



Arquitectura LoRaWAN para entornos sin cobertura

Félix Delgado-Ferro, Jorge Navarro-Ortiz, Natalia Chinchilla-Romero, Juan José Ramos-Muñoz

Departamento de Teoría de la Señal, Telemática y Comunicaciones,

Universidad de Granada

C/ Periodista Daniel Saucedo Aranda, s/n. ETSI Informática y de Telecomunicación

felixdelgado@correo.ugr.es, jorgenavarro@ugr.es, nataliachr@ugr.es, jjramos@ugr.es

Este trabajo describe el diseño, desarrollo e implementación de una red pensada para emplearse en zonas sin cobertura como montaña o desierto mediante tecnologías de comunicaciones inalámbricas (Bluetooth Low Energy -BLE y Low Power Wide Area Network -LPWAN). En la implementación se emplean elementos de bajo consumo para la creación de redes de amplio rango de cobertura.

Palabras Clave- Internet of Things, LoRaWAN, Bluetooth Low Energy, Android

I. INTRODUCCIÓN

Las comunicaciones inalámbricas son a día de hoy el tipo de tecnología más empleada en la sociedad. Por este motivo, éstas siguen prosperando y alcanzando límites que años atrás ni imaginábamos. Estos avances nos permiten la comunicación tanto con otras personas como con las máquinas y el entorno.

Hoy en días, existen aún zonas sin cobertura en España, normalmente zonas rurales donde no llegan las operadoras o zonas que se conocen como sombras. Debido a esta problemática y motivado por la participación de grupos de emergencias como Protección Civil en zonas sin cobertura ni conectividad (e.g. Sierra Morena), nos planteamos la posibilidad de crear una red de interconexión móvil para dar soporte a estos equipos de emergencias.

El proyecto se centra en el uso de diversas tecnologías de comunicación inalámbrica como Bluetooth y LoRaWAN para garantizar la conectividad en zonas sin cobertura. Estas tecnologías se seleccionaron debido a que emplean bandas de frecuencias sin licencias (ISM) y presentan características como envío de datos con bajo consumo energético y rentabilidad.

II. ESTADO DEL ARTE

Actualmente, existen algunas soluciones comerciales alternativas para la conectividad en zonas sin cobertura como SPOT X [1] que permite la conectividad empleando un hardware específico al que el móvil se conecta mediante Bluetooth, Uepaa! App [2] que permite envío de mensajes de emergencias utilizando *multihop networks* y Beartooth [3] que permite crear redes amplias empleando hardware específico para el envío de localización y mensajes entre usuarios.

III. FUNDAMENTOS TEÓRICOS

A. Bluetooth Low Energy

Bluetooth Low Energy (BLE) es parte de la versión 4.0 del núcleo de especificaciones Bluetooth [4]. Fue diseñado con el objetivo de optimizar costes con bajo ancho de banda, baja potencia y baja complejidad.

Existen diferentes versiones de dispositivos dependiendo del tipo de conexiones que soporten [5]. Estas son Bluetooth clásico (solo BR/EDR), monomodo (solo BLE) y dual (ambos).

El protocolo Bluetooth Low Energy (BLE) mantiene una estructura de tres partes: controlador, host y aplicación. Estos albergan todas las funcionalidades de su pila de protocolos.

- **Controlador:** gestiona las comunicaciones entre dispositivos BLE mediante maestro-esclavo.
- **Host:** gestiona las claves de seguridad, el intercambio de datos entre aplicaciones y controla las conexiones.
- **Aplicación:** proporciona un servicio al usuario.

B. LoRaWAN

LoRaWAN es una especificación que define tanto el protocolo de comunicaciones como la arquitectura de red. El protocolo LoRaWAN [6] describe las siguientes capas:

- **Capa física:** establece la comunicación y opera en la banda ISM (868 MHz en Europa). Además, utiliza la modulación LoRa para establecer enlaces de comunicación a largo alcance con poca potencia. Esto es posible al emplear el parámetro *spreading factor* (SF) que permite buscar un compromiso entre robustez y velocidad, modificando además el número de canales ortogonales donde transmitir sin colisiones.
- **Capa MAC:** gestiona el acceso al medio, es decir, los canales y parámetros de conexión. En la especificación, existen tres tipos de dispositivos dependiendo de la bidireccionalidad y los tiempos de espera [6] (Clase A, B y C).

