Wireless Networks (2024) 30:5759–5770
https://doi.org/10.1007/s11276-023-03347-z

BLE-based secure tracking system proposal

Candelaria Hernández-Goya[1] · Ricardo Aguasca-Colomo[2] · Cándido Caballero-Gil[1]

Accepted: 3 April 2023 / Published online: 29 April 2023

Abstract
As communication capabilities of mobile devices continue to advance, ensuring reliability and security has become increasingly crucial. The aerial tracking system presented in this paper provides a useful solution for tracking and tracing objects in various scenarios. To ensure reliability and security, the system incorporates appropriate mechanisms, including Lightweight Cryptography, to prioritize confidentiality and integrity. The Android component of the system has two modes of operation: Tracker Mode, running on a smartphone mounted on a drone (RPA), and Client Mode, running on mobile devices on the ground. In Client Mode, users transmit their positioning and trajectory information via Bluetooth Low Energy beacon mode, which is then relayed to the server backend via the 4G/5G network once the RPA enters an area with coverage. The system provides a reliable and secure solution for situations where tracking and tracing are essential, such as the supervision and control of public areas with capacity control or tracking and localizing people in isolated environments.

Keywords Beacon tracking · BLE · Lightweight cryptography · RPA

1 Introduction

Bluetooth Low Energy (BLE) technology is a popular choice for Internet of Things (IoT) applications. It boasts low power consumption and high data transfer rates, making it ideal for object tracking, equipment and personnel monitoring, indoor navigation, and targeted marketing messages. Additionally, its signal has an omnidirectional coverage that can extend up to 100 ms, allowing for versatile use cases. In these scenarios, high data transfer rates and accuracy are vital, and the secure transmission of sensitive information, such as real-time location data, is critical.

✉ Candelaria Hernández-Goya
mchgoya@ull.edu.es

✉ Ricardo Aguasca-Colomo
ricardo.aguasca@ulpgc.es

Cándido Caballero-Gil
ccabgil@ull.edu.es

[1] Ingeniería Informática y de Sistemas, Universidad de La Laguna, San Cristóbal de La Laguna, S/C de Tenerife, Spain

[2] Instituto Universitario de Sistemas Inteligentes y Aplicaciones Numéricas en Ingeniería, Universidad de Las Palmas de Gran Canaria, Las Palmas de Gran Canaria, Spain

BLE technology can also be used with beacons to serve as gateways for transmitting processed sensor data from IoT systems. This can provide a valuable connection between IoT devices and the cloud, allowing for data analysis and real-time monitoring [1].

1.1 Motivation

The potential of BLE beacons for enabling innovative IoT applications is currently being explored, as evidenced by recent studies such as [2–4].

In certain emergency scenarios, as depicted in Fig. 1, mobile network coverage availability cannot be guaranteed. The coverage map shown in Fig. 2, obtained from a mobile operator on the island of Tenerife (Spain) [5], highlights large areas without any coverage. The proposed system can serve as a valuable tool in such circumstances.

Our system can gather data that is suitable for processing with AI-based tools, which can enable powerful decision-making capabilities in time-critical situations. Figure 3 shows a representation of the data gathered by the RPA, which collects information from various users in the field. The data identifies individual positions, and vectors indicate direction and velocity. Additionally, the system can identify "hot" zones or clusters that pose a risk based

