BLE beacon system for smartphone secure tracking [1]

Hernández-Goya, C.[1[0000-0002-9468-708X]], Cruz-Rodriguez, D. [1], Aguasca-Colomo, R.[2[0000-0003-2217-8005]] and Caballero Gil, P. [1[0000-0002-0859-5876]]

[1] Dept. Ingeniería Informática y de Sistemas, Universidad de La Laguna, Tenerife, España
[2] CEANI-IUSIANI, Universidad de Las Palmas de Gran Canaria, Las Palmas, España
mchgoya@ull.edu.es

Abstract. Applications with mobile devices are growing exponentially, as their communication capabilities improve. Therefore, it is necessary to guarantee their reliability and security. This contribution presents an aerial tracking application for these devices, with the aim of providing a tool for tracking and tracing in certain situations. To ensure the reliability of the application, information protection mechanisms to guarantee confidentiality and integrity are included. Three main parts have been considered for the application design: a Web application, an Android application and the backend. The system follows a microservice structure over containers, allowing simple management and distribution. Two modes of operation are supported on the Android application, a Tracker Mode, which will be executed on a smartphone on board a drone (RPA), and a Client Mode executed by mobile devices on the ground. Users in Client Mode employ the Bluetooth Low Energy (BLE) beacon mode to transmit information related to their positioning and trajectory. The user obtains this data via BLE and relays it using the 4G/5G network to a web server. The server allows the graphical representation of the data and its exploitation. Deployment is proposed in different scenarios, such as the supervision and control of public areas with capacity control or the tracking and localization of people in isolated environments.

Keywords: BLE beacon, Secure Tracking, RPA, Lightweight Cryptography.

1 Introduction

One of the most remarkable features of Bluetooth Low Energy (BLE) technology is its low power consumption with data transmission rates of up to 2 Mbits/s. This, together with its better range characteristics (up to 100 mt) and Omni-directionality of the signal, make it especially interesting for IoT applications. Among the most common applications are those dedicated to tracking objects in processes, tracking people and equipment, orientation in closed environments, marketing messaging, and where a high rate

[1] Research 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.

of data transfer and accuracy is required. The aim is to carry out these functions securely, as they involve the transmission of sensitive information (positioning of people) in real time.

Currently, there are several applications that allow tracking users using BLE technology on mobile phones. An interesting reference is the Wikiloc application [1], which is capable of tracking outdoor routes. This application allows subsequent consultation of the activities carried out along with some statistics (speed, time, 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.

Also known from the recent situation is the COVID Radar app. The app generates random codes every 10 to 20 minutes and transmits them via Bluetooth to nearby mobile phones that have the app installed. These devices will pick up the codes so that they both record the code from the other device. Each mobile phone stores the codes for a period of 14 days. When someone receives a positive diagnosis and reports it in the app, the codes recorded in the last 14 days are requested. These codes are downloaded daily and allow the app to generate an alert as to whether or not there has been a risk of infection [2].
This paper mainly provides the following contributions:

- Designing an RPA-based system capable of performing surveillance missions using BLE-based technology.
- An infrastructure capable of transmitting GPS information using BLE technology.
- Encrypt and authenticate the transmission of the information collected in real time using lightweight cryptography.
- Store the information into a database for further processing with AI.

This paper is then organized with a summary of published work in this field in section 2, a description of the proposed system in section 3, then a description of the architecture defined is included in section 4, followed by conclusions and future work in section 5.

2 Related work

There are many publications on RPA applications in surveillance and tracking, but most of them use optical sensors and cameras. As indicated below, the main issue with this technology is the correct AI identification and/or classification of real-time video images of potentially dangerous situations. In [3] the drone is used to train a CNN with real-time images to detect road traffic accidents. In [4] a surveillance system using RPAs based on the images acquired by the camera is presented. With these images, a FasterRCNN is trained to identify possible threats such as people carrying weapons. Following the same theme, in [5] the problem of classifying aerial images comprising large areas in order to identify potentially dangerous situations is analyzed in depth. Again, a CNN is trained with a collection of images of offensive situations previously taken from different heights. Regarding the security of communications between ground control and RPA, reference [6] provides an interesting analysis of the security