Por otro lado, LoRaWAN define una arquitectura de red y las funcionalidades de seguridad que implementa. La red está formada por cuatro tipos de dispositivos [6]:

- **Mota:** dispositivos integrados de comunicación de baja potencia.
- **Gateway:** pasarela de las transmisiones entre motas y servidores de red.
- **Servidor de Red:** autentica los nodos, comprueba la integridad de los mensajes y los enruta a los servidores de aplicación.
- **Servidor de Aplicación:** incorpora mecanismos de confidencialidad de mensajes y gestiona las solicitudes.

Además, permite la activación de los dispositivos mediante activación por personalización (ABP), en la que las claves están pregrabadas en la mota, o mediante activación por aire (OTAA), en la que las claves se generan tras el envío de señalización [6].

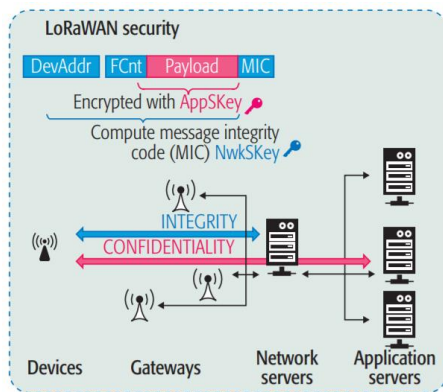


Fig. 1. Arquitectura de la red LoRaWAN [7]

IV. DISEÑO DE RED

La idea fundamental de la red es que permita transmitir información (mensajes y localización) desde cualquier dispositivo móvil al nodo central, incluso si estos se encuentran en zonas sin cobertura.

La red TeamUp se implementa fundamentalmente sobre una arquitectura LoRaWAN. Esta arquitectura será

el núcleo de nuestra red e integramos nuevas funcionalidades y elementos a la red como una aplicación móvil, un servidor propio en el nodo central y una interfaz web de gestión de la red.

Respecto a la conectividad entre los dispositivos que conforman la red y las funcionalidades que cada uno atribuye se especifican a continuación.

- **Smartphone:** integra la aplicación *TeamUp*.
- **Mota:** permite el envío/recepción de mensajes hacia/desde el móvil usando comunicaciones BLE y cambia el mensaje a LoRa para la retransmisión hasta el gateway o viceversa.
- **Gateway:** se encarga de la recepción de los mensajes LoRa de las motas y lo retransmite al nodo central empleando Wi-Fi o viceversa.
- **Nodo central:** permite trabajar en modo estático o dinámico, siempre que ofrezca cobertura al equipo de emergencias. Se encarga de actualizar los servidores (chirpstack & propietario). Estos servidores se emplean para el almacenamiento de los mensajes que se enviaron desde la aplicación móvil, permiten peticiones web para gestión e incluyen funcionalidades de reenvío a otros usuarios.
- **Interfaz Web:** facilita la visualización de los datos a través de un navegador web e incluye funcionalidades como registro de usuario, visualización de mensajes o localización, etcétera.

La red debe permitir el envío de mensajes entre miembros del equipo de emergencias y la visualización para gestión de la red vía web.

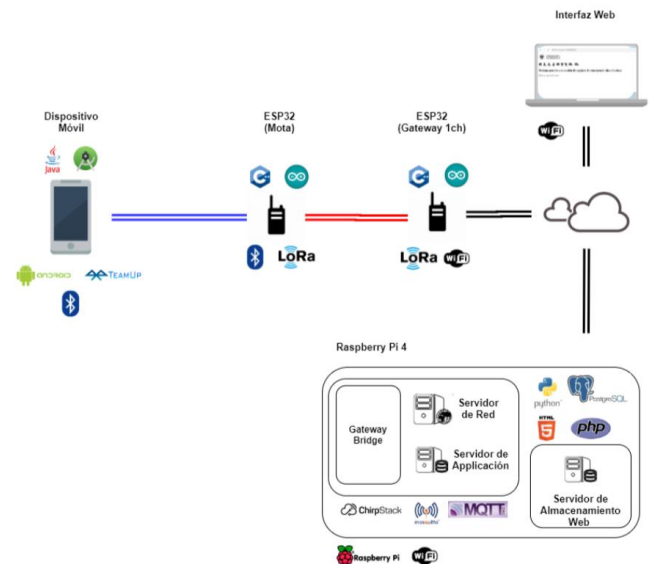


Fig. 2. Diseño de la red TeamUp [8]