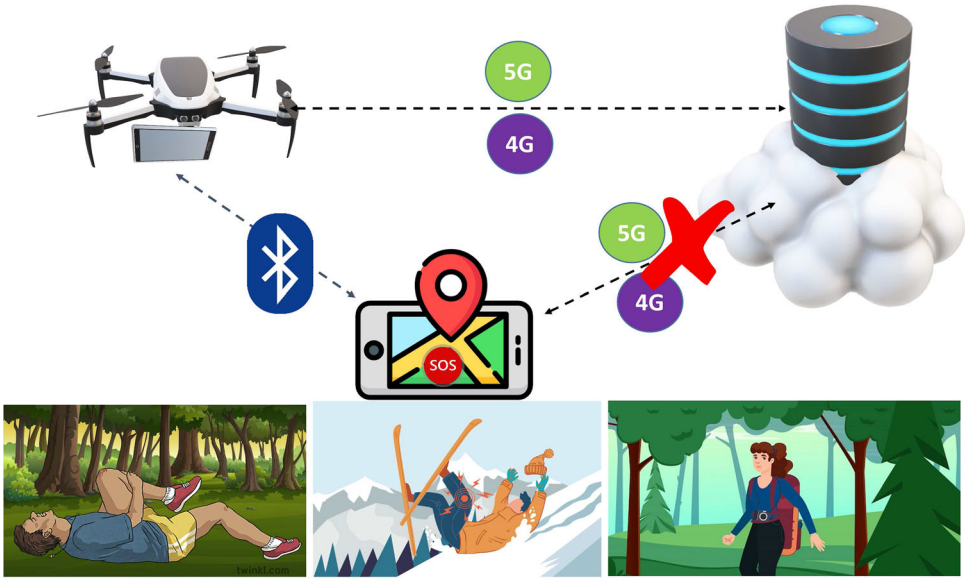

Fig. 1 System global view

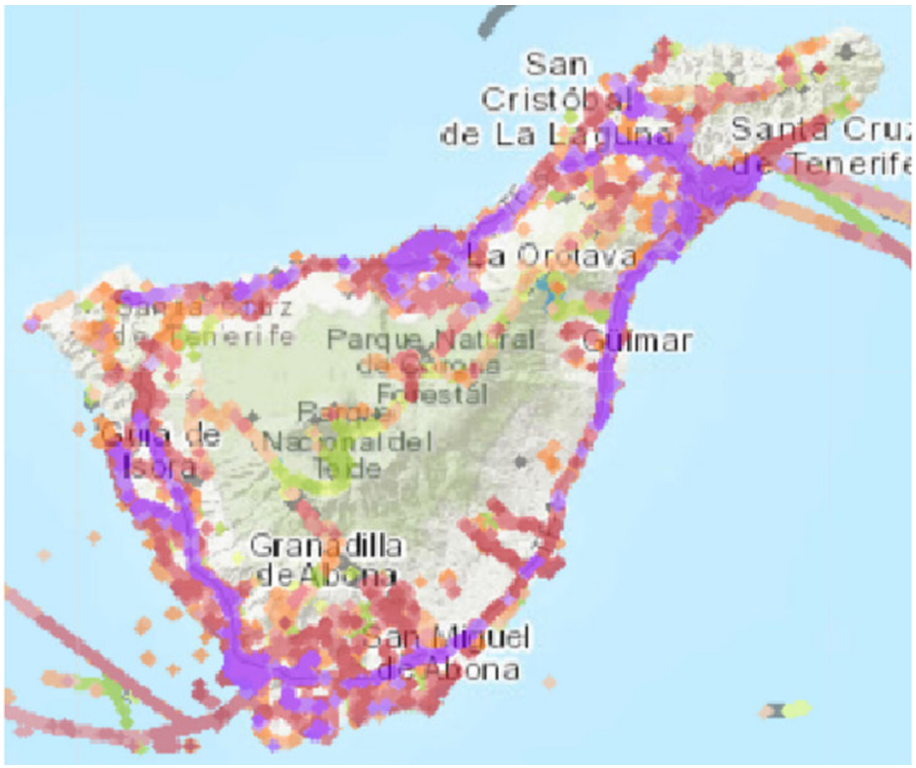

Fig. 2 Signal coverage in Tenerife

on proximity, and it can track a user's position history to ensure they have not crossed any forbidden zones.

1.2 Aims and contributions

In this research, we introduce a tracking solution that leverages Bluetooth Low Energy (BLE) beacons. The collected data is captured by a mobile phone integrated into an RPA and sent to a cloud-based server for further processing once mobile network coverage is available. Figure 1 illustrates the system architecture. To ensure data security, we have implemented protection mechanisms to safeguard transmitted information.

Our work has several notable contributions. We have designed a robust RPA-based system for conducting surveillance missions using BLE technology and developed

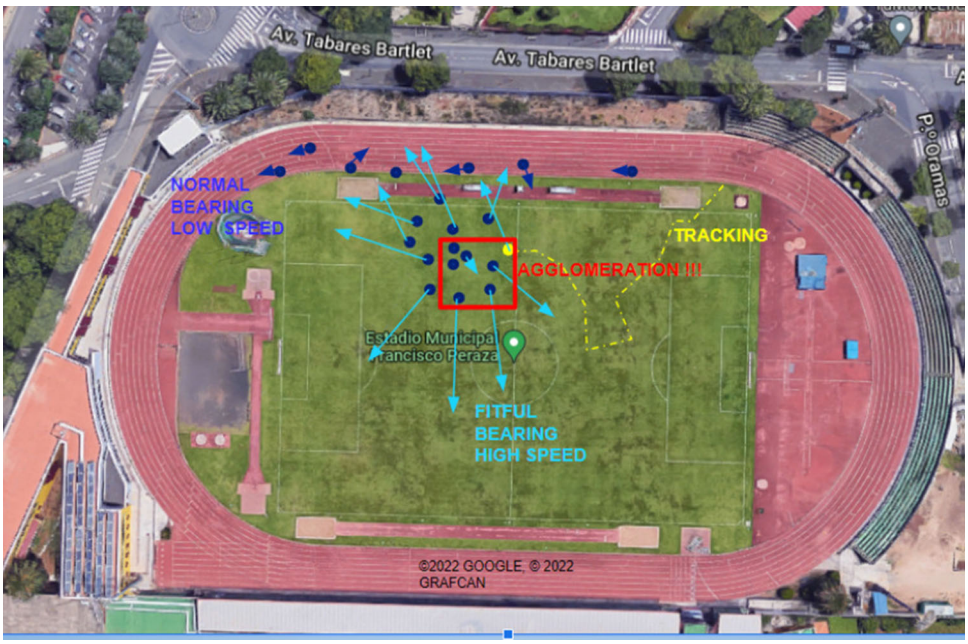

Fig. 3 Visualizing tracked data

an infrastructure that enables location information transmission through BLE technology. To guarantee secure transmission of real-time data, we have also implemented robust security measures, including authentication and encryption using lightweight cryptography. Finally, all data collected by the system is stored in a cloud-based database and can be processed using advanced AI techniques.

1.3 Related work

Currently, there are several applications that allow tracking users. An interesting reference is the "Wikiloc" application [6], which is capable of tracking outdoor routes. This application allows subsequent consultation of the activities carried out along with some statistics (speed, time, the route followed, number of stops, etc.). All this information can be shared in the community so that the application recommends routes close to an area where the user is.

"Life360" [7] is a family tracking app that uses GPS location data to track the location of family members in real-time. It also has other features such as messaging, safety alerts, and driving safety analysis.

The "Find My" app, developed by Apple [8], is designed to help users locate their Apple devices and share their location with others. In addition, the app leverages the "Find My network," a crowdsourced network of Apple devices that can locate lost items, even in cases where the lost device is not connected to Wi-Fi or cellular networks. This is made possible by using Bluetooth signals emitted by nearby Apple devices. According to Apple, the Find My network is a secure and private way to locate lost items, as the location data is encrypted and anonymous.

There is a wealth of literature on the use of RPA technology for surveillance and tracking, with many publications focusing on the use of optical sensors and cameras. However, the main challenge with these technologies is the accurate identification and classification of real-time video images, especially when it comes to potentially dangerous situations. Some recent research efforts have aimed to overcome these challenges, including the use of drones to train convolutional neural networks (CCN) in real-time image detection of road traffic accidents [9], and the development of surveillance systems using RPAs with cameras to identify possible threats, such as people carrying weapons [10].

