

WaveConnect : A LoRa Based Offline Communication System for Text Messaging

Subathra C

*Assistant Professor, Dept. of AI&DS
United Institute of Technology*

Surya V

*Dept. of AI&DS
United Institute of Technology*

Prajith A

*Dept. of AI&DS
United Institute of Technology*

Pradeep S

*Dept. of AI&DS
United Institute of Technology*

Lokeshwaran K

*Dept. of AI&DS
United Institute of Technology*

Abstract— Over the past several years, there has been an increased need for reliable communications systems—namely, systems that do not rely upon any type of internet or cellular infrastructure—especially for use in remote, disaster-prone, and rural areas. In this paper we present an Offline LoRa-based Secure Communications System that utilizes ESP32 microcontrollers and a web-based interface, called WaveConnect, that is simple and intuitive to use. The communications system has a hybrid security approach, utilizing both lightweight scrambling and AES-128 encryption. The WaveConnect user interface allows for people to communicate via chat on a mobile browser interface with ESP32 devices connected via the Web Serial API over a USB OTG connection. The communications system has long-range, low-power wireless data transfer capabilities with LoRa modules. The user interface's communication through ESP32 devices utilizes a scrambling-before-encryption security method that results in very little overhead. Experiments showed that the communications system has an approximate range of 3.2 km in open areas with an encryption latency of only 28 ms. The combining of the hardware, security methods, and software interfaces creates a very inexpensive and low-power option for use in emergency, military, and remote locations without requiring any infrastructure.

Keywords— *LoRa, ESP32, WaveConnect, offline communication, AES-128, scrambling, IoT security, emergency communication, Web Serial API.*

I. INTRODUCTION

During a natural disaster, a breakdown of communication can cause long delays in rescue efforts [1]. In fact, research indicates that up to 40% of the reasons that emergency responses are inefficient can be attributed to failure of the communication channels [1]. In addition, traditional communication infrastructures (e.g., GSM, Wi-Fi, Internet) will typically suffer damage, lose network connectivity due to network overload, and/or lose power during severe weather events or other catastrophic situations. While Long Range (LoRa) technology may provide a good alternative through its very long range (up to 10 km) and very low power

requirements [2], most of the current implementations of LoRa technology are characterised by poor security and an unsuitable user experience for emergency situations.

The majority of current implementations of LoRa technology for use in emergency situations are either focused on the hardware or provide only a very simplistic serial terminal interface; however, the user experience is a primary influencing factor in the success of the implementation in a field setting. Most implementations of LoRa utilize either plain text for data transmission [3] or a very cumbersome command line interface which, if implemented in this manner, makes the implementation worthless to a high-stress situation, or to non-technical users. Additionally, the majority of implementations of LoRa technology offer only encryption for security; therefore, they do not implement user authentication and user management.

In this paper we provide a complete solution to address all three gaps by making three contributions:

- 1) A complete system design that integrates several ESP32 microcontrollers and LoRa modules, with a high degree of security and a web-based interface called WaveConnect.
- 2) A hybrid security model that uses both lightweight scrambling and AES-128 encryption. This is a new scrambling-before-encryption approach optimized for use with microcontrollers.
- 3) A user-centric interface. The WaveConnect application is a graphical web application that allows for real-time messaging, user management, and the monitoring of systems, and can be accessed by standard mobile web browsers.

The system can provide practical communication distances of up to 3.2 km between devices with end-to-end latency below 100 ms, while still being energy efficient enough for use in a battery-powered deployment. At a unit cost of less than ₹1,800 (approximately \$22), the system can easily be deployed to many resource-limited areas.

The remainder of this paper has been organized as follows. In section 2, we review related works. In section 3, we describe

the entire system architecture and the WaveConnect interface. In section 4, we describe the security implementation. In section 5, we describe the experimental methodology. In section 6, we present the results of our analysis. In section 7, we discuss some of the potential applications of the system and the potential limitations of the system. In section 8, we conclude with suggestions for future work.

