Alexandria Engineering Journal (2022) 61, 8997–9009

HOSTED BY

Alexandria University

Alexandria Engineering Journal

www.elsevier.com/locate/aej
www.sciencedirect.com

LoRaWAN End Nodes: Security and Energy Efficiency Analysis

Miralem Mehic [a,b,*], Mugdim Duliman [a], Nejra Selimovic [a], Miroslav Voznak [b]

[a] Department of Telecommunications, Faculty of Electrical Engineering, University of Sarajevo, Zmaja od Bosne, Kampus Univerziteta u Sarajevu, 71000 Sarajevo, Bosnia and Herzegovina
[b] Dept. of Telecommunications, VSB - Technical University of Ostrava, 17. listopadu 2172/15, 708 00 Ostrava, Czech Republic

Received 2 June 2021; revised 12 February 2022; accepted 12 February 2022
Available online 28 February 2022

KEYWORDS

LoraWaN;
Eenergy;
Security;
Networks;
IoT

Abstract With the development of electronics and communication techniques, the interest in realizing sensor networks with a large number of end nodes is growing. The main idea is to install devices in remote locations without direct supervision, which requires an uninterrupted power supply and secure communication to the rest of the network. In this paper, an experimental comparative analysis of popular practical LoraWAN end nodes (WisNode RAK811 and Seeeduino SX1301) regarding energy consumption in different security modes, spreading factors, energy consumption in sleep/idle mode as well as the security keys extraction from memory was performed.

1. Introduction

There has been noticeable progress in developing new sensor devices and technologies that have applications in various fields such as smart cities, ecology, security, transport, logistics, industrial automation, agriculture, etc. Internet of Things (IoT) has become synonymous with applications and systems capable of collecting, transmitting, and analyzing data from sensors located in a wider geographical area without human influence. IoT networks are characterized by limited resources such as memory and energy resources. Collecting data from a wider geographical location is challenging since, in most cases, it is not possible to establish physical connections between all devices. Therefore, the need for new solutions in wireless communication was underlined.

One of the primary issues in catering to (IoT) use-cases is energy. The energy-intensive operation of sensor end-nodes positioned in a hostile industrial environment or inaccessible locations (e.g., in many industrial monitoring use-cases) makes regular battery replacement impracticable. Furthermore, these batteries are a disposable resource with negative environmental consequences. On the other hand, while an ideal sensing interval to create warnings can help minimize rapid battery drain, even a tiny delay in displaying an urgent alert might result in many damaged goods, wasting valuable resources on the production line. The issue becomes much more acute when the manufacturing expenses of the created items are very high, and prompt detection of numerous anomalies at various stages of production can save a smart industry from significant

* Corresponding author at: Department of Telecommunications, Faculty of Electrical Engineering, University of Sarajevo, Zmaja od Bosne, Kampus Univerziteta u Sarajevu, 71000 Sarajevo, Bosnia and Herzegovina.
E-mail address: miralem.mehic@ieee.org (M. Mehic).

Peer review under responsibility of Faculty of Engineering, Alexandria University.

https://doi.org/10.1016/j.aej.2022.02.035

financial losses. But, in addition to the importance of timely data collection, it is equally important to make sure that data is unaltered and they were not created as a result of a security attack. Therefore, it is necessary to consider security mechanisms for communication protection. The increased number of messages needed to establish session keys and other security values can further increase the consumption of limited battery resources. Thus, choosing between energy-efficient operation and continuous security monitoring during the manufacturing process, which are two incompatible aims, necessitates a tight line trade-off.

One of the network solutions developed for wireless connectivity in a wide area is Long Range Wide Area Network (LoRaWAN), characterized by low power consumption and long-range using battery power. It uses a newly developed modulation at the physical level called Long Range (LoRa), which communicates at ranges up to 50 km in rural areas. In terms of security, encryption techniques are applied on both the network and application layer to reduce energy consumption. Many industrial solutions on the market differ in implementation, performance, and price.

2. Related Work and Contribution

In [1], the energy efficiency of LoRaWAN nodes was considered. The LoRaWAN technology is suitable for the Industrial Revolution 4.0 since it works in unlicensed spectrum provides minimal uplink latency, cost-effectiveness, and low power consumption. There are two states during the operation of end nodes: sleep and active state. The authors calculated the average battery consumption considering the Semtech nodes assuming a transmission interval between the 60s and 300s. As a result, the daily battery consumption is up to 5.5 times lower when the node sends every 300s compared to the 60s sending interval. Also, as the sending interval increases, so does the number of years of battery life. When it comes to different signal emission powers for a transmission interval of 300s, it was obtained that for a power of 13 dBm the battery life reaches eight years, while for an emission power of 28 dBm, the battery life drops to 2 years. When comparing different propagation factors, the authors found that increasing Spreading Factor (SF) reduces the number of messages in the network for the different duty cycles.

In [2], the authors performed experiments using LoRaWAN nodes without batteries, using capacitors that they charged using renewable energy sources. They concluded that a model with a specifically selected configuration of parameters (capacitor size, switching threshold) and different applications (sending interval, packet size, etc.) can achieve continuous node communication. It is recommended to use capacitors of lower capacity and lower SF because these capacitors charge faster. They showed that a capacitor with a 4.7 mF and SF7 could support packet sending every 60s at a power collection rate of 1 mW.

The paper [3] combined the energy efficiency information of the nodes. They also performed experiments independently using a power meter, which was used to observe the node's energy. A node for different SFs has the same behavior and state changes, except the time durations of those states.

The impact of activation and poor conditions in the channel on the energy efficiency was analyzed in [4]. The authors have shown that poor channel quality leads to losses and placing the node on hold until reactivation. By reactivating Over-The-Air Activation (OTAA) and exchanging keys, the node consumes 11% more energy. They also considered the scenario of a different number of channels to use. They concluded that more channels reduce the possibility of interference and thus less power consumption because the nodes do not send repeated activation requests. They came to the same conclusion when the number of nodes in the network increased.

The authors [5] performed an analysis of the energy efficiency of the use of LoRa nodes in traffic. They observed different baud rates (from Dynamic Range (DR) 0 to DR 4) in combination with varying payload sizes. They used a 1000mAh battery and achieved different speeds with varying SF and spectrum width combinations. The power consumption decreases as the data speed increases, but the opposite is obtained when the size of the payload increases, related to the theoretical conclusions. The battery life for DR1 (SF 9, width BW 125 kHz) at different transmission intervals of 1 to 6 min was calculated, providing the estimated lifetime value of 306 to 1790 days.