Other research efforts have focused on classifying aerial images of large areas to identify potentially dangerous situations, using CNNs trained on collections of images of offensive situations [11]. The security of communications between ground control and RPAs has also been studied, with some research exploring the use of NTRU encryption to secure transmissions between them [12].

In addition to surveillance applications, RPAs have also been used for real-time analysis of multispectral images for agricultural crops, with different image-processing algorithms tested in conjunction with GPS data [13]. Another interesting application of RPAs is shown in [14], where a fully autonomous RPA is used for spraying disinfectant in public areas. The system analyses images in real-time and can emit an audible warning message through a built-in

loudspeaker if necessary. The RPA is activated and deactivated via a mobile phone using a BLE module.

While there is a vast body of literature on RPA applications in surveillance and tracking, papers that specifically focus on defining security services in the BLE Beacon environment are relatively scarce. Typically, the applications developed with this technology are primarily focused on marketing and advertising, and as a result, papers related to this topic often concentrate on safeguarding anonymity rather than protecting the information that is transmitted. The authors of [1] present a lightweight and dynamic symmetric encryption algorithm, known as DLS, for secure data transmission through BLE beacons. Although they claim that the xor function is the core of their approach, the cipher algorithms they employ are actually block ciphers.

In [15] a block cipher is proposed together with the use of the MD5 hash function. The cipher implemented in our system is a stream cipher, as it is better adapted to the constraints present and the hash function used for information authentication is robust. MD5 is considered obsolete and is not suitable for devices with as many constraints as beacons.

According to the authors in [16], the Eddystone-EID protocol is designed to ensure the privacy of users and prevent their tracking, while still enabling BLE beacons to provide relevant information. The protocol employs cryptographic primitives, including a symmetric key solution based on the AES cipher, to safeguard telemetry data. In contrast, our approach utilizes authenticated encryption from lightweight cryptography, which enables us to transmit tracking data through AltBeacons using smartphones.

The paper [17] introduces a key agreement protocol and authenticated encryption algorithm that are based on Elliptic Curve Cryptography. These cryptographic primitives are used to generate ephemeral beacon identifiers. Nevertheless, it is worth noting that the transmitted data through the beacons are not protected by encryption. In the proposed approach, Raspberry Pi devices are utilized to simulate the behavior of the beacons.

To address privacy concerns in the proposed system, one potential solution is to incorporate k-anonymity mechanisms. This approach has been implemented in several studies, such as [18] and [19], which offer practical solutions for preserving the anonymity of users when using Bluetooth Low Energy devices.

1.4 Paper organisation

The following section (Sect. 2) provides a detailed description of BLE beacon technology, followed by a comprehensive overview of the proposed solution in Sect. 3. This section covers the key components of the system, including the preprocessing and encoding of information emitted by the beacons.

In Sect. 4, we provide a thorough description of the security mechanisms implemented in the proposed system, along with a justification for their effectiveness.

To conclude, the paper presents the main findings of our work, as well as some open questions that require further research.

2 BLE beacon capabilities and limitations

A BLE beacon [2] is a compact wireless device that regularly emits a Bluetooth signal using a broadcasting method and a small advertising protocol data unit (PDU). Although the BLE protocol in beacon mode operates within specific parameters, it is sufficiently flexible to enable users to customize BLE profiles for specific applications. BLE beacons typically transmit small sequences of information, such as a unique device identifier, as well as status information like battery level, temperature, and spatial location. Occasionally, a URL may also be included. Because of this, beacons have historically been utilized in stationary settings for proximity detection, location-based services, and activity monitoring [20]. This allows the creation of scalable systems with minimal power consumption. The signal is transmitted based on a specific protocol, without the need for the beacon and receiving devices to be paired beforehand.

The three main specifications for BLE beacons are iBeacon, Eddystone, and AltBeacon, developed by Apple Inc., Google, and Radius Networks, respectively. These specifications define how the beacons transmit information, including identifiers and sensor data, and how the information can be received and processed by compatible devices. Each specification has its own unique features, such as iBeacon's support for background scanning on iOS devices, Eddystone's support for multiple frame types, and AltBeacon's defines an open-source design. AltBeacon was chosen for this project due to its open-source nature, and the possibility of transmitting larger amounts of data in a single frame packet, up to 26 bytes for payload encoding. Additionally, AltBeacon allows the identification of the application for which the beacons are intended, simplifying device management within the designed tracking application.

When GPS signals are unavailable, using BLE beacons for absolute positioning is generally inefficient. Nevertheless, relative positioning based on proximity is still highly valuable. For instance, motion detection utilizing accelerometers is a fascinating application in this regard. In order to evaluate motion detection by proximity, machine learning analysis of signals is highly suitable, especially for

mobile beacons. Signal analysis is also a highly relevant approach for this type of evaluation.

With their low energy consumption and utilization of the 2.4 GHz frequency, BLE beacons are becoming more prevalent. AFH (Adaptive Frequency Hopping) technology is utilized to ensure compatibility with other WiFi signals [21]. Specifically, when operating in beacon mode, BLE employs channels 37 to 39, thereby minimizing interference with other wireless signals.

The Received Signal Strength (RSS) parameter is a crucial aspect of the beacon protocol, regardless of the profile, as it provides information about the received power (Pr) in relation to the transmit power at the receiver. To determine the location of BLE beacons, the received signal strength indicator (RSSI) of radio frequency (RF) signals are commonly used. While the maximum range of a Bluetooth beacon signal is 150 m, this distance can only be achieved when there are no physical barriers blocking the signal's path between the transmitter and the receiver. In a clear line-of-sight environment, Pr can be utilized to estimate the distance between the beacon and the receiver, as it is inversely proportional to the square of the distance. However, the accuracy of the estimated distance may be affected by deviations in the receiver's measurement sensitivity (RSSI), especially when multiple beacons are in close proximity to each other.