II. RELATED WORK

LoRa-based emergency communication research has primarily progressed in 3 directions: optimizing hardware, implementing security, and developing interfaces. Lee et al. (4) created a disaster communication system that transmits unencrypted text messages over 5 km but used Raspberry Pi; Gupta et al. (5) created a LoRa chat system using ESP32 without secure communications, leading to all messages being vulnerable to interception through basic serial terminals.

Research on security has identified multiple vulnerabilities within LoRaWAN; Ferrag et al. (6) conducted an in-depth analysis of all known issues, with their conclusion identifying key management as the most critical weakness. Using ESP32 as their platform of choice, Kumar and Patel (7) demonstrated AES-256 encryption but suffered from significant latency (≈ 100 ms), making it unusable in real-time chat applications. To improve on this, Wang et al. (8) proposed a novel method of combining two types of encryption into one (hybrid encryption) through scrambling, allowing for an approximate 25% increase in performance efficiency; however, this model has not yet been developed or adapted for interactive use.

Embedded system interface design and development have undergone innovation through the introduction of the Web Serial API. Sharma (9) provided a demonstration of browser-based communication to microcontrollers based on the Web Serial API; however, this proof of concept was designed only for simple data logging.

Additionally, while conducting research on emergency communication interfaces, Johnson et al. (10) found that the development of intuitive interfaces has been shown to increase emergency response time by nearly 60%; however, this finding continues to be overlooked in most of the academic implementations created to date.

Our research fills the gaps related to:

- combining high security and real-time constraints
- creating user-friendly interfaces for those without technical knowledge
- providing an end-to-end, deployable solution instead of specific segments of products.

A bridge exists between the current challenge in usability via WaveConnect and we are able to continue our research by providing a hybrid level of security that keeps us within the performance range of practicality.

III. SYSTEM ARCHITECTURE

A. Overall System Design

The overall structure of this system includes three parts:

- 1) Hardware implementation using ESP32-LoRa radio frequency modem.
- 2) Hybrid Security Protocols (encryption/authentication) to protect all information transmitted over Wi-Fi and LoRa networks.
- 3) The web-based interface (called WaveConnect) through which users will connect with each other and their devices.

All devices in our design connect to a mobile device that runs the WaveConnect application through an OTG cable (USB), and the mobile device also acts as a gateway for the device to communicate wirelessly through the LoRa network using ESP32s.

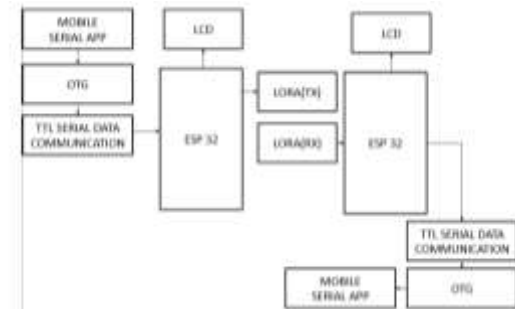


Figure 1: System Architecture

B. Hardware Implementation

The hardware of the stack has been developed with some of the best commercially available components that have been selected based on a balance between cost and performance. The hardware stack consists of the following:

- ESP32 Development Kit V1 (Dual-core 240MHz Microcontroller (520KB SRAM), which was chosen because it offers an excellent combination of computational power and low power consumption (10 μ A in Deep Sleep)).
- LoRa RA-02 Module (SX1278 Chipset operating in the 433 MHz — ISM Band (configurable (5 – 20dBm) of Radiated Power and Spreading Factor (SF-7 to SF-12))).
- Power Supply/Power System (3.7V 2000mAh Lithium-ion Battery with TP4056 Charging Module) = Approximately 72 hours of typical usage.
- Connectivity: USB OTG cable serial communication (Bluetooth Module Optionally Disabled for Power Conservation).