V. IMPLEMENTACIÓN DE LA RED

A. App TeamUp

La aplicación TeamUp se ha diseñado de forma que visualmente y funcionalmente sea intuitiva para los usuarios y sea ameno el navegar entre las seis pantallas.

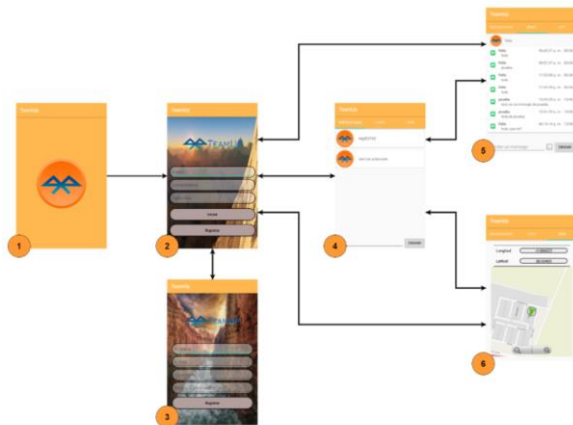


Fig. 3. App TeamUp [8]

Estas pantallas permiten diferentes funcionalidades como registro, login, envío y recepción de mensajes y conectarse a las motas vía BLE. La implementación de estas funcionalidades se ha realizado modularizando dependiendo de la funcionalidad y empleando Android Studio de forma que fuese compatible con el 99% de los dispositivos Android.

B. Motas

Las motas se han implementado empleando diferentes tipos de dispositivos debido a las características que incorporaban cada uno de ellos.

Primero, empleamos ESP32-LoRa que es un dispositivo de clase A y nos permite reducir costes económicos y energéticos, mientras que la cobertura es de gran alcance, pero con baja capacidad de transmisión. Por otro lado, se ha empleado Pycom FiPy que son dispositivos clase C, es decir, son más costosos económicamente y energéticamente, pero permanece escuchando en todo momento y mejora el tiempo de interacción y la latencia.

C. Gateway

El gateway se encarga del envío y recepción de los mensajes LoRa con las motas y la retransmisión mediante Wi-Fi a los servidores. En el despliegue se han empleado tres tipos de dispositivos que ofrecen distintas variedades en la implementación.

- **IMST Gateway Lite:** realiza un barrido de los 8 canales disponibles, obteniendo altas capacidades. Utiliza una Raspberry Pi 1.
- **ESP32 - Single Channel Gateway:** es una implementación más económica, aunque reduce el rendimiento dado que es un dispositivo de clase A. La implementación se ha realizado empleando la librería desarrollada por Marteen Westenberg que encontramos en Github [9].
- **LoPy 4.0 - Nano Gateway:** implementa la misma funcionalidad que el Single Channel Gateway. La implementación es similar, pero empleando librerías de Python [10].

D. Servidores Chirpstack

La plataforma Chirpstack es el núcleo de la red LoRaWAN, implementando los servidores de red y aplicación (además de otras entidades relacionadas) y, por tanto, de la red TeamUp. Estos servidores se encargan de la recepción, descifrado y almacenamiento de los mensajes. Además, implementan mecanismos de seguridad basados en confidencialidad e integridad a dos niveles, es decir, los mensajes mantienen la integridad hasta los servidores de red y la confidencialidad hasta los servidores de aplicación.

La instalación y configuración de los servidores Chirpstack consiste en la instalación de las dependencias y los paquetes software de Chirpstack, la configuración de la organización, perfil de servicio, aplicación, perfil de dispositivo y, finalmente, configurar los dispositivos.