The paper [6] analyzes security attacks on battery energy efficiency. The authors carried out a Denial of Service (DoS) end node attack by flooding the network with another device, interfering with sending packets to Gateway (GW). The authors measured battery consumption without any attacks with SF 7 and SF 12, where consumption on SF12 is 18 times higher than SF7. The measurements were repeated with a DoS attack. The obtained results showed that the battery consumption, in this case, increased as much as five times, which drastically reduces the battery life.

The paper of [7] test security vulnerabilities in LoRaWAN networks providing the following conclusions:

- the authors recommend using OTAA activation mode, protection from physical access to the device due to possible destruction, reboot, etc.
- to prevent packet eavesdropping, use random packet numbering in relation to monotonous magnification, thus avoiding the possibility of restarting the device and monitoring the counter magnification.
- if the attacker has control over GW can selectively forward packets to the application server or generate packets.

The authors of [8] state that the transmission time in LoRaWAN networks is a critical parameter because the signal transmission can go up to 1.5 s. If an attacker has physical access, the GW can intercept keys sent via a radio processor via Serial Peripheral Interface (SPI) or Universal Asynchronous Receiver Transmitter (UART) to the main microprocessor. The signal interference can cause any end node by sending packets at a specific frequency and with a particular SF. Although LoRaWAN is very robust, however, if it is about the same frequency and the same SF, problems arise [31].

The paper [9] provides the following conclusions when it comes to security:

- GW cannot be trusted because it is outside the controlled area. If an attacker gains control of the GW he can access the internal network. For this reason, it is recommended

to use a Virtual Private Network (VPN) server that allows encryption and authentication for GWs connected to the internal network.
- The network server is responsible for network control and MAC commands (session management, acknowledgments, duplicate packet elimination, Adaptive Data Rate (ADR)). The network server communicates with multiple entities such as GW, application server, and database. As the network server also stores the network key (NWkSKey), it is recommended to use a firewall that restricts all unnecessary communication except to its entities.
- The application server is responsible for managing all keys in the network (generation and storage). Therefore, it is necessary to secure the application server from intrusions by implementing functions that check payload, whether it meets the criteria (length, format, range, and allowed characters). As for the network server, so for the application, it is necessary to enable a firewall that provides connection only to the network server and database.
- network access management via SSH for authorized users only

In [10], the authors focused on security when activating nodes in the OTAA way. They state that the process of joining/activating a node consists of exchanging two messages, a request for joining (join request) and joining accepted (join accept), where *join request* message is unencrypted. This message consists of a random number DevNonce where the same number is likely to be generated, as shown by the example of the WiMOD SK-iM880A. The same DevNonce was obtained by interfering with high power and attenuating the power of the broadcast to get an identical Received Signal Strength Indicator (RSSI) each time the node ignited. This is possible with cheaper nodes, i.e., for the SX1272 node, a random sequence is generated based on the received signal strength. The bottom line is that device manufacturers should pay attention to generating DevNonce to the pure random number each time a node is turned on, which would improve protection against DoS attacks.

The authors of [11] carried out attacks of sniffing and misrepresentation (Sniffing and Spoofing) where they aimed to show how, when performing an attack and reporting nodes to the network, it is possible to intercept sensitive data and how to identify that an attack is in progress. They performed the identification based on the RSSI value because identifying with the DevNonce number is unreliable. They considered two different cases, with a fixed node and a mobile position. It is much easier to identify an attack in a fixed position because the RSSI value of the actual node is always the same and is stored on the server; however, during node mobility, the RSSI value of the actual nodes itself changes. For this reason, the authors have suggested that in addition to the DevNonce and RSSI values, there is a rule between the node and the server that only they know.

2.1. Our Contribution

Most research in the field of LoRa/LoRaWAN energy efficiency focuses on modeling energy performance. Detailed analysis of approaches based on the datasheet or empirical measurements for considering consumption in sleep and traffic sending/receiving states is available in[12]. These reports offer detailed analyzes of commercially widely available transceivers such as Semtech SX1272, Semtech SX1276, and Microchip RN2483. Reports include experimental measurement data for different indoor distances [13] and device settings [14], while Bouguera [15], Casals [12] and Sherazi [16] developed the analytical energy model of LoRaWAN end-devices. However, they do not take into account the safety aspects of communication. Our paper considers the impact of using security OTAA and Activation By Personalization (ABP) modes on energy efficiency. A Survey on the Noncooperative Environment in Smart Nodes-Based Ad Hoc Networks is available in [17].

In [18–21], a detailed theoretical overview of security issues, threats, and possible mitigation techniques is given. However, these papers only consider the security aspect of communication without considering energy efficiency.

Based on our knowledge and available literature, our work represents a unique contribution to the experimental consideration of the impact of different safety settings on energy efficiency by considering end devices based on Seeeduino (Semtech SX1301) Wisnode RAK811 end devices.

The rest of the paper is organised as follows: Section 3 describes LoRaWaN architecture and its main components. The outline of our experiments is given in Section 4. We discuss the obtained results in Sections 5and 6, while Section 7 concludes our study.

3. LoraWaN overview

It is important to distinguish between the term LoRa, which refers to the physical layer, and LoRaWAN (open standard), which refers to multiple layers: MAC, network, and application layer.

LoRa is a spectrum propagation modulation whose main characteristics depend on the SF, Bandwidth (BW), and Coding Rate (CR). The spreading factor is the ratio between the symbolic and the chip rate $SF = log(R_c/R_s)$ with values ranging from SF = 7 to SF = 12. The spreading factor provides a compromise between the communication speed and range with the technique of forward error correction (FEC) [22,23]. With increasing SF, the transmission duration and sensitivity increase, but the transmission speed decreases. [24,25]. The CR code rate defines the level of Forward Error Correction (FEC) in the LoRa framework with values ranging from 0 to 5. Increasing SF and CR also increases transmission time (Time on Air (ToA), T_{air}). By increasing the BW (125, 250, 500 kHz), less ToA can be achieved, but the receiver's sensitivity decreases. LoRa uses different frequency bands in different regions of the world, such as US (902 to 928 MHz), Europe (863 to 870 MHz), and China (779 to 787 MHz). It uses 125 kHz bandwidth for signal transmission. Using not too narrow channels allows LoRa to show robustness towards some channel characteristics such as selectivity and the Doppler effect.