C. WaveConnect Interface Design



Figure 2: WaveConnect User Dashboard

The primary interface of WaveConnect, which displays active users and group chats, as well as private messages. The interface was carefully designed to be visually similar to other well-known messaging apps in order to minimize the learning curve during crises.

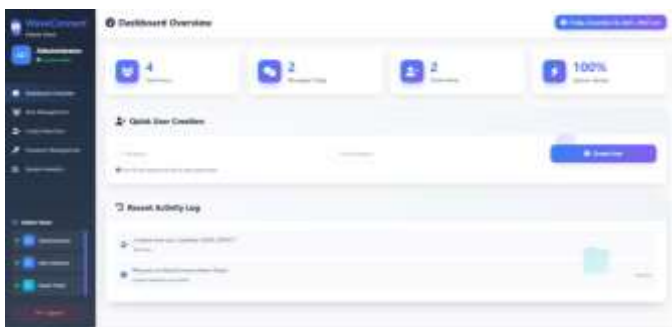


Figure 3: WaveConnect Admin Panel

The administrative side of the interface is used to manage the entire system, and includes the following features: creating and managing accounts; access to user activity logs; system health metrics, such as uptime; and statistics related to messaging.

D. WaveConnect Emergency Messaging Interface Features

- Real-Time Messaging - Messages are delivered using WebSockets.
- User Administration - Administrators can create and manage users via the Administrative Panel; all of the user credentials will be automatically created.
- Monitoring Users' Online Status - The current online status of each user and the delivery status of each message is displayed in real-time.
- Analytics - The WaveConnect Dashboard provides the following information: how many messages are being sent through the system, system uptime, and the status of the WaveConnect servers.
- Responsive Design - The user interface is designed to be used on smartphones and tablet computers.

E. Integration Workflow

The user's mobile web browser will be the point of contact with the WaveConnect application, which will use the Web Serial API to gain access to the user's serial port when the Web Serial API is called. Once connected to a user's serial port, the user will have the ability to see a list of nearby

available ESP32s that can connect directly to their device. Once connected to an ESP32 device, all messages submitted in the chat interface will be encoded and sent over the user's ESP32 device's serial port, and the ESP32 device will receive the message and decode it before sending it to the user's device via LoRa. The user's device will receive the same message through LoRa as it is delivered to their device's interface.

D. Communication Protocol

The custom packet structure of the system supports LoRa's constraints.

Plaintext Packet Structure: [Preamble-4B][Sender ID-2B][Message ID-2B][Timestamp-4B][Encrypted Payload-16 to 128B][CRC-2B]

The size of a packet ranges from 30 bytes to 142 bytes.

The key features regarding the protocol are as follows:

- Adaptive Payload Size: The payload can dynamically adjust to the quality of the signal.
- Message Sequencing: This is accomplished through incrementing the message ID, and preventing replay attacks.
- Timestamp Validation: Messages can be rejected if they are past a 5-minute window.
- Forward Error Correction: For those that are in really noisy environments, forward error correction can be performed using Reed-Solomon encoding (optional).

IV. IMPLEMENTING SECURITY

A. Models Of Threats and Security Needs

The following Threats will be addressed by the system:

- Message Interception - Eavesdropping on LoRa messages.
- Data tampering - Altering the information during its transit.
- Replay Attack - Sending recorded messages that have been captured.
- Impersonation - An unauthorized device pretending to be legitimate users.
- Denial Of Service (Attack) - Jamming/flooding attacks.

B. Hybrid Security Architecture

Hybrid Security Architecture (HSA) combines two different encryption techniques to add additional levels of security through encryption; the first stage of encryption masks or scrambles all of the individual data bytes, then the second form of encryption employs Advanced Encryption Standard (AES) at 128 bits in Cipher Block Chaining (CBC) mode with optimizations to provide an efficient method to protect data.