of communications between the two, both in terms of control orders and images received for subsequent processing using deep learning techniques. In this paper, the minimum hardware necessary to transmit images with a given speed (in fps) using NTRU encryption is tested. Another typical surveillance application supported by RPAs can be found in [7]. In this paper, methodologies for real-time analysis of multispectral images applied to agricultural crops are developed. The aim of this paper is to test different image processing algorithms in conjunction with GPS data. Finally, [8] shows a curious application of a fully autonomous RPA. The system is used to spray disinfectant in public areas, it analyses the images in real, and if necessary, by means of a built-in loudspeaker, it can be warned by emitting an audible message from a voice processing module. The equipment incorporates a BLE module to activate or deactivate the RPA from a mobile phone.

Papers defining security services in the BLE Beacon environment are not very numerous. Given that the main field of applications developed with this technology is limited to marketing and advertising, these papers have generally focused on protecting anonymity, but not on protecting the information transmitted. The proposal included in the Eddystone specification is to use Advanced Encryption Standard (AES) as the encryption scheme, but this alternative is not suitable for devices with as many restrictions as beacons. In [9] 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.

3 System overview

This paper proposes a tracking system based on Bluetooth Low Energy (BLE) beacons for the transmission of data related to the positioning of devices. This data is collected through a mobile phone embedded in an RPA, and security services for the protection of the transmitted information are developed.

Fig. 1. System global view.

The system consists of the following main components (Fig. 1):

- Android mobile application.
  - Tracker mode: executed by a smartphone on board an RPA.
  - Client mode: broadcasting location and trajectory information.
- The backend hosted in the cloud running a web server.
- The web application.
- Security services for communications: integrity and confidentiality.

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 advantages of using BLE beacon protocol include no need for pairing between devices and low power consumption. However, the use of the beacon protocol poses serious restrictions on the size and structure of the information frames to be transferred, as well as on the information protection tools supported. For this reason, all the elements necessary for the encryption, encoding and authentication of the information sent by the clients have been developed ad-hoc.

To ensure electromagnetic compatibility between the different integrated devices, open RPA control systems such as Mission Planner are used to programme flight paths, and radio frequency control equipment in the 868 MHz ISM band is also used. This ensures compatibility between the frequencies used by the BLE device, the 4G/5G communication frequencies and the drone's flight tracking control, as well as the images transmitted in real time.

This allows the AI processing of the GPS data received from the different beacons to be backed up by the images received in the control center, as they are synchronized.

3.1 BLE beacon technology

A BLE beacon is a small device that transmits a Bluetooth signal at regular intervals using broadcasting with a small data payload (advertising protocol data unit, PDU) in a way that allows extremely scalable systems to be defined with limited power consumption. This signal is broadcast according to a certain protocol, without the need for the devices to be paired.

Generally, the information transmitted by these devices contains a device ID, status information (battery level, temperature, spatial information, etc.) and in some cases a URL. Because of this, beacons have traditionally been used in static environments for location applications, proximity detection and activity detection [10]. However, considering that they consume less energy compared to conventional Bluetooth devices and that they use the 2.4 GHz frequency, this paradigm is changing, enabling the development and deployment of applications in multiple scenarios. In this work, a new alternative to the use of beacons is provided, as they are integrated in a tracking system on mobile objects (people in this case).

The most widespread BLE beacon specifications are: iBeacon (proposed by Apple Inc.), Eddystone (defined by Google) and AltBeacon (proposed by Radius Networks). In this development, the latter has been chosen because it is open source and the amount of data available for transfer in a frame packet is higher than the other specifications. In fact, up to 26 bytes can be counted for the encoding of information transmitted as payload. Another advantage is the possibility of specifying an identifier associated with the application for which they are intended, a property that simplifies some issues related to the management of the devices in the designed tracking application.

The frame defined by AltBeacon for a packet is shown in figure 2. The first two bytes (0-1) are used to inform the manufacturer of the beacon, in our case we use AltBeacon's own "0x0118". The next two bytes (2-3) specify the structure that the manufacturer specifies for the rest of the frame. With these first 4 bytes, any device receiving the packet is able to identify the fields and their length.

The rest of the packet is composed of three fields called identifiers, a 1-byte value representing the average received signal strength at 1m. from the transmitter (RSSI) and another 1-byte value reserved to be used by the manufacturer to implement special features.