LoRaWAN is designed from the bottom up to optimize Low Power Wide Area Networks (LPWANs) [26] whose devices can run for up to several years on battery power and with a range of up to almost 50 km in rural areas. They use 128-bit AES encryption, where security is ensured by cross-checking end nodes, data authentication, retransmission pro-

tection, and integrity [27]. The system architecture and protocols are developed by the non-profit LoRa Alliance, which covers a wide range, from chipmakers to cloud providers.

3.1. LoRaWAN data layer

The topology in LoRaWAN networks is not a standard star topology. Nodes can be connected to one gateway with these networks, which is the standard star topology. Still, one of these nodes can also be connected to another gateway, creating a new topology called star-of-stars, as shown in Fig. 1.

Communication between end devices and the gateway is based on frequency changes during radio transmission, so-called Frequency Shift Keying (FSK), using different spectrum spreading factors. The end device in LoRaWAN operates without synchronization, which means that at any time, via any available channel and at any speed, it can send data [27].

3.2. Adaptive Data Rate

LoRaWAN supports ADR, which allows the Network Server (NS) to dynamically change node parameters such as transmit power, frequency list, SF, and retransmission [27,28].

The choice of transmission speed (0.3 to 50 kbps) can affect the range and duration of the transmission. A lower speed achieves a more extended range, but it also takes more time to transfer data [27]. Fig. 2 shows the impact of the ADR scheme, which maximizes battery life and network capacity utilization.

A higher SF increases sensitivity and range but also extends transmission time. The disadvantage of longer transmission times is the risk of collision. Adjusting the transmission power achieves a near-far effect, where it is ensured that the end devices will use less energy to communicate with the GW if they are closer to the same, where unnecessary interference for other end devices is reduced [27,26,28]. The ADR also allows the gateway to be managed by limiting the number of nodes that can send data to it, thus reducing the need to receive redundant data. Redundant data are those that the other gateway, which is less "employed", also receives and processes. [27,28].

3.3. Security

Since the beginning of LoRaWAN development, security has been a critical part of development. The LoRaWAN security design uses standard, well-tested algorithms and end-to-end protection, as well as symmetric cryptography [27]. As shown in Fig. 3, two security levels are used: one for the network layer (device authenticity) and the other for the application layer (the user can only read data) [26]. The primary security features of the LoRaWAN system are mutual authentication, integrity protection, and confidentiality.

For the end node to send data to the network, it must be authenticated and activated. There are two ways to activate the end node: via OTAA or ABP [29]. The activation methods are briefly described below: [30]:

- **OTAA** - Contains the exchange of two messages between the end node and the server. The first message is the *join request* which is sent by the end node to the NS. It contains a 64-bit globally unique Device Identifier, the unique **APPEUI! (APPEUI!)**, DevNonce, whose value is zero when the device is first activated. Each time a re-request is sent, it increases by a random value to prevent replay attacks. The NS will analyze the received values to check whether the

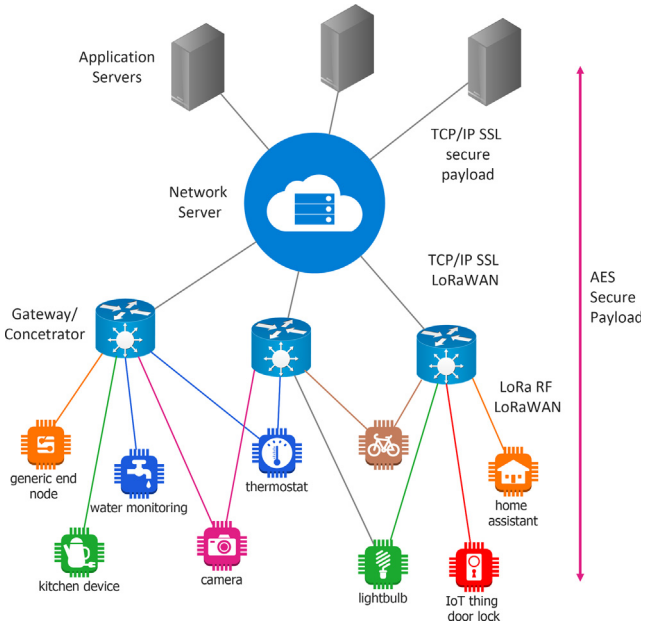

Fig. 1 Star-of-Stars: LoRaWAN Network Topology.

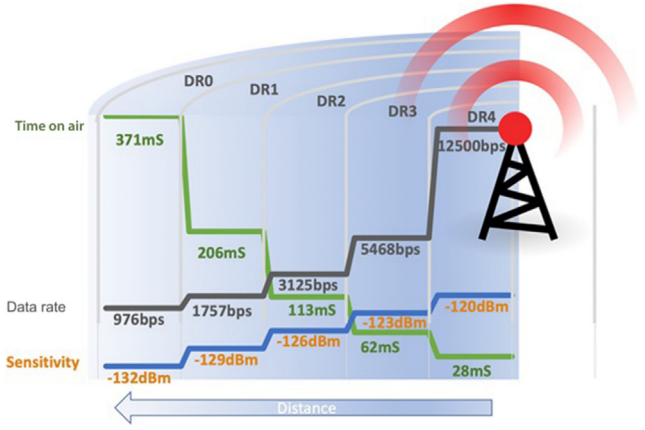

Fig. 2 Transmission speed as a function of range and transmission duration. Reprinted from [27]. Permission to reuse the figure was obtained via RightsLink (order number. 5235840330790).

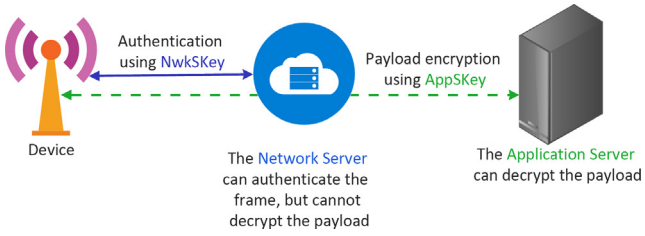

Fig. 3 Mutual authentication and end-to-end encryption.

obtained values were sent by the end node identified using DevEUI and AppEUI. If the keys are incorrect, the end node does not receive any response, and the process is aborted. If the keys are correct, the procedure accepts and delivers the *join accept* message which contains three values: AppNonce-a random number generated by NS, DevAddr-address of the end node assigned by NS and NetID-network identifier for separating different LoRaWAN networks.