E. Servidor Propietario

El servidor propietario tiene dos funciones que podemos distinguir fácilmente. La primera se encarga de la actualización de la base de datos y reenvío en caso de ser mensajes para el grupo de emergencias. La segunda función consiste en el despliegue de un servicio web para la visualización y gestión de la red por parte del administrador de la misma. Este servidor se ha implementado empleando Python y lenguajes de peticiones web como PHP. Además, este servidor interactúa con la plataforma Chirpstack a través del protocolo MQTT y de la REST API.

VI. PRUEBAS DE CONCEPTO

Las pruebas de concepto se han realizado para comprobar las capacidades de cada uno de las conexiones y la cobertura que podía ofrecer la red. Una vez la red estaba completamente integrada, se han realizado comprobaciones de las funcionalidades. A modo de ejemplo completo, se va a comentar una prueba que consiste en el inicio de sesión de un usuario en la red TeamUp [8]. Esta prueba sigue los siguientes pasos:

1. **Puesta en marcha del servidor:** se inicia el servidor propietario con el comando `python` para que se actualice la base de datos privada constantemente.
2. **Conexión LoRaWAN:** se espera a que la mota y el gateway establezcan la conexión automáticamente.
3. **Conexión BLE:** registramos al usuario en la aplicación y se escanean los dispositivos BLE desde. Posteriormente, se realiza la conexión con la mota. En este momento, se envía automáticamente el mensaje de *Login* para iniciar sesión y guardar el registro en la base de datos. Este mensaje tiene un formato específico: `#L# [usuario] [contraseña]`
4. **Recepción del mensaje Login ACK:** el mensaje llega al servidor y es procesado. Al detectar que se trata de

un mensaje de *Login* se contesta al usuario mediante un mensaje *Login ACK* que tiene un formato #LACK.

5. **Mensajes en la Interfaz Web:** el administrador de la red tiene la posibilidad de visualizar los mensajes y entre ellos, puede ver los mensajes de inicio de sesión con los formatos especificados.

Registro de los mensajes recibidos por el servidor de aplicación LoRaWAN:

id	dev_eui	message	time	direction
1	70b3d54994de968f	#L# Felix 123456	2020-07-03 22:26:36.542257+01	uplink
2	70b3d54994de968f	#LACK#	2020-07-03 22:26:37.339966+01	downlink
3	70b3d54994de968f	Esto es una prueba	2020-07-03 22:27:30.233449+01	downlink

Fig. 4. Mensajes en el Interfaz Web [8]

VII. CONCLUSIONES

Este trabajo presentaba una motivación personal y una alta complejidad debida a la interconexión de miembros de un equipo de emergencias en zonas sin cobertura.

Principalmente, se hizo un estudio de las tecnologías disponibles y se valoraron las opciones que tenían estas en el proyecto, llegando a elegir LoRaWAN por el gran rango de cobertura que ofrece y su bajo consumo energético y Bluetooth Low Energy para la conexión del smartphone dado que éstos no soportan LoRaWAN.

Posteriormente, se diseñó una red que permitiera suplir los problemas de conectividad. La implementación de la red se centra en un núcleo de red LoRaWAN (Chirpstack) al que se han incluido funcionalidades extras como una aplicación móvil, un servidor propietario para gestionar mensajes y ofrecer un servicio web e una interfaz web.

Finalmente, se realizaron un conjunto de pruebas para comprobar el funcionamiento de la red. Ésta funciona como se esperaba en zonas sin cobertura, permitiendo a los usuarios enviar y recibir mensajes a través de la red. Por tanto, afirmamos que la red es 100% operativa, aunque este es el primer paso.

VIII. TRABAJOS FUTUROS

La red está en pleno funcionamiento y ese es el punto perfecto para realizar pruebas y análisis de rendimientos. Las pruebas de rendimiento podemos dividir las en dos experimentos dependiendo de la funcionalidad del mismo.

1. Comprobación del rendimiento y rango de cobertura en distintos escenarios, es decir, se pretende realizar un análisis de rendimiento en función de la distancia para escenarios naturales, rurales y urbanos. Estos análisis se realizarán en las localizaciones marcadas sobre la figura 5.
2. Caracterización del rendimiento dependiendo de la movilidad de la red, es decir, la red es centralizada, pero permite que el nodo central sea dinámico y se desplace. Por tanto, es interesante comprobar como el movimiento tanto de motas como del nodo central a la vez afectan al rendimiento de la red. Obviamente se compararán los resultados con las mediciones en estático.