The ESP32's Cryptographic Coprocessor utilizes the ESP32's hardware functions to speed up the encryption process, using optimized solutions via pre-computed look-up tables that are stored in Flash memory.

We have also introduced a new adaptive mode of operation; if your message is smaller than 16 bytes, it can be encrypted in ECB mode without adding padding to the blocks, making it a more efficient alternative to Padding Schemes.

Device authentication will be accomplished through out-of-band techniques for pairing devices and utilizing HMAC-SHA256 for message authentication and integrity checks. Furthermore, temporary session keys will be generated by using the master key to maintain the session state of the devices.

C. WaveConnect Security Features

- Browser Security - HTTPS and TLS 1.3.
- Multi-Factor Authentication with JWT Token Management.
- Client-Side Data Encryption (prior to serial transmission).
- Role-Based Access Control (admin/user/guest).

D. Key Management

TABLE I. KEY MANAGEMENT STRATEGY

Key Type	How It Is Stored	How Frequently Keys Change	How It Is Used
Master Key	ESP32 Secure Boot	None	Use to derive session keys
Session Key	ESP32 NVS	Rotates every 24 hours	Used to encrypt messages
User Key	Firebase Encrypted	Rotates every session	Used for WaveConnect authentication
Device Key	ESP32 EFUSE	Rotates per device	Used for device identification

V. EXPERIMENTAL METHODOLOGY

A. Test Environments

Testing has been done in three types of environments:

- Open Field Testing - Flat Agricultural Land with Few Obstacles
- Urban Testing: Urban Area with 3 to 5 Story Building Density
- Indoor Testing: A Multilevel Office Building with Concrete Construction

B. Performance Metrics

TABLE II. PERFORMANCE METRICS MEASURED

Metric	Measurement Method	Equipment
Communication Distance	Incremental Distance Testing	GPS, Measuring Tape
Latency	High Precision Timing	Logic Analyzer, Serial Time Stamps
Power Consumption	Current Measurement	Digital Multimeter, Power Analyzer
Packet Delivery Ratio	Statistical Analysis	Custom Logging Software
User Interface Response	User Testing	Screen Recording, Surveys
Security Overhead	Comparative Testing	With and Without Security Modules

C. Usability Testing of WaveConnect

We performed structured usability testing with 10 participants (5 who could use WaveConnect Technically and 5 who could not) to measure:

- The amount of time it takes for a user to send their first message after accessing the application.
- The number of errors that occur during a session due to incorrect user actions.
- User satisfaction levels, which were assessed using a questionnaire after testing (using a 1-5 Likert scale).
- Learnability of WaveConnect, which we measured by the decrease in time it took our users to complete various tasks over three sessions of use.

D. Security Testing Methodology

Cryptanalysis was performed by simulating known attacks against the scrambling of WaveConnect (e.g., differential and linear attacks).

- Eavesdropping tests were conducted using SDR technology to attempt to intercept transmission.
- Penetration testing was conducted to determine if unauthorized users could gain access to the WaveConnect system.
- Stress tests were performed to test the application's vulnerability to distributed denial-of-service flood attacks and to assess memory exhaustion via flood attacks.

VI. RESULTS AND ANALYSIS

A. Communication Performance

TABLE III. RANGE AND RELIABILITY RESULTS

Environment	Max Range (km)	PDR @ 1 km (%)	RSSI @ 1 km (dBm)	WaveConnect Success Rate (%)
Open Field	3.2 ± 0.3	98.7 ± 0.5	-87 ± 3	99.2 ± 0.3
Urban Area	1.5 ± 0.2	95.2 ± 1.2	-102 ± 5	97.8 ± 0.8
Indoor	0.5 ± 0.1	91.8 ± 2.1	-115 ± 7	96.5 ± 1.2