- **ABP** - There is no join procedure in ABP activation mode. The end node does not contain DevEUI, AppEUI, and Application Key (AppKey) to be used in the join process, but it contains four session keys (FNwkSIntKey, SNwkSIntKey, NwkSEncKey, AppSKey) that are stored in the end node. The end node with session keys is ready to send data to the server the first time the end node is turned on.

After successfully activating nodes in either OTAA or ABP mode, the end node has the following information [29]:

- a end node address assigned to it by the network server - DevAddr,
- triplet network session keys - FNwkSIntKey, SNwkSIntKey and NwkSEncKey,
- application session key - AppSKey,
- network session frame counters (FCntUP, NFCntDwn) and application session counters (FCntUP, AFCntDwn).

OTAA is a safer way to activate a node and is recommended for more demanding applications. In this case, it is vital to securely store root keys and keep them from unauthorized access because if an attacker comes into their possession, they can create their device and generate session keys. On the other hand, the ABP activation method has the advantage of the ease of implementation, but at the expense of reduced security because endpoints use the same session keys throughout their lifetime. It is essential that these keys differ by the end node and that the frame counters are stored in permanent memory because, in this mode of node operation, the counter does not restart. The counter restart only happens by physically restarting the node. [29].

4. Experimental Setup

To implement a LoRaWAN network, three elements are required: gateway, at least one node, and a server receiving and performing data analysis. We used the following devices for practical implementation:

- *Raspberry Pi 3 + Gateway* module RHF0M301868 - 10-channel LoRaWAN gateway based on Semtech SX1301[1].
- Seeeduino LoRaWAN - development board focused on the Arduino with built-in LoRaWAN protocol. It is based on the RHF76-052AM radio module and
- RAK811 WisNode LoRa module - a development board that comes as an add-on (shield) to the Arduino development board. It has a built-in microprocessor and a radio module that can work without an Arduino board. The RAK811 module is integrated with Semtech's SX1276 and STM32.

In addition to these devices, servers (network, application, join server) are required, which make up one complete solution obtained from TheThingsNetwork[2] whose solutions were used in our experiments.

For all experiments performed in our work, a battery with a nominal voltage of 3.7 V and a capacity of 2200mAh was used, which was charged to a voltage level of 4.025 V. The reason for the charge level up to 4.025 is to prevent potential damage to the battery, and electronics from a portable battery for mobile phones (power bank) were used to manage the charge.

Fig. 4 shows a connection scheme of end devices with a battery and battery voltage measurement approach. The battery voltage level was measured using a single Seeeduino node, in such a way that 5 x 1 MΩ resistors were connected in series, giving a total resistance of 5 MΩ that did not affect battery consumption. The Seeeduino was connected to the third resistor in a row, analog input for reading.

Due to the accuracy of the measurement, the reading was performed every second in 5 min and then averaged and scaled to the voltage value. Scaling was performed with basic mathematical operations and numbers obtained by calibration.

Calibration was performed with a multimeter Fluke 113 whose accuracy is ±2% and the display resolution of the measured value is 0.001 V. On the Arduino development board, which was used as the basis for the WisNode shield, two LED SMD signal lamps were soldered to reduce power consumption by 7 mA. This gives a more realistic environment and a more realistic comparison.

5. Experimental Results and Discussion

Experimental analyzes included consideration of several practical tests. In all experiments, the gateway was set at the height of two meters, while nodes were set at the height of one meter in direct line-of-sight mode. The measurement log files are available at the Gdrive storage[3].

5.1. Experiment #1 - Energy Consumption vs Node-Gateway Distance

In the first experiment analyzing the energy efficiency of communication, devices are placed at distances of 5, 10, 15, and 20 meters and configured to communicate in OTAA mode with ADR. To measure energy consumption, the battery voltage values of the device were measured every 30 min. Each of these measurements was performed for 8 h (480 min) and with each device separately. The payload size is 4 bytes, and messages are sent every 5 s. Fig. 5 shows the obtained values from which it can be seen that for short distances, Seeeduino results with lower energy consumption. However, as the node's distance from the gateway increases, more energy is required to transmit the information, and the WisNode is more energy efficient (lower consumption).

[1] More details available at www.semtech.com/products/wireless-rf/lora-gateways/sx1301

[2] https://console.thethingsnetwork.org/

[3] https://drive.google.com/drive/folders/17fc0Ey4fxv6mJaHISbRj3WSwx_utwow3?usp = sharing

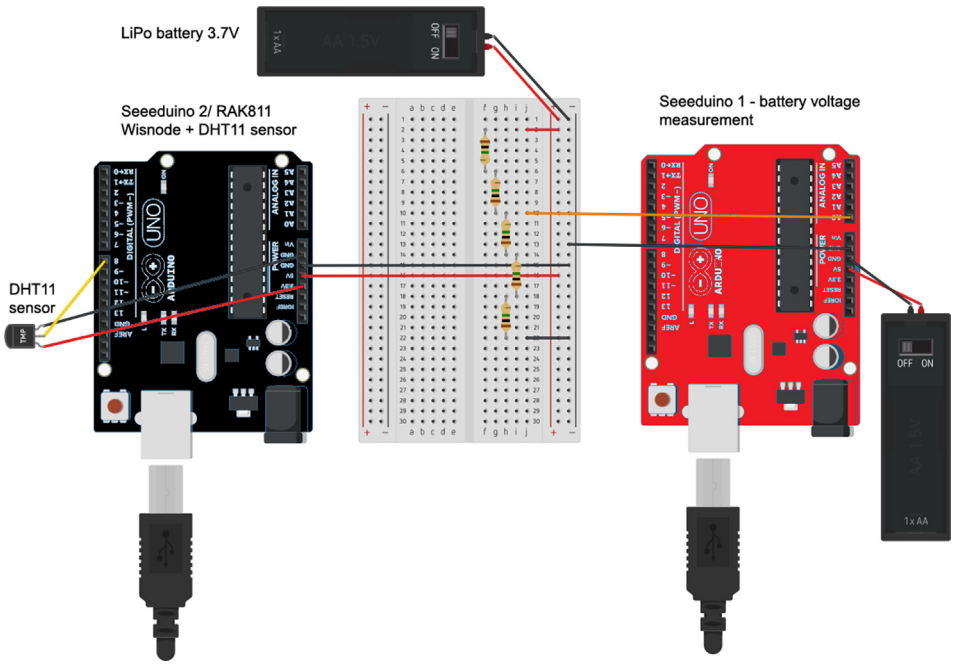

Fig. 4 The scheme of connection and battery measurement.