Aparte de estos análisis que se pretenden realizar, existen posibles mejoras sobre la red o implementaciones alternativas que se mencionan a continuación:

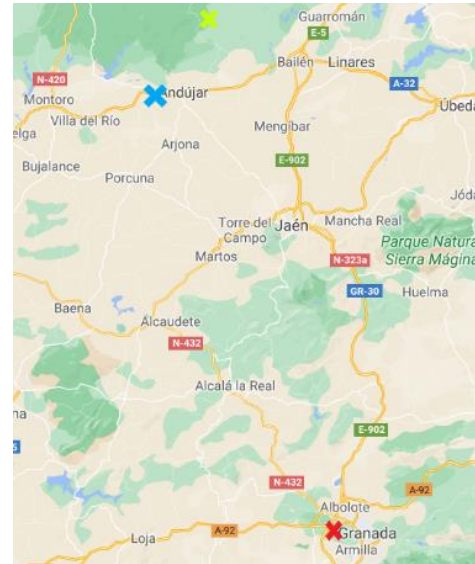


Fig. 5. Localización de las pruebas de rendimiento

- Mejora el diseño de la interfaz web dado que solo se ha llegado a implementar la parte funcional, pero no es atractivo para el usuario final.
- Implementación y comprobación de la ruta de un usuario desde la interfaz web empleando Google Maps, es decir, hacer un *tracking* de los usuarios de la red en tiempo real.

AGRADECIMIENTOS

Este trabajo ha sido parcialmente financiado por la Agencia Andaluza del Conocimiento (proyecto A-TIC-241-UGR18) y el Ministerio de Economía y Competitividad (proyecto TEC2016-76795-C6-4-R).

REFERENCIAS

- [1] SPOT X Globalstar. Disponible: <https://www.globalstar.com/es-la/products/personnel-safety/spotx> (visitado 14-04-2020)
- [2] Uepaa Safety App Losung. 'Die clevere App-Losung fur Alleinarbeiter mit Totmannfunktion. Leichtgewichtige Smartphone App mit 24/7 Notrufzentrale'. url: <https://safety.uepaa.ch/de/> (visitado 14-04-2020)
- [3] Official Page of Beartooth. url: <https://beartooth.com> (visitado 14-04-2020)
- [4] Bluetooth Technology. 'Core Specifications'. Disponible: <https://www.bluetooth.com/specifications/bluetooth-core-specification/> (visitado 14-04-2020)
- [5] Bluetooth Technology. 'Radio Versions'. Disponible: <https://www.bluetooth.com/learn-about/bluetooth/bluetoothtechnology/radio-versions/> (visitado 14-04-2020)
- [6] LoRaWAN® Specification v1.1, LoRa Alliance®. Disponible: https://lorawan-alliance.org/resource_hub/lorawan-specification-v1-1/ (visitado 14-04-2020)
- [7] Navarro-Ortiz, Jorge and Sendra, Sandra and Ameigeiras, Pablo and Lopez-Soler, Juan M. (2018,02). 'Integration of LoRaWAN and 4G/5G for the Industrial Internet of Things'. IEEE Communications Magazine. volumen 56, págs. 60-67. Disponible: <https://ieeexplore.ieee.org/document/8291115> (25-06-2021)
- [8] Delgado Ferro, Félix. 'Interconexión de Miembros de un Equipo de Emergencias en Entornos sin Cobertura usando Dispositivos Móviles'. Disponible: <https://wpd.ugr.es/jorgenavarro/thesis/2020TFG-FelixDelgadoFerro.pdf> (visitado 25-06-2021)
- [9] Version 5 of Single Channel LoRa Gateway, Jac Kersing GitHub Disponible: <https://github.com/kersing/ESP-1ch-Gateway-v5.0> (visitado 20/05/2020)
- [10] Pycom. 'API Reference for Nano Gateway'. Disponible: <https://docs.pycom.io/tutorials/loralorawan-nano-gateway/> (visitado 14-04-2020)