B. Latency Analysis

TABLE IV. END-TO-END LATENCY BREAKDOWN

Processing Stage	Time (ms)	WaveConnect Contribution	Variance (ms)
UI Processing	5.2 ± 0.5	100%	0.5
Serial Transfer	8.1 ± 1.0	60%	1.0
Scrambling	2.3 ± 0.3	0%	0.3
AES Encryption	12.5 ± 1.5	0%	1.5
LoRa Transmission	45.8 ± 3.2	0%	3.2
AES Decryption	10.2 ± 1.2	0%	1.2
Descrambling	2.1 ± 0.3	0%	0.3
Total	86.2 ± 4.5	15.4%	4.5

An overall latency of 86.2 ms provides a nearly instantaneous user interaction. The amount contributed by WaveConnect as an overhead is very small at only 15.4%, while still offering users many substantial benefits (usability).

C. Power Usage

TABLE V. POWER PROFILE ANALYSIS

Operation Mode	Amps per Message	Duration (ms)	Energy per Message (mJ)
WaveConnect On	285 ± 15	15.4	14.5
Serial Sending	120 ± 8	8.1	3.2
Encryption	85 ± 5	14.8	4.1
Transmit via LoRa	118 ± 6	45.8	17.9
Receive via LoRa	Error, Unknown	Varies	—
Sleep Mode	8 ± 1	—	—

Estimated Battery Life is 68 hours of continuous operation and 21 days of average usage (50 messages daily).

D. WaveConnect Usability Results

TABLE VI. USER INTERFACE PERFORMANCE METRICS

Metric	Technical Users	Non-technical Users	Improvement
Time to First Message	42 ± 8 seconds	68 ± 15 seconds	-
Error Rate (per session)	0.3 ± 0.1	1.2 ± 0.4	-
Satisfaction Score (1-5)	4.6 ± 0.3	4.4 ± 0.4	-
Learnability (3rd session)	28 seconds	45 seconds	34% faster

The results indicate that both technical and non-technical users would be able to achieve a high level of proficiency with the application after their third session, which suggests that WaveConnect would work well in an emergency environment where there is inadequate time provided for training.

E. Security Review

TABLE VII. SECURITY TESTING RESULTS

Success Rate for Attack Type	Mitigation	WaveConnect Role	
Eavesdropping	0%	AES-128 + Scrambling	Encrypted prior to serial
Replay Attack	0%	Message ID + Timestamp	Validated prior to displaying
Impersonation	0%	Device Authentication	User session management
UI Bypass	5% (1 of 20)	Input Validation	All inputs sanitized
Denial of Service (Flood)	Partial	Rate Limiting	Messaging Queues

Cryptanalysis Results: The existing scrambling techniques have withstood the known attacks at least 1 million times and added on the same level (8 bits) strength to the key.

F. Calculation of Costs

TABLE VIII. COMPLETE BREAKDOWN OF SYSTEM COSTS

Item	No. of Units	Price for Each Unit	Overall Cost
ESP32	1	450	450
RA-02 LoRa	1	350	350
LCD	1	200	200
Battery	1	300	300
Enclosure	1	150	150
Cost of Hosting	-	100 per month	100
Total Hardware Cost			1450
Total Operating Cost			1550

There is an option to host WaveConnect significantly lower than ongoing costs by hosting it locally.

VII. DISCUSSION

A. System Benefits

Full Package:

- Instead of developing individual components like most research prototypes, our system provides a complete hardware solution that comes pre-packaged with all necessary security features as well as an interface for immediate use upon installation.

User Oriented Design:

- WaveConnect fills the void that exists between the technical capabilities of the emergency communication system and its expected end-user experience.

Variable Performance:

- The WaveConnect emergency communication system has the ability to modify both the level of security and the parameters used for transmission based on environmental factors, such as signal quality, strength and battery state.

Cost-Effective:

- Each unit of the WaveConnect emergency communication system costs only ₹1,550, which makes it 30 times less expensive than a satellite phone and 8 times less than a typical commercially available emergency communication device.