5.2. Experiment #2 - Energy Consumption using ABP/OTAA Security Settings

If nodes are set at a fixed distance and a fixed-size payload transfer is considered, the performance for different security settings can be compared. Fig. 6 shows the results obtained during the 8-h measurements for each of the devices separately, resulting in the 32-h experiment. The nodes are positioned at a distance of 10 meters, the payload size is 4 bytes, and messages are sent every 5 s. Communication with adaptive speeds in ABP and OTAA mode is considered.

Fig. 6 shows that communication in OTAA mode is more energy demanding because it is necessary to exchange join-request and activation messages to the gateway. There is no transmission of the specified messages in ABP mode since the device can communicate without prior activation. Due to the predictable number of messages, the energy consumption can also be predicted, which is evident from the linearity of the curves for the ABP mode of operation.

5.3. Experiment #3 - Energy Consumption for Different Payload Sizes

The third experiment involved considering energy efficiency with different payload sizes. In an experiment that lasted eight hours for each device, the transmission of messages with a minimum (1 byte) and a maximum (51 bytes) payload size was considered. Nodes are placed at a distance of 10 meters.

Table 1 lists the obtained results from which the influence of the amount of data on energy consumption can be noticed. The WisNode shows better results in both cases, which is explained by a faster and more sensitive ADR solution for changing the communication speed.

5.4. Experiment #4 - Energy Consumption for Different Spread Factor

An experiment in which devices transfer an 8-byte payload every 5 s with SF = 7 and SF = 12 was considered. This experiment is closely related to the previous one because both differ in message transmission duration. Thus, for SF = 7, the transmission time is 30,976 ms, and for SF = 12, 827,392 ms, almost 27 times more time. The nodes are positioned at a distance of 10 meters.

This difference is also notable in battery consumption, as shown in Fig. 7. The obtained results show that a lower SF (higher transfer rates) improves energy savings. It is known that smaller SF is used for nodes closer to the gateway, so it makes sense to use less battery, while remote nodes require more energy.

5.5. Experiment #5 - Security Settings Analysis

IoT end devices are most often installed in fixed positions. Given the remote exposure from the rest of the network, there is always the threat of a physical takeover of the device through which an attacker could read security settings and carry out attacks on other network nodes [7,9,29].

Seeeduino node is an end node made on the Arduino principle and is independent, i.e., no additional devices other than sensors or actuators are required for its operation. Code compiled on an Arduino chip is challenging to extract in a readable form and from which sensitive data can be obtained. However, the WisNode node comes as an add-on to the Arduino. It has a LoRa chip and a memory in which it stores data. It is enough to connect the device via a USB cable to a computer and connect to WisNode via serial communication to access the

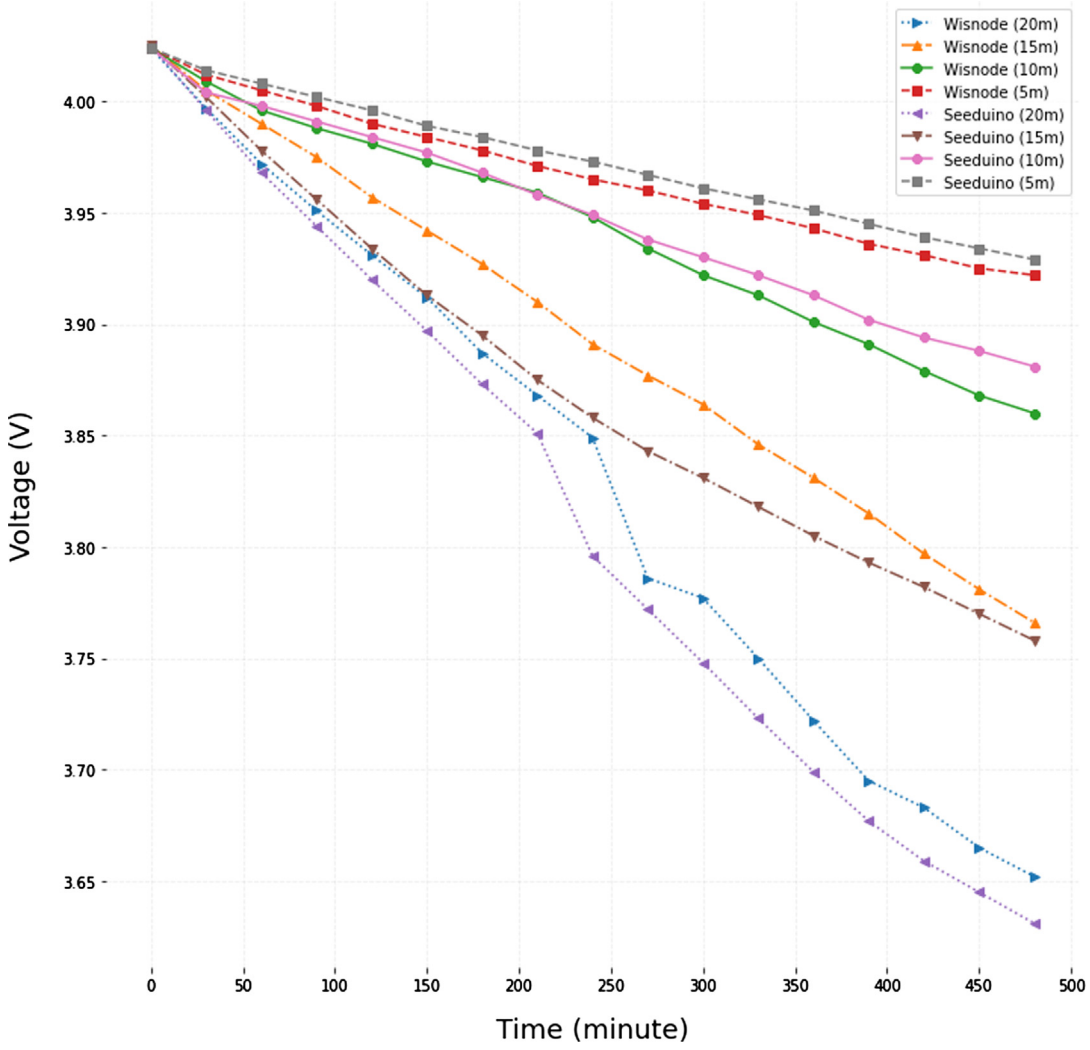

Fig. 5 Energy Consumption vs Node-Gateway Distance. The payload size is 4 bytes. Messages are sent every 5 s in OTAA mode with ADR enabled.

memory. WisNode implements a set of commands, which are used to set the configuration but also to read it.