To prevent BLE signals from overlapping with the 14 Wi-Fi channels, BLE strategically uses only 40 channels out of the total 79 allocated to Bluetooth, each spaced equally at 2 MHz, within the 2.4 GHz ISM band. However, in an environment with randomly placed beacons, limiting scanning to a narrow time window may cause a smartphone to miss some beacon signals. Experimental tests in [2] revealed that the RSS signal from multiple beacons can vary considerably over time, even when the beacons are stationary. Under optimal conditions, each beacon requires less than a second to be detected, but detection time can exceed five seconds under unfavorable conditions due to signal propagation and environmental factors, such as relative motion. To obtain stable readings, the speed of the RPA carrying the tracker must be limited, depending on the environment and environmental conditions, and flight altitude. In flight tests, a maximum travel speed of 0.5 m/s and a height of 25 m yielded stable readings.

3 Tracking solution description

This particular study proposes a new and innovative application of Bluetooth Low Energy (BLE) beacons, wherein they can be integrated into a tracking system for mobile objects or individuals while also ensuring the protection of sensitive information.

3.1 Components

The system consists of the following main components:

- Mobile application.
  - Tracker mode: executed by a smartphone on board an RPA.
  - Client mode: broadcasting location and trajectory information.
- Cloud server containing:
  - Server backend.
  - A web application designed to visualize and exploit the data collected.

In our system, the Android application implements two modes of operation: tracker and client. The application running on the clients uses the Bluetooth Low Energy (BLE) beacon mode to transmit information related to its positioning and trajectory (latitude, longitude, height, heading, and speed) along with a randomly generated unique identifier for each client device. The smartphone onboard the RPA will run the application in tracker mode, transmitting the collected data, via 4G/5G, to a web server to be rendered and exploited. The BLE beacon protocol offers several benefits, such as low power consumption and no requirement for device pairing. Nonetheless, the use of this protocol presents significant limitations on the size and structure of information frames that can be transferred, as well as the security measures that can be employed. Therefore, bespoke solutions have been developed to facilitate encryption, encoding, and authentication of information transmitted by clients.

To ensure electromagnetic compatibility, the control systems of the remotely piloted aircraft (RPA) are employed to program flight paths, while utilizing radio frequency control equipment that operates within the 868 MHz ISM band. This approach guarantees the compatibility of the frequencies utilized by the Bluetooth Low Energy (BLE) device, communication frequencies, and the drone's flight tracking control. Furthermore, the real-time synchronization of image transmission with GPS data processing enables effective GPS data processing in conjunction with images by AI in the control center.

3.1.1 Android application

The Android native application has been created to be compatible with BLE and can operate on versions above 8.0. According to [22], this guarantees compatibility with 90.5% of Android devices in January 2023.

The application relies on three background services, each dedicated to a specific task and interconnected with an

information notification feature. This approach streamlines the maintenance and testing of the code, ensuring simplicity and efficiency. The first service is responsible for gathering the GPS location of the system (Fig. 4A), while the second service manages Bluetooth communications (Fig. 4B). The third service enables secure communication with the backend (Fig. 4C).

In addition, the application includes an information notification system (Fig. 4D), a library for encryption and message authentication codes (Fig. 4B), and a user-friendly interface (Fig. 4E).

User management is supported by resources provided by the Parse platform backend and the SDK for communications. The application collects device data and verifies its accuracy by communicating with the platform, thereby granting or denying system access. Depending on whether the user role is client or tracker, the application will detect and respond accordingly.

The application downloads encryption (k_c) and authentication (k_i) keys from the backend, which are then stored in the client device preferences. These keys are generated and distributed by the server, with each beacon being assigned a different list of keys. The key distribution stage takes place before the beacons are deployed.

When a client device changes location, the device uses the implemented library to encode, encrypt, and authenticate the information through a Message Authentication Code (MAC) calculation. The resulting information is packaged and transmitted in Bluetooth beacon mode. This process eliminates the need for mobile network coverage, allowing the application to be used in areas with poor network connectivity.

In tracker mode, the application listens for Bluetooth beacon signals and filters those that contain the identifier defined ad-hoc for the tracking application. Once it detects beacons belonging to the system, it collects all the information found, along with the tracker's current location, and stores them. By storing the location along with the beacon information, the system can provide a more accurate and reliable tracking service.

3.1.2 Cloud server

The responsibility of the application backend is to manage the storage, security, and access control of the application data for legitimate users. This is achieved by using the Parse platform, which includes the Parse server container for managing the project, designing the system's data structure, and implementing specific functions to solve problems. Additionally, a webhook is utilized to enhance the platform's capabilities [23, 24]. The backend architecture comprises five distinct containers, as illustrated in Fig. 5, each serving a specific purpose in the system:

- Proxy NGiNX (Fig. 5A) container handles reverse proxy and load balancing.
- Let's Encrypt NGiNX Proxy Companion (Fig. 5B) container provides automatic SSL/TLS certificate generation and renewal for HTTPS connections.
- Parse Server (Fig. 5C) container hosts the Parse platform, managing data storage, security, and access control.
- Mongo DB (Fig. 5D) container is responsible for database management and storage.
- Webhook (Fig. 5E) container offers additional functionality by allowing external systems to integrate with the Parse server.

Together, these containers enable the efficient and secure management of the application data, while also providing a flexible and scalable architecture for future development.