The structure of the identifiers in our application has been defined and codified on an ad hoc basis as follows:

- Identifier 1 (Bytes 4 to 16): contains the device tracking information encrypted.
- Identifier 2 (Bytes 20-21): is a membership code used to indicate that a given beacon belongs to our system, with value "0xffff".

- Identifier 3 (Bytes 22-23): identifies the device that sends the message. In this way, a total of 65546 different devices may be considered.

The last Msg ID field is used to transmit a packet identifier, which is necessary to synchronize the decryption.

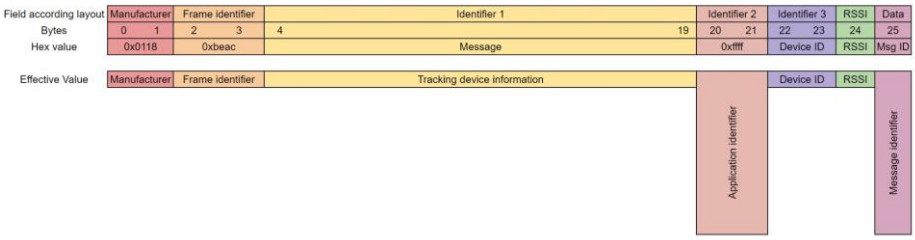

Fig. 2. AltBeacon adaptation.

4 System Architecture

4.1 Android application

The application has been developed entirely with Android Studio and is compatible with Android versions higher than 8.0 in order to use BLE. This ensures compatibility with 82% of the devices currently in use.

Three background services, linked to an information notification, are used for the correct functioning of the application. Each of the services is responsible for a single task, allowing for simple and efficient maintenance and testing of the code. These services are responsible for collecting the GPS location of the system (Figure 3: A), managing Bluetooth communications (Figure 3: B) and communicating with the backend (Figure 3: C). It also contains an information notification system (Figure 3: D), a library for encryption and message authentication codes (Figure 3: E), the user interface (Figure 3: F) and communication with the backend (Figure 3: G).

User management is supported by the resources provided in the backend by the Parse platform and the SDK provided for communications. The application collects the device data and communicates with the platform to verify that they are correct, allowing or not to enter the system. At this point, the application detects whether it corresponds to the client or tracker role. If the client mode corresponds, it downloads from the backend the keys that will be used to encrypt and authenticate the information to be transferred. These keys will be saved in the preferences of the device itself, allowing the application to be used without the need for an Internet connection.

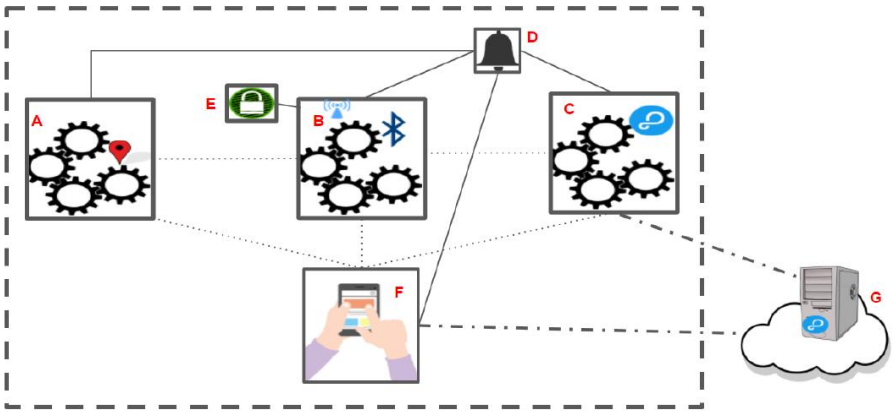

Fig. 3. Android application architecture.

The device running the application in this mode communicates with the C library to encode, encrypt and calculate the MAC of the message when it receives a location update. Once the generated information has been packaged, it starts transmitting this message in Bluetooth beacon mode.

In tracker mode, the application listens for Bluetooth beacons by filtering those containing the identifier defined ad hoc for the application. When it detects beacons belonging to the system, it collects all the beacons found and stores them, together with the location of the tracker itself.

4.2 Web application

The objective of the Web Application is to provide visualisation of the data collected by the application. It is designed as a micro-service for simple deployment on Docker.