Figs. 8 and 9 show and label the keys for the OTAA and ABP access modes, respectively. As can be seen, only one command ($at + get\_config = lora : status$) is enough to get all the data to access the network. An attacker can use these keys and transfer them to his device, which is then presented to the network as the original device and delivers false data to the server.

As noted in [6,11] the additional security problem can arise if the attacker floods the network with fake packets. These packets may be of the same frequency and propagation factor as those that are part of that network. Interference occurs so that none of the packets will be received on the gateway and thus not forwarded to the server.

We performed this experiment so that one node (seeeduino) sends with an SF = 12 and at a fixed frequency of 868.5 MHz. We targeted SF = 12 because it achieves the longest signal emission and is the easiest to cause interference. The second node (WisNode), which served as a jammer, also sent 4-byte data at 868.5 MHz. The application on node one is programmed to send 17-byte packets every 10 s. After five packets are sent, the application on node two also generates 17-byte packets every 10 s, but it starts sending 30 ms before the expected time of sending packets from the first node. So, the application on node two sends its packets 30 ms before the first node does.

Fig. 10 shows a situation where data from the first node arrives until the moment when the second node starts sending packets on the same frequency and the same SF. It can be seen that the gateway does not receive data from the first node during the attack but processes the packet from the second node since these packages were first received. Interference can be avoided even at the same frequencies, but in the case when different SF settings are used.

5.6. Experiment #6 - Battery Drain Testing

This experiment aimed to show below which voltage level one LoRaWAN node cannot operate and how long it took to be active to drain the battery to the specified level. The obtained results showed that the node ceases to be active after the bat-

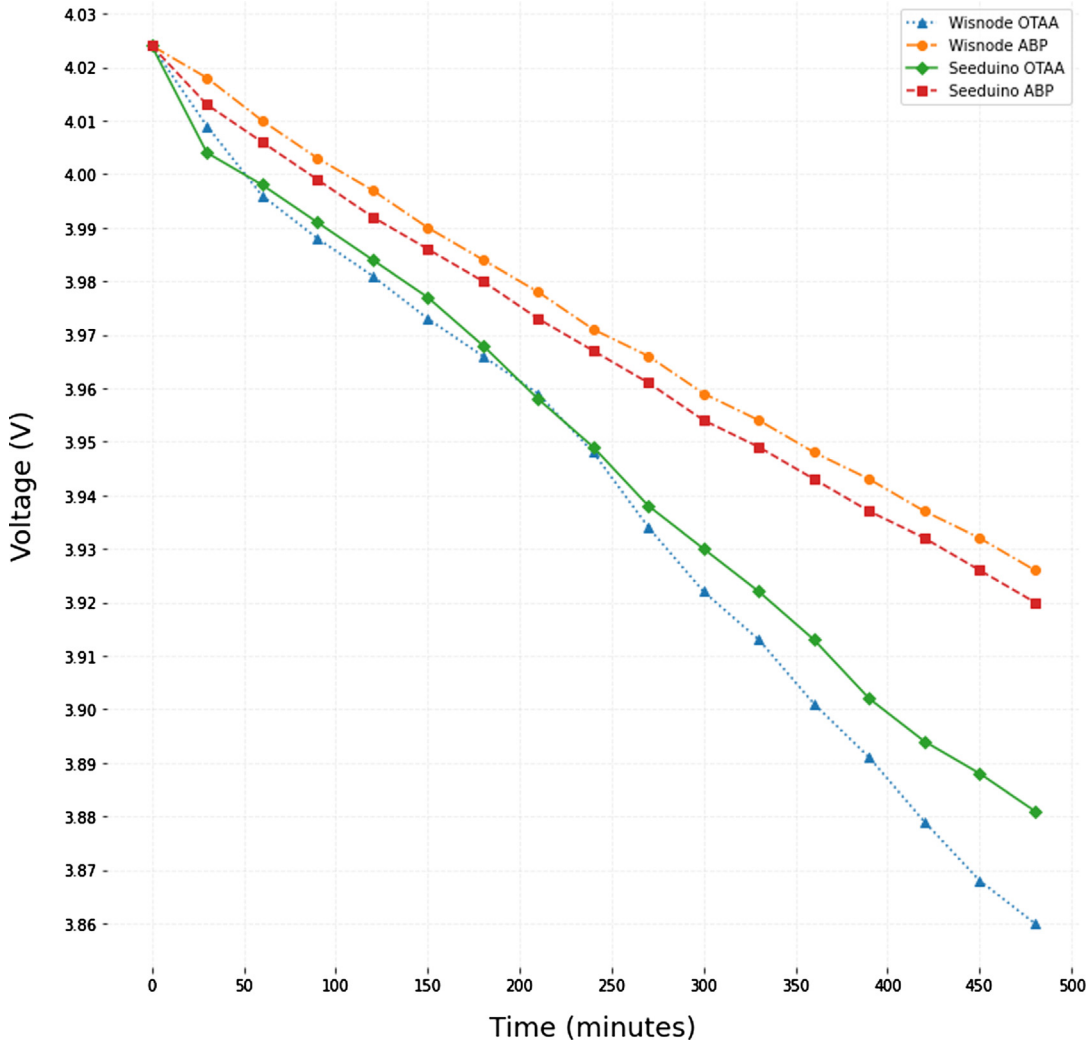

Fig. 6 Energy Consumption using ABP/OTAA Security Settings. The payload size is 4 bytes. Messages are sent every 5 s. The nodes are positioned at a distance of 10 meters.

tery voltage level drops below 3 V (approximate value, due to measurement device limitation). For this experiment, the gateway was set at the height of 2 m, while the seeeduino node was set at the height of 1 m in free-line-of sight ABP mode. The node worked in stress mode, i.e., sent 4-byte packets every 5 s. It took three days and 10 h of continuous operation to lower the voltage from 4,025 V to 3 V, where almost 60,000 packets were sent from the node to the gateway.

The battery capacity is 2200 mAh, and the average consumption of Seeeduino nodes when sending every 5s 27 mA. Based on the theoretical approach(Eq. 1), it is obtained that the node can work for 81.5 h, which is approximately 82 h obtained by the obtained experimental results.

$$time = \frac{battery\_capacity}{average\_consumption} \tag{1}$$

As shown in Fig. 12-bottom, the average consumption of WisNode is 18 mA in active state (measured in the AT mode). Thus, when supplied with the same battery of 2200 mAh, it can operate for 122.2 h.

5.7. *Experiment #7 - Sleep Time Testing*

The Arduino has the addition of the *doSleep* function allowing the node to switch to hibernate/sleep. The procedure requires the implementation of a software library[4] that is only supported for seeeduino devices. As shown in Fig. 11 the current consumption in sleep mode is 0A (bottom image), while in the time interval of 6s, the current consumption is 20 mA, with a short-term peak up to 58 mA. The current change at the time the device *wakes up* and sends data corresponds to [3], where the authors recorded the current change with the analyzer.

Based on official data from the Seeeduino website[5] the manufacturer claims that the minimum power consumption of the used Seeeduino node is 80 μA, and based on the equation from the previously described experiment, it can be calculated that for a 2200 mAh battery the node can send data 4 times a day for three years.

[4] More details available at the https://tum-gis-sensor-nodes.readthedocs.io/en/latest/seeeduino_lorawan/

[5] https://wiki.seeedstudio.com/Seeeduino_LoRAWAN/

Table 1 Energy Consumption for Different Payload Sizes vs Node-Gateway Distance. Messages are sent every 5 s.

	Wisnode Energy Consumption (V)		Seeeduino Energy Consumption (V)	
Distance(m)	Payload 1 byte	Payload 50 bytes	Payload 1 byte	Payload 50 bytes
0	4.025	4.024	4.025	4.024
30	4.017	4.017	4.017	4.018
60	4.012	4.01	4.011	4.012
90	4.007	3.999	4.004	4.004
120	4.002	3.991	3.998	3.997
150	3.997	3.983	3.992	3.99
180	3.991	3.978	3.986	3.985
210	3.986	3.974	3.979	3.974
240	3.981	3.967	3.974	3.968
270	3.977	3.96	3.967	3.957
300	3.971	3.953	3.962	3.952
330	3.966	3.947	3.956	3.945
360	3.961	3.941	3.951	3.939
390	3.955	3.935	3.946	3.932
420	3.951	3.929	3.942	3.923
450	3.946	3.924	3.937	3.918
480	3.942	3.917	3.933	3.913

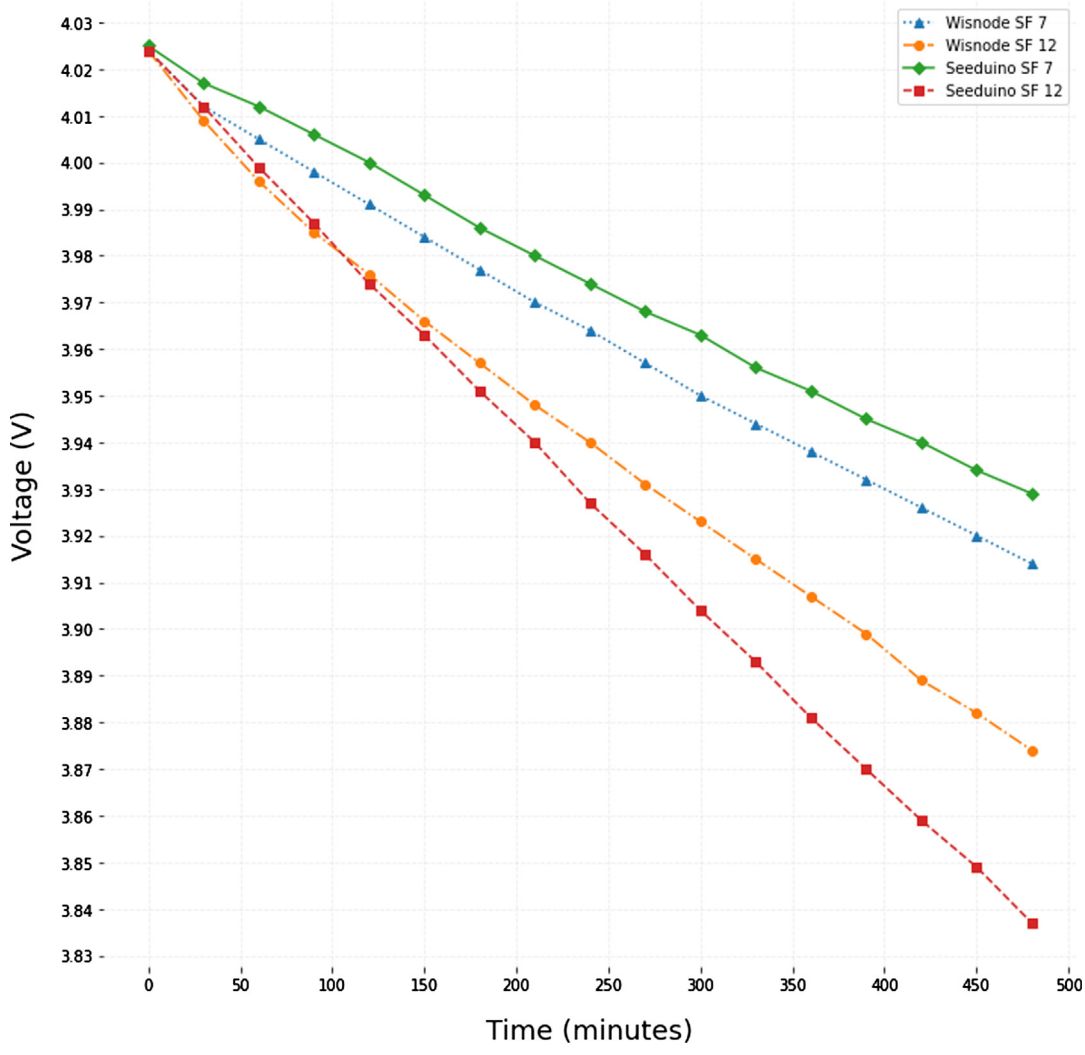

Fig. 7 Energy Consumption for Different Spread Factor vs Node-Gateway Distance. The payload size is 8 bytes. Messages are sent every 5 s.

```
at+get_config=lora:status
OK.
************************************************
==============LoRaWAN Status List================
Work Mode: LoRaWAN
Region: EU868
Send_interval: 600s
Auto send status: false.
Join mode: OTAA
DevEui: 7E2B445E97C2CF45
AppEui: 70B3D57ED00335FC
AppKey: 1B3FC20A26200192047B07071F317D3F
Class: A
Joined Network:true
IsConfirm: true
AdrEnable: false
EnableRepeaterSupport: false
RX2_CHANNEL_FREQUENCY: 869525000, RX2_CHANNEL_DR:3
RX_WINDOW_DURATION: 3000ms
RECEIVE_DELAY_1: 1000ms
RECEIVE_DELAY_2: 2000ms
JOIN_ACCEPT_DELAY_1: 5000ms
JOIN_ACCEPT_DELAY_2: 6000ms
Current Datarate: 0
Primeval Datarate: 0
ChannelsTxPower: 0
UpLinkCounter: 0
DownLinkCounter: 0
==================List End=====================
************************************************
```