Parse offers a comprehensive set of user management tools, customizable data structures, and cross-platform software development kits (SDKs) for efficient application development. It also provides the flexibility to extend its

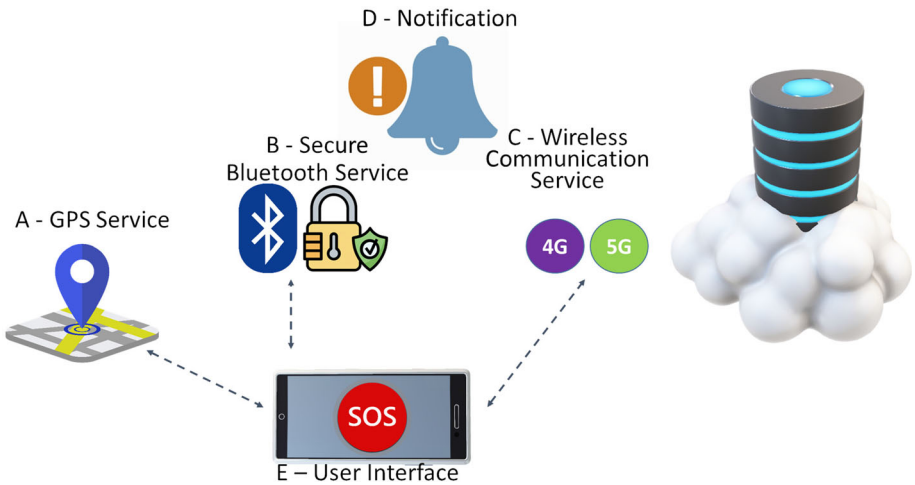

Fig. 4 Android application architecture

Fig. 5 Backend structure

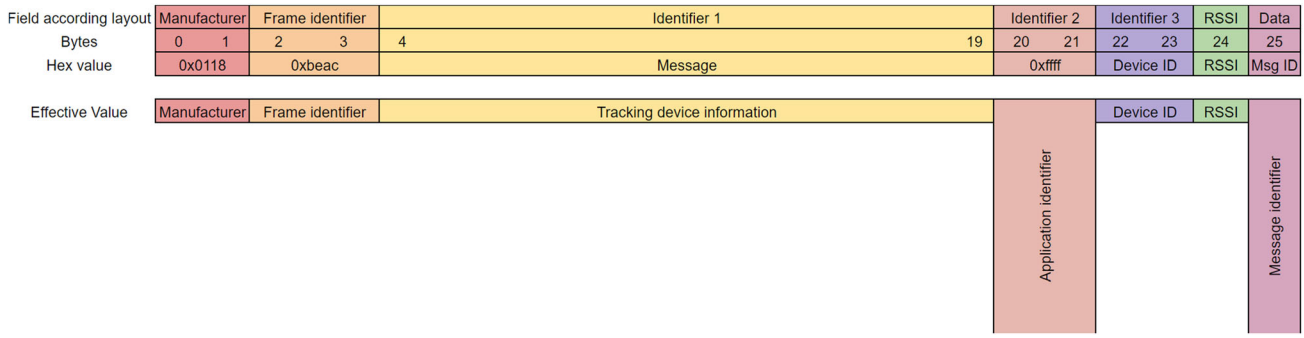

Fig. 6 Customisation of AltBeacon

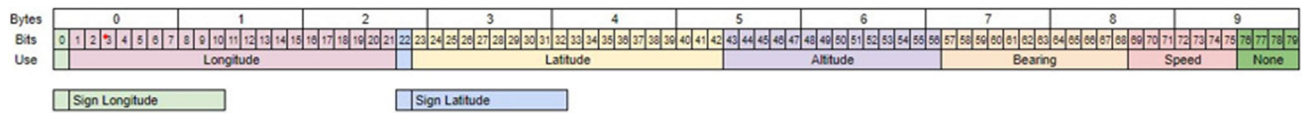

Fig. 7 Tracking parameters codification

capabilities with custom JavaScript code (Cloud Code) and third-party integration via webhooks. In the present project, both Cloud Code and a Python-based custom webhook have been employed.

The Parse platform supports both Postgres and MongoDB. This project uses version 4.4.6 of MongoDB deployed as a Docker container. The Python-based webhook has been designed as a microservice architecture using uWSGI, NGiNX, and Flask tools on an Alpine Linux distribution. Flask is a lightweight web framework that allows fast development of web applications with minimal lines of code, while uWSGI facilitates parallelism and scalability to handle multiple requests simultaneously. NGiNX acts as the web/proxy server to publish the web.

The ultimate objective of the web application is to provide a user-friendly interface for visualizing the data collected by the application. It has been designed as a microservice for ease of deployment on Docker.

3.2 Customizing beacon packet payload

The AltBeacon packet frame format, shown in Fig. 6, comprises a 4-byte header with the following fields:

- Byte 0: A fixed value of 0x02, which indicates that the packet is an advertising packet.
- Byte 1: A fixed value of 0x15, which indicates that the packet uses the Bluetooth LE "Manufacturer Specific Data" advertising data type.
- Bytes 2–3: A company identifier assigned by the Bluetooth SIG to identify the manufacturer of the beacon. For AltBeacon, this value is 0x0118.

These first four bytes enable any device to identify the fields and their lengths. The remaining bytes constitute a variable-length payload, which can be customized for a particular use case. The maximum payload length is 26 bytes.

In addition to the header, the remaining part of the AltBeacon packet is composed of several fields, including three identifiers, a 1-byte value indicating the RSSI, and a 1-byte value reserved for implementing manufacturer-specific features.

In our application, the content of the identifiers has been defined and codified on an ad-hoc basis as illustrated in Fig. 7. The identifiers consist of the following three fields (Fig. 6):

- Identifier 1 (Bytes 4 to 16): contains the encrypted device tracking information.
- Identifier 2 (Bytes 20–21): is a membership code indicating that a given beacon belongs to our system, with a value of "0xffff".
- Identifier 3 (Bytes 22–23): identifies the device that sends the message. This allows for a maximum of 65536 different devices to be considered.

To synchronize decryption, a field called Msg_{ID} is included to transmit a packet identifier.

Data sent by each client via Bluetooth beacon protocol includes longitude, latitude, altitude, bearing, speed, and counter. All of these parameters are collected from the GPS of the client devices. As previously mentioned, the first 16-byte identifier (4-16) contains this data encrypted and authenticated, using the ChaCha20 stream cipher, and the Chaskey algorithm.