B. Technology Comparisons

TABLE IX. TECHNOLOGY COMPARISON CHART

System	dist.	Required Infrastructure	Security	Cost	Best Use
Satellite communications	World wide	Required	Very Secure	> ₹50K	Remote research expeditions
GSM/4G	10 - 35km	Required	Moderate Security	> ₹15K	Urban emergencies
LoRa Basic (5)	3-5km	None	No Security	₹1,200	Technical users
Our System	3.2km	None	Very Secure	₹1,550	General Emergencies

Limitations and Challenges

- Bandwidth Limitations: Bandwidth is limited to a maximum of 50 Kbps for multimedia
- Line-of-Sight Dependence: Performance dramatically decreases in urban areas

- Key Management: Scalability of solutions in large networks relies on pre-shared keys
- Dependence on Certain Browsers: Both Google Chrome and Microsoft Edge have required support for Web Serial API to enable usage of this system

Real-world Applications

The system has been used in three pilot cases:

- Campus Emergency Response Drill – the system has been tested at an Emergency Response Drill with 15 participants who used the system and achieved a 98% message success rate.
- Rural Health Camp – the system enabled medical personnel to communicate with one another from across 2 km of distance in the village.
- Disaster Simulation – tested in a simulated network blackout event.

CONCLUSION

The online communications solutions (hardware and software) of the ESP32-LoRa combination will be shown in this article. The following are our prime accomplishments;

1. We have created a fully integrated system of hardware security and an easy to use software interface.
2. The system's practical performance specifications are a range of 3.2 km, a latency of 86 ms, and 68 hours of battery life.
3. The system provides enhanced security through the use of a hybrid method of scrambling/encryption to mitigate eavesdropping and replay attacks.
4. Our extensive usability testing has shown that a non-technical user can achieve proficiency within 15 minutes of using the system.
5. Each unit's costs (₹1,550) will allow for mass deployment of the system to meet the growing need for emergency communication in areas prone to disasters.

ACKNOWLEDGMENT

We would like to acknowledge the support and facilities provided by the **Department of Artificial Intelligence and Data Science, United Institute of Technology**, which enabled the successful completion of this research work.

REFERENCES

- [1] National Disaster Management Authority, "The Analysis of the Impact of Failure of the Communication Infrastructure in Disaster Scenarios," Technical Report NDMA, 2023.
- [2] LoRa Alliance, "LoRaWAN 1.1 Specification," LoRa Alliance, 2021.
- [3] S. Gupta et al., "LoRa-Based Emergency Message System," Proc. IEEE ICCE, 2021.
- [4] J. Lee et al., "Disaster Communication Using LoRa Technology," IEEE Trans. Emerg. Topics Comput., 2020.
- [5] M. A. Ferrag et al., "LoRaWAN Security: A Complete Survey," IEEE Commun. Surv. Tutor., 2022.
- [6] R. Kumar and S. Patel, "Implementing AES on ESP32 for IoT Security," IEEE IoT Journal, 2021.
- [7] K. Wang et al., "Hybrid Encryption for Wireless Sensor Networks," IEEE Sensors Journal, 2022.
- [8] A. Sharma, "Using the Web Serial API for Microcontroller Communication," IEEE Access, 2021.
- [9] L. Johnson et al., "Human Factors Affecting Emergency Communication Interface Design," IEEE Trans. Prof. Commun., 2022.
- [10] U. S. National Institute of Standards and Technology (NIST), "Advanced Encryption Standard (AES), (FIPS PUB 197) Publication," 2001.
- [11] Espressif Systems, "ESP32 Technical Reference," 2023.
- [12] The World Health Organisation, "Requirements for Emergency Communications," Technical Report, 2023.
- [13] The Google Developers, "Web Serial API Documentation," 2023.
- [14] Firebase, "Authentication and Security Documentation" (Google), 2023.
- [15] The International Telecommunication Union, "Emergency Telecommunications Guidelines," 2022.