4.3 Backend

The application backend is in charge of storing the application data, verifying that it is secure and giving access to it to the legitimate users of the application. Parse platform is used to manage the application backend [11]. The Parse server container is used to manage the project, design the data structure of the system, the functions required to solve specific problems and also allows the use of a webhook to provide it with more features [12].

The backend consists of the following containers (Fig. 4):

- Proxy NGiNX (Fig. 4.A)
- Let's Encrypt NGiNX Proxy Companion (Fig. 4.B)
- Parse Server (Fig.4.C)
- Mongo DB (Fig. 4.D)
- Webhook (Fig. 4.E)

Parse provides user management tools, creation of customised structures and SDKs to be able to work with these features from any platform on which we want to develop the application. It also makes it possible to add functionalities with internal code developed in Javascript (Cloud Code) and the possibility of connecting to external logic via webhook. In this development, the server uses both Cloud Code and a custom webhook, developed in Python.

The Parse platform can use either Postgres or MongoDB. In this case, version 4.4.6 deployed as a Docker container was used. The webhook was created under the micro-service architecture with the Python 3.6 language using the uWSGI NGiNX and Flask tools on an Alpine Linux distribution. Flask is a minimalistic framework that allows us to create web applications quickly with a minimum number of lines of Python code. uWSGI allows us to parallelise and scale the application to be able to respond to multiple requests simultaneously. NGiNX is our web/proxy server, allowing us to publish the web.

In order to protect communications, the Transport Layer Secure Transfer (TLS) protocol is used. Two containers have been defined for the deployment, the first one is an NGiNX Proxy that will be in charge of giving access from outside the network to our Parse server. And secondly, the container "letsencrypt-nginx-proxy-companion", in charge of communicating with the free certification authority Let's Encrypt, generating the corresponding certificate for our system and installing it in the NGiNX Proxy, improves the access to the application.

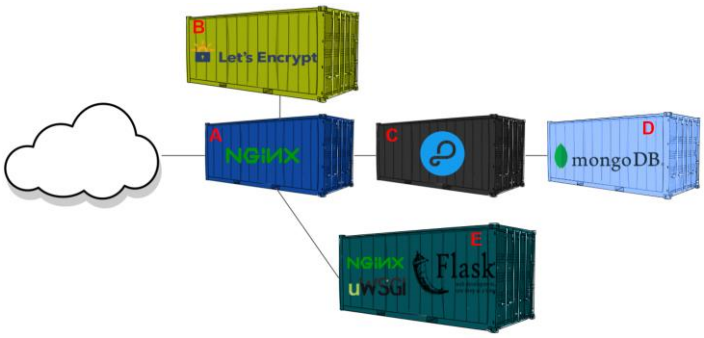

Fig. 3. Backend structure.

4.4 Security

To preserve security, confidentiality and integrity of the information generated and transferred must be guaranteed. Different solutions to these issues have been integrated, depending on the communication channel used and the characteristics of the devices.

As shown in the diagram in figure 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 elements of the Internet of Things. The decision to use primitives from this subset is mainly motivated by the restrictions defined in the communication frames when using Bluetooth beacon mode.

Also taking into account the restrictions associated with the described scenario, it has been chosen to use Authenticated Encryption (AE) using the "Encrypt-then-MAC" approach since it is one of the most suitable methodologies to simultaneously provide confidentiality and integrity through symmetric cryptography [13].

The cipher implemented in the mobile application is the Chacha20 stream cipher [14] developed in 2008 from the Salsa20 cipher. It is based on a pseudo-random generator defined on 32-bit ARX operations. The key is 256 bits and a counter is included in order to synchronise the streams between client device and tracker. It 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 [15]. This algorithm is part of the ISO/IEC 29192-6:2019 standard and is also based on an ARX construct with a key length of 128 bits. Using these constructure for both security services improve efficiency.

For the other communications (mobile application - backend and backend - web application) TLS is used as described in the previous section.

5 Conclusions and future work

The main contribution of this work is the proposal to use beacons and RPAs for tracking and tracing devices. For this purpose, the payload of the information packets has been completely redefined. Furthermore, the security requirements of the information and the capabilities of the devices have been analyzed in order to select the most suitable light-weight cryptographic primitives.

The implementation of the system has been carried out paying special attention to its modularity and possible reusability.