The encoding used to pack the data obtained from the GPS, taking into account the range of each parameter, was done as follows (see Fig. 7):

- Longitude: its range goes from 180.0000 to −180.0000. A single bit is allocated for the sign and 21 bits are reserved for the longitude value.
- The latitude has a range of 90.000 to −90.000. To encode it, 1 bit is allocated for the sign and 20 bits for the value of latitude.
- Altitude: 14 bits are utilized for binary encoding, given that its range falls between 0 to 9000.
- Bearing: Its range is from 0.0 to 360.0 and 12 bits are utilized.
- Speed: because the intention of this project is to track devices associated with people who move on foot, a maximum value of 12 m/s was determined for speed. Therefore, the speed range considered goes from 0.0 to 12.0. To encode this variable, 7 bits are used.

All these parameters are represented as integers and then encoded in binary.

4 Security architecture and mechanisms

In order to ensure confidentiality and integrity, depending on the communication channel and device characteristics different solutions have been integrated.

4.1 Client-tracker communication: Lightweight authenticated encryption

As shown in the diagram in Fig. 1, the communication channel used between the clients and the tracker is BLE, using beacon mode, in order to avoid the need to pair the devices to be connected. For this communication channel, confidentiality and integrity services have finally been implemented using cryptographic primitives belonging to Lightweight Cryptography. This part of cryptography is especially recommended for devices with low computational and communication capabilities, such as sensor networks and IoT elements. The decision to use primitives

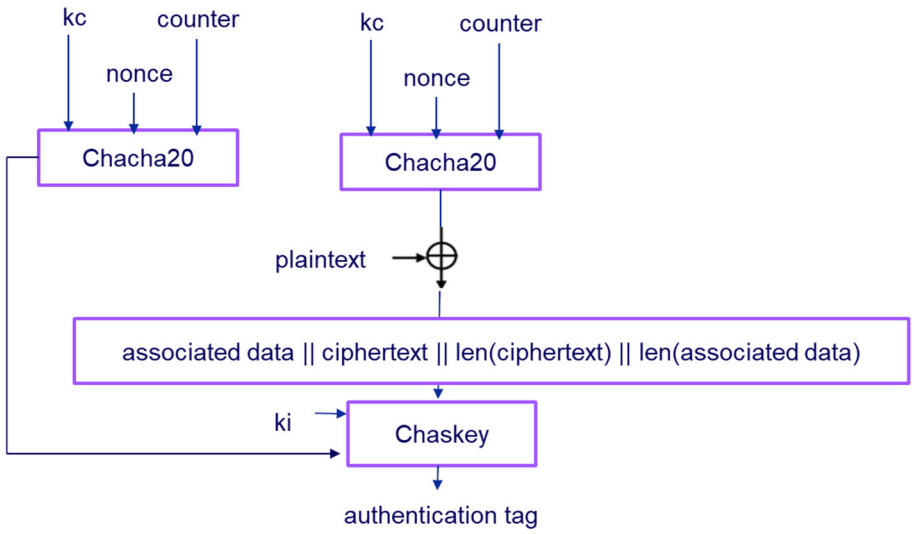

Fig. 8 Implementation of Encrypt-then-MAC approach

from this subset is mainly motivated by the restrictions defined in the communication frames when using Bluetooth beacon mode. Taking also into account the restrictions associated with the described scenario, it has been chosen to use Authenticated Encryption (AE) using the "Encrypt-then-MAC" approach Fig. 8 since it is one of the most suitable methodologies to simultaneously provide confidentiality and integrity through symmetric cryptography [25]. The cipher implemented in the mobile application is the Chacha20 stream cipher [26], developed in 2008 from the Salsa20 cipher. It is based on a pseudo-random generator defined on 32-bit ARX (Add-Rotate-XOR) operations. The key is 256 bits and a counter is included in order to synchronize the streams between the client device and the tracker.

We have opted for a stream cipher due to its suitability when dealing with restricted bandwidth and energy. In [27] authors analyze the latency of commonly used lightweight encryption algorithms by executing them on multiple widely used embedded modules. They also measure power consumption while running these algorithms to understand their impact on battery life. According to [27], ChaCha20 is a more suitable approach for streaming data when the data packet size is sufficiently small.

Chacha20 is usually combined with the MAC Poly1305, but in this implementation, due to the restrictions defined on the Bluetooth beacon frame length, it has been replaced by Chaskey [28]. This algorithm is part of the ISO/IEC 29192-6:2019 standard and is also based on an ARX approach with a key length k_i of 128 bits. The essential features that make Chaskey an appropriate selection for this project are detailed in [29]. For this implementation, the algorithm performs 16 iterations, and the generated MAC is truncated, considering the first 48 bits.

4.2 Tracker-backend communication: TLS

When it comes to communication between the tracker and the backend, security is guaranteed by implementing the Transport Layer Security (TLS) protocol. For deployment purposes, two Docker containers have been defined: the NGiNX Proxy container, which enables external network access to the Parse server, and the "letsencrypt-nginx-proxy-companion" container that interacts with the free certification authority Let's Encrypt to produce and install the required SSL/TLS certificates on the NGiNX Proxy, thus ensuring secure application access.

4.3 Security analysis

This section will assess the effectiveness of the security mechanisms in withstanding various forms of attacks. Before being deployed, the server provides each beacon with a random identifier (ID_b) and two keys: K_c and K_i. The key K_i is a 128-bit key that is used by the Chaskey cryptographic algorithm to generate a message authentication code (MAC) for ensuring message integrity. The key K_c is a 256-bit cipher key used for encrypting and decrypting data transmitted between the beacon and the server. There is the possibility of implementing key rotation updating K_c and K_i if the server preinstalls a list of possible keys in the beacon. In this way, the beacon periodically switches to a new key from the list. The frequency of key rotation is a security parameter that can be determined based on the level of security required and the resources available. This mechanism is commonly used in IoT ecosystems.

Subsequently, we will detail how the aforementioned security measures mitigate the impact of prevalent attacks:

- Brute-force attack: Due to the variability of the information transmitted and the brevity of the beacon packets, attackers are unable to gather enough information to make an educated guess about the keys being used.
- Denial-of-Service (DoS): Packet filtering is implemented based on beacon and application identifiers and there is a limit defined on the rate of messages a beacon can send in a period of time.
- Man-in-the-middle attacks: An attacker attempting to intercept and alter data transmitted between a beacon and a receiver can be thwarted by the use of authenticated encryption. With authenticated encryption, any alteration to the data during transmission would cause the authentication to fail, alerting the receiver that the data has been tampered with.
- Quantum computing: The cryptographic primitives utilized in this work are built on the ARX (Add-Rotate-XOR) construction, which is believed to be resilient against quantum computing attacks, up to a certain level of quantum computing power. Additionally, the Chaskey and Chacha20 algorithms implement appropriate length keys and a large number of rounds, which enhances their capacity to withstand quantum attacks.
- Replay attack: Since the information transferred is authenticated through Chaskey, any attempt by an attacker to tamper with or replay the message will be detected and rejected.
- Spoofing is a security threat in which an attacker attempts to impersonate the beacon by using compromised keys and gain unauthorized access to the network. Key rotation is a security measure that periodically changes the keys associated with the beacon, making it difficult for an attacker to carry out spoofing attacks. Additionally, randomly generating the

beacon identifiers is a valuable security measure that further reduces the likelihood of successful spoofing attacks.

5 Conclusions and open challenges

This work proposes the utilization of beacons and Remotely Piloted Aircrafts (RPAs) to track terrestrial targets from the air. This approach presents a compelling alternative for security and emergency applications, including but not limited to search and rescue missions, event monitoring, and mountain route tracking. This methodology holds great potential for diverse applications by leveraging the strength of AI-driven decision-making capabilities.

The system comprises two key agents, namely trackers, and users. The user's smartphone determines its location via GPS, and subsequently encodes, encrypts, and transmits the data utilizing BLE beacon mode. The tracker, located on a second smartphone aboard an RPA, is responsible for collecting the transmitted data using the BLE beacon protocol through the system's developed app. The tracker then sends the acquired information to a backend server for evaluation and representation. Trackers collect all BLE beacons in range in a synchronized manner, with the collected information being transmitted periodically to the backend. If the RPA loses coverage, the tracker stores the information until coverage is regained. To ensure confidentiality and integrity in data transmission, all communications are encrypted and authenticated, utilizing both BLE and 4G/5G technologies.

The information packet payload has been redefined, and suitable lightweight cryptography primitives have been chosen based on information security requirements and device capabilities. Our system is modular and designed for reusability. However, a more robust architecture for key management is currently under study.

A further issue that needs addressing is the system's reliability in situations where a large number of beacons are present in a confined region. Guaranteeing accurate tracking of all signals will bolster the system's resilience.

It is important to examine the top speed at which the RPA (Remotely Piloted Aircraft) can travel to avoid losing beacon signals while covering a wider area in a shorter time. In this way system's efficiency will be enhanced.

Funding Open Access funding provided thanks to the CRUE-CSIC agreement with Springer Nature. This research was supported by the CDTI (Centre for the Development of Industrial Technology), the Ministry of Economy, Industry and Competitiveness, Celtic-Plus EUREKA and the European Regional Development Fund, under Project IMMINENCE C2020/2-2.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Banani, S., Thiemjarus, S., Wongthavarawat, K., & Ounanong, N. (2021). A dynamic light-weight symmetric encryption algorithm for secure data transmission via BLE beacons. *Journal of Sensor and Actuator Networks, 11*(1), 2.
2. Jeon, K. E., She, J., Soonsawad, P., & Ng, P. C. (2018). BLE beacons for internet of things applications: Survey, challenges, and opportunities. *IEEE Internet of Things Journal, 5*(2), 811–828. https://doi.org/10.1109/JIOT.2017.2788449
3. Spachos, P., & Plataniotis, K. N. (2020). BLE beacons for indoor positioning at an interactive IoT-based smart museum. *IEEE Systems Journal, 14*(3), 3483–3493. https://doi.org/10.1109/JSYST.2020.2969088
4. Gómez-de-Gabriel, J. M., Rey-Merchán, Md. C., López-Arquillos, A., Fernández-Madrigal, J.-A., et al. (2022). Monitoring worker exposure to covid-19 and other occupational risks using BLE beacons. *Journal of Sensors, 2022*.
5. nPerf: NPERF, Coverage 3G/4G/5G Spain (2023/02/13). https://www.nperf.com/es/map/ES/-/-/signal/
6. Wikiloc: Wikiloc: rutas del mundo (2023/02/16). https://es.wikiloc.com
7. Inc., L.: life360 (2023/2/7). https://www.life360.com/
8. Apple: Find Me (2023/2/7). https://www.life360.com/
9. Viswanath, S., Krishnamurthy, R. J., & Suresh, S. (2021). Terrain surveillance system with drone and applied machine vision. *Journal of Physics: Conference Series, 2115*(1), 012019. https://doi.org/10.1088/1742-6596/2115/1/012019
10. Iqbal, M. J., Iqbal, M. M., Ahmad, I., Alassafi, M. O., Alfakeeh, A. S., & Alhomoud, A. (2021). Real-time surveillance using deep learning. *Security and Communication Networks, 2021*, 6184756. https://doi.org/10.1155/2021/6184756
11. Srivastava, A., Badal, T., Garg, A., Vidyarthi, A., & Singh, R. (2021). Recognizing human violent action using drone surveillance within real-time proximity. *Journal of Real-Time Image Processing, 18*(5), 1851–1863. https://doi.org/10.1007/s11554-021-01171-2
12. Kumiawan, F., Cahyani, N. D. W., & Satrya, G. B. (2021). Quantum resistance deep learning based drone surveillance system. In *2021 4th International Conference of Computer and Informatics Engineering (IC2IE)* (pp. 491–495). https://doi.org/10.1109/IC2IE53219.2021.9649188
13. Dabali, K., Latif, R., & Saddik, A. (2022). Conception of a novel drone based on the multispectral camera dedicated to monitoring of vital parameters in agricultural fields. In M. Elhoseny, X.