Fig. 8 WisNode security data for OTAA mode.

```
at+set_config=lora:join_mode:1
LoRa configure ABP success
OK
at+get_config=lora:status
OK.
************************************************
==============LoRaWAN Status List================
Work Mode: LoRaWAN
Region: EU868
Send_interval: 600s
Auto send status: false.
Join_mode: ABP
DevAddr: 26013FAF
AppsKey: DA52D2AA7A9421EBA73FDC0FD7E85BF9
NwksKey: 51CDE7144E2967D7C49DE8FA79393992
Class: A
Joined Network:false
IsConfirm: true
AdrEnable: false
EnableRepeaterSupport: false
RX2_CHANNEL_FREQUENCY: 869525000, RX2_CHANNEL_DR:3
RX_WINDOW_DURATION: 3000ms
RECEIVE_DELAY_1: 1000ms
RECEIVE_DELAY_2: 2000ms
JOIN_ACCEPT_DELAY_1: 5000ms
JOIN_ACCEPT_DELAY_2: 6000ms
Current Datarate: 0
Primeval Datarate: 0
ChannelsTxPower: 0
UpLinkCounter: 0
DownLinkCounter: 0
==================List End=====================
************************************************
```

Fig. 9 WisNode security data for ABP mode.

Since WisNode consumes 11 mA in sleeping mode, 65 mA in message receiving downlink mode, and 18 mA in uplink mode (Fig. 12-top), it is operable for approximately 8 days. It is important to note here that the measurement of the WisNode module was performed directly without an Arduino board that would additionally consume battery resources. The firmware version of used WisNode was 3.0.0.13-H.

6. Discussion

Table 2 lists all experiments performed and the indication of better results obtained. In experiment #1, ADR was used, and WisNode achieved better results on longer end node distances because it possesses better speed change and propagation factor techniques compared to the seeeduino node. Also, it performed better when less secure ABP communication is analyzed in experiments #2 and #3 considering different payload sizes.

In experiment #4, we analyzed communication with a different propagation factor for minimum SF = 7 and maximum SF = 12 values. The difference in signal emission time was 27 times less for SF7, which was also felt on battery consumption. With a smaller spectrum spreading factor, battery consumption is reduced. Also, the lower spectrum spreading factor is associated with the distance from the gateway in that closer devices use the lower spectrum spreading factor while more distant devices use a higher spectrum spreading factor. For communication with lower SF values, it is noticed that WisNode consumes less energy. But, with the increase of SF values, Seeeduino is more energy efficient.

Regarding the safety aspect, it has been shown in experiment #2 that different ways of activating OTAA (safer) and ABP (less secure) end devices consume the battery differently as the OTAA activation method sends additional activation request messages when the device is activated. The exchange of keys during communication was expected to consume more energy, which was confirmed by the experiment in this paper.

As shown in experiment #5, the biggest security threat is physical access to the device. It is evident in the case of WisNode due to the possibility of reading the keys used to activate the node that an attacker can use for a new device. With the Seeeduino device, it is challenging to read data from memory which means it is less exposed. It is important to note that although LoRaWAN networks use network and application layer encryption, there is still the possibility of security vulnerabilities due to the media they use for communication.

As part of experiment #6, we analyzed the operating time of end nodes when it comes to energy efficiency. Based on the collected values, we have shown that it has better performance because it consumes 18 mA in the active state compared to Seeduino, which consumes 27 mA. Also, to maintain the system's longevity, the end nodes should consume minimal energy when in sleep mode, which is challenging for most manufacturers. As analyzed in experiment #7, the seeeduino can consume only 80 μA when the device is idle when no data is required. In such conditions, with a battery capacity of 2200 mAh, it is possible to send messages four times a day for three years. On the other hand, WisNode consumes 11 mA in sleeping mode, and if powered by the same battery, it can stay in the sleeping mode for just over 8 days.

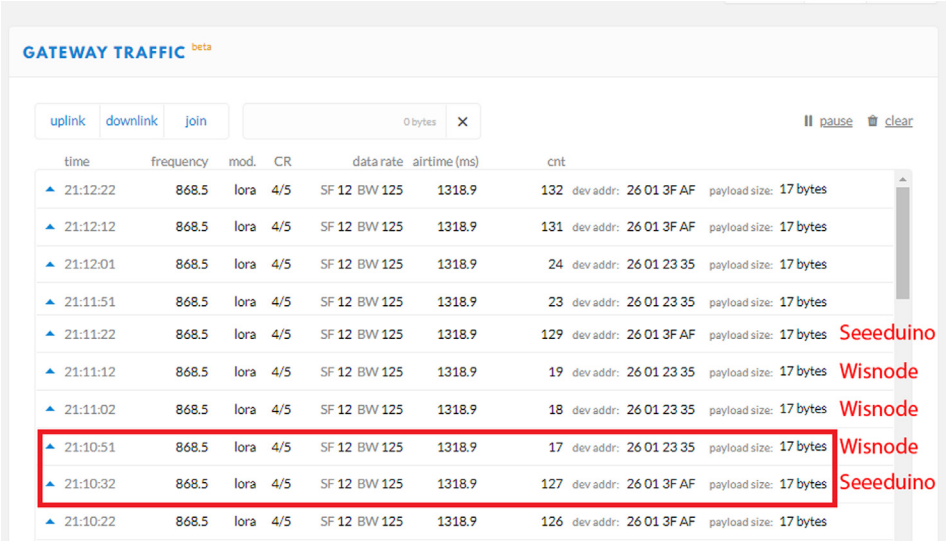

Fig. 10 Influence of interference on traffic reception. View received messages via the gateway. Traffic from the WisNode device interferes with traffic from the seeeduino device.

