D. Formica, "Performance evaluation of bluetooth low energy: A systematic review," Sensors, vol. 17, no. 12, p. 2898, 2017.

[31] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the internet of things," in Proceedings of the first edition of the MCC workshop on Mobile cloud computing, 2012, pp. 13–16.

[32] H. K. Hughes and L. K. Kahl, A Manual For Pediatric House Officers, The Harriet Lane Handbook, 2018, Elsevier, Inc. Elsevier, Inc, 2018.

[33] A. D. Rubin and P. Honeyman, "Nonmonotonic cryptographic protocols," in Proceedings The Computer Security Foundations Workshop VII. IEEE, 1994, pp. 100–116.

[34] A. Nicholson, M. Corner, and B. Noble, "Mobile device security using transient authentication," Transactions on Mobile Computing IEEE, vol. 5, no. 11, pp. 1489–1502, 2006.

[35] M. A. Mughal, X. Luo, A. Ullah, S. Ullah, and Z. Mahmood, "A lightweight digital signature based security scheme for human-centered internet of things," IEEE Access, vol. 6, pp. 31 630–31 643, 2018.

[36] A. Braeken, M. Liyanage, P. Kumar, and J. Murphy, "Novel 5g authentication protocol to improve the resistance against active attacks and malicious serving networks," IEEE Access, vol. 7, pp. 64 040–64 052, 2019.

[37] P. Shabisha, A. Braeken, P. Kumar, and K. Steenhaut, "Fog-orchestrated and server-controlled anonymous group authentication and key agreement," IEEE Access, vol. 7, pp. 150 247–150 261, 2019.

[38] "Bluetooth low energy overview : Android developers." [Online]. Available: https://developer.android.com/guide/topics/connectivity/bluetooth-le

Placide Shabisha received his M.Sc degree in Communication Networks from the University of Sidi Bel Abbes (Algeria) in 2012. Since 2013, he is an Assistant Lecturer at the Department of Information and Communications Technology, University of Burundi. He is currently pursuing the PhD degree in Engineering Sciences at the Vrije Universiteit Brussel, Belgium with a scholarship from the VLIR-UOS project: IUC 2017 Phase 3 UB. His research interests include Internet of Things, Cloud computing, Elliptic curve cryptography, Computer and network security.

Chamara Sandeepa is from the Faculty of Engineering, University of Ruhuna, Galle, Sri Lanka, graduated with a Bachelor of Science of Engineering (B.Sc. Eng. (Second Class Honours Upper Division)) in Electrical and Information Engineering. His degree program included subjects from Electrical, Telecommunication, Electronics and Software Engineering while he is specialized under Software Engineering sector. His research interests are, IoT, e-Healthcare, Computer Science and Data Science.

Charuka Moremada is currently working as a Temporary Instructor at Department of Computer Engineering, Faculty of Engineering, University of Peradeniya, Sri Lanka. He has obtained his, Bachelor of the Science of Engineering degree from Faculty of Engineering, University of Ruhuna in 2020, specialized in Electrical and Information Engineering (B.Sc.Eng.(First Class Honours)). He has co-authored some publications in certain research areas and his current research interests are, Blockchains, Quantum Resistant Security, 5G, 6G, Software Defined Networking, Internet of Things, Ambient Assisted Living, BLE-Contact Tracing, Mobile Communications and Artificial Intelligence, Data Science and Machine Learning.

.

Nadeeka Dissanayaka is currently working as a Temporary Demonstrator at Department of Electrical and Electronic Technology, Faculty of Technology, Rajarata University of Sri Lanka. She has obtained her, Bachelor of the Science of Engineering degree specialized in Electrical and Information Engineering (B.Sc.Eng.(Second Class Honours upper Division)),from the department of Electrical and Information Engineering Faculty of Engineering, University of Ruhuna, Galle, Sri Lanka in 2020. She has followed electrical engineering, telecommunication engineering, electronic engineering and software engineering. Her research interests are, IoT, e-Healthcare and electrical engineering aspects.

Tharindu Gamage is currently working as a Lecturer in the Department of Electrical and Information Engineering, University of Ruhuna, Sri Lanka. He obtained his first degree from the Department of Computer Science and Engineering, University of Moratuwa, Sri Lanka in 2009. He worked as a research assistant for several years in the Department of Electronic and Telecommunication Engineering, University of Moratuwa, Sri Lanka. During the time he was working as a research assistant, he obtained his M.Sc. from the same department of University of Moratuwa, Sri Lanka in 2014. His research interests are IoT, Embedded Systems, High Performance Computing and Medical Image Processing. URL: http://eie.eng.ruh.ac.lk/team/tharindu-gamage/

An Braeken obtained her MSc Degree in Mathematics from the University of Gent in 2002. In 2006, she received her PhD in engineering sciences from the KULeuven at the research group COSIC (Computer Security and Industrial Cryptography). She became professor in 2007 at the Erasmushogeschool Brussel (currently since 2013, Vrije Universiteit Brussel) in the Industrial Sciences Department. Prior to joining the Erasmushogeschool Brussel, she worked for almost 2 years at the management consulting company Boston Consulting Group (BCG). Her current interests include security and privacy protocols for IoT, cloud and fog, blockchain and 5G security. She is (co-)author of over 120 publications. She has been member of the program committee for numerous conferences and workshops (IOP2018, EUC 2018, ICNS 2018, etc.) and member of the editorial board for Security and Communications magazine. She has also been member of the organizing committee for the IEEE Cloudtech 2018 conference and the Blockchain in IoT workshop at Globecom 2018. In addition, she is since 2015 reviewer for several EU proposals and ongoing projects, submitted under the programs of H2020, Marie Curie and ITN. She has cooperated and coordinated more than 12 national and international projects. She has been STSM manager in the COST AAPELE project (2014-2017) and is currently in the management committee of the COST RECODIS project (2016-2019).

Kris Steenhaut received the master in Engineering Sciences in 1984 and the master in Applied Computer Sciences in 1986 and the PhD degree in Engineering Sciences from Vrije Universiteit Brussel (VUB) in 1995. Currently she is professor at the department of Electronics and Informatics (ETRO) and the department of Engineering technology (INDI), Faculty of Engineering, Vrije Universiteit Brussel, Belgium. Her research interests include the design, implementation and evaluation of Wireless Sensor Networks for building automation, environmental monitoring, autonomous ground vehicle applications and smart grids.

Madhusanka Liyanage (S07, M16, SM20) received his Ph.D. in Communication Engineering in 2016 from University of Oulu, Oulu, Finland. Currently, he is working as Assistant Professor/ Ad Astra Fellow at School of Computer Science, University College Dublin, Ireland. He has been a Visiting Research Fellow at the Department of Computer Science, University of Oxford, Data61, CSIRO, Sydney, Australia, the Infolabs21, Lancaster University, U.K., and Computer Science and Engineering, The University of New South Wales during 2015-2018. He is also an adjunct professor at the University of Oulu, Finland and a recipient of prestigious Marie Skłodowska-Curie Actions Individual Fellowship during 2018-2020. He has co-authored over 80 publications including three edited books and holds one patent His research interests are SDN, IoT, Block Chain, mobile and virtual network security. URL: http://madhusanka.com