Yuan, & S.-D. Krit (Eds.), *Distributed Sensing and Intelligent Systems* (pp. 133–145). Springer.
14. Patil, V., Potphode, V., Potdukhe, U., Badgujar, V., & Upadhyaya, K. (2022). Smart UAV framework for multi-assistance. In T. Senjyu, P. N. Mahalle, T. Perumal, & A. Joshi (Eds.), *ICT with Intelligent Applications* (pp. 241–249). Springer.
15. Banani, S., Thiemjarus, S., Wongthavarawat, K., & Ounanong, N. (2022). A dynamic light-weight symmetric encryption algorithm for secure data transmission via BLE beacons. *Journal of Sensor and Actuator Networks, 11*(1), 2. https://doi.org/10.3390/jsan11010002
16. David, L., Hassidim, A., Matias, Y., Yung, M., & Ziv, A. (2022). Eddystone-EID: Secure and private infrastructural protocol for BLE beacons. *IEEE Transactions on Information Forensics and Security, 17*, 3877–3889. https://doi.org/10.1109/TIFS.2022.3214074
17. Campos-Cruz, K.J., Mancillas-López, C., & Ovilla-Martinez, B. (2021). A lightweight security protocol for beacons BLE. In *2021 18th international conference on electrical engineering, computing science and automatic control (CCE)* (pp. 1–6). https://doi.org/10.1109/CCE53527.2021.9633037
18. Chen, Z., Hu, H., & Yu, J. (2015). Privacy-preserving large-scale location monitoring using bluetooth low energy. In *2015 11th international conference on mobile ad-hoc and sensor networks (MSN)* (pp. 69–78). IEEE.
19. Guo, J., Yang, M., & Wan, B. (2021). A practical privacy-preserving publishing mechanism based on personalized k-anonymity and temporal differential privacy for wearable iot applications. *Symmetry, 13*(6), 1043.
20. Bello-Ogunu, E., Shehab, M., & Miazi, N. S. (2019) Privacy is the best policy: A framework for BLE beacon privacy management. In *2019 IEEE 43rd annual computer software and applications conference (COMPSAC)* (vol. 1, pp. 823–832). https://doi.org/10.1109/COMPSAC.2019.00121
21. Pei, L., Liu, J., Chen, Y., Chen, R., & Chen, L. (2017). Evaluation of fingerprinting-based wifi indoor localization coexisted with bluetooth. *The Journal of Global Positioning Systems, 15*(1). https://doi.org/10.1186/s41445-017-0008-x
22. 9to5google: Android 13 is running on 5.2% of all devices five months after launch (2023/02/13). https://9to5google.com/2023/01/18/android-13-device-distribution/.
23. platform, P.: Parse platform (2023/2/7). https://parseplatform.org/.
24. Parse: Parse server container (2023/2/7). https://github.com/parse-community/parse-server#docker-container.
25. Bellare, M., & Namprempre, C. (2008). Authenticated encryption: Relations among notions and analysis of the generic composition paradigm. *Journal of Cryptology, 21*(4), 469–491. https://doi.org/10.1007/s00145-008-9026-x
26. Nir, Y., & Langley, A. ChaCha20 and Poly1305 for IETF Protocols. RFC 8439 (2023/2/7). https://www.rfc-editor.org/info/rfc8439.
27. Sarker, V. K., Gia, T. N., Tenhunen, H., & Westerlund, T. (2020). Lightweight security algorithms for resource-constrained IoT-based sensor nodes. In *ICC 2020—2020 IEEE international conference on communications (ICC)* (pp. 1–7). https://doi.org/10.1109/ICC40277.2020.9149359.
28. Mouha, N., Mennink, B., Van Herrewege, A., Watanabe, D., Preneel, B., & Verbauwhede, I. (2014). Chaskey: An efficient mac algorithm for 32-bit microcontrollers. In A. Joux & A. Youssef (Eds.), *Selected areas in cryptography—SAC 2014* (pp. 306–323). Springer.
29. Choi, S.-K., Ko, J.-S., & Kwak, J. (2019). A study on IoT device authentication protocol for high speed and lightweight. In *2019 international conference on platform technology and service (PlatCon)* (pp. 1–5). https://doi.org/10.1109/PlatCon.2019.8669418.

Candelaria Hernández-Goya was born in Santa Cruz de Tenerife, Spain, on June 18, 1970. She received the M.S. and the Ph.D. degrees in Mathematics from the University of La Laguna, Spain in 1995 and 2003, respectively. She has been Lecturer at the University of La Laguna since 1998 and Senior Lecturer since 2010. Her major interests are security in vehicular ad hoc networks, authentication, and cryptographic protocols.

Ricardo Aguasca-Colomo was born in Las Palmas de G.C., Spain in 1962. He studied Industrial Engineering at the University of Las Palmas de Gran Canaria (ULPGC) and received his Ph.D. in Industrial Engineering from the ULPGC in collaboration with the Department of Electrical Engineering, Electronics and Control of the UNED (Madrid) in 1994. Associate Lecturer at the ULPGC since 1989 and Senior Lecturer since 1996 in the Department of Electronic and Automatic Engineering. He has collaborated in research publications since 1992, including JCR journals, books, and papers in national and international conferences. His areas of interest are related to Electronic Technology, applications in Control and Decision Making using Fuzzy Logic, Reliability/Safety based design, and RPAS integration in emergency management systems. He is currently collaborating in the development of a UAS system for Security and Emergency applications in the Autonomous Community of the Canary Islands. He is currently a researcher at IUSIANI—ULPGC.

Cándido Caballero-Gil is Senior Lecturer in Computer Architecture and Technology at the University of La Laguna, Spain. He received his degree in Computer Science Engineering from the University of Las Palmas de Gran Canaria in 2007 and his Ph.D. from the University of La Laguna in 2011. His research focuses on cryptography, VANET security, especially key management and privacy. He is a member of the CryptULL research group, dedicated to the development of cryptology projects (since 2007), and is involved in several projects and publications in this area. He has authored several conference and journal papers.