An issue that requires improvements and is being studied is the definition of a robust architecture for key management.

The exploitation of the data obtained with AI is being developed. This will provide a powerful tool for decision-making in situations where response time is a critical variable.

Acknowledgment

Research supported by RTI2018-097263-B-I00: ACTIS.

References

1. “Wikiloc: rutas del mundo”, https://es.wikiloc.com, last accessed 2022/06/15.
2. I. Ramírez, “Radar covid: qué es y cómo funciona la app oficial de rastreo de contactos de España”, https://www.xataka.com/basics/radar-covid-que-como-funciona-app-oficial-rastreo-contactos-espana, last accessed 2022/06/15.
3. Shreya Viswanath et al: Terrain surveillance system with drone and applied machine vision, 2021. In: J. Phys. Conf. Ser. 2115 012019. https://doi.org/10.1088/1742-6596/2115/1/012019
4. Muhammad Javed Iqbal, Muhammad Munwar Iqbal, Iftikhar Ahmad, Madini O. Alassafi, Ahmed S. Alfakeeh, Ahmed Alhomoud, "Real-Time Surveillance Using Deep Learning", Security and Communication Networks, vol. 2021, Article ID 6184756, 17 pages, 2021. https://doi.org/10.1155/2021/6184756
5. Srivastava, A., Badal, T., Garg, A. et al. Recognizing human violent action using drone surveillance within real-time proximity. J Real-Time Image Proc 18, 1851–1863 (2021). https://doi.org/10.1007/s11554-021-01171-2
6. F. Kumiawan, N. D. W. Cahyani and G. B. Satrya, "Quantum Resistance Deep Learning based Drone Surveillance System," 2021. In: 4th International Conference of Computer and Informatics Engineering (IC2IE), 2021, pp. 491-495, https://doi.org/10.1109/IC2IE53219.2021.9649188.
7. 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: Elhoseny, M., Yuan, X., Krit, Sd. (eds) Distributed Sensing and Intelligent Systems. Studies in Distributed Intelligence . Springer, Cham. https://doi.org/10.1007/978-3-030-64258-7_12
8. Patil, V., Potphode, V., Potdukhe, U., Badgujar, V., Upadhyaya, K. (2022). Smart UAV Framework for Multi-Assistance. In: Senjyu, T., Mahalle, P.N., Perumal, T., Joshi, A. (eds) ICT with Intelligent Applications. Smart Innovation, Systems and Technologies, vol 248. Springer, Singapore. https://doi.org/10.1007/978-981-16-4177-0_26

9. Banani S, Thiemjarus S, Wongthavarawat K, Ounanong N. “A Dynamic Light-Weight Symmetric Encryption Algorithm for Secure Data Transmission via BLE Beacons”. Journal of Sensor and Actuator Networks. 2022; 11(1):2. https://doi.org/10.3390/jsan11010002
10. K. E. Jeon, J. She, P. Soonsawad and P. C. Ng, "BLE Beacons for Internet of Things Applications: Survey, Challenges, and Opportunities”, in IEEE Internet of Things Journal, vol. 5, no. 2, pp. 811-828, April 2018, doi: 10.1109/JIOT.2017.2788449.
11. “Parse platform”, https://parseplatform.org/, last accessed 2022/06/15.
12. “Parse server container.”, https://github.com/parse-community/parse-server#docker-container, last accessed 2022/06/15.
13. M. Bellare and C.Namprempre, “Authenticated encryption: Relations among notions and analysis of the generic composition paradigm”, Journal of Cryptology, vol. 21, no. 4, pp. 469–491, Oct 2008.https://doi.org/10.1007/s00145-008-9026-x
14. Y. Nir and A. Langley, “ChaCha20 and Poly1305 for IETF Protocols”, RFC 8439 June 2018, https://www.rfc-editor.org/info/rfc8439, last accessed 2022/7/20.
15. N. Mouha, B. Mennink, A. Van Herrewege, D. Watanabe, B. Preneel, and I. Verbauwhede, “Chaskey: An efficient mac algorithm for 32-bit microcontrollers”, in Selected Areas in Cryptography – SAC 2014, A. Joux and A. Youssef, Eds. Cham: Springer International Publishing, 2014, pp. 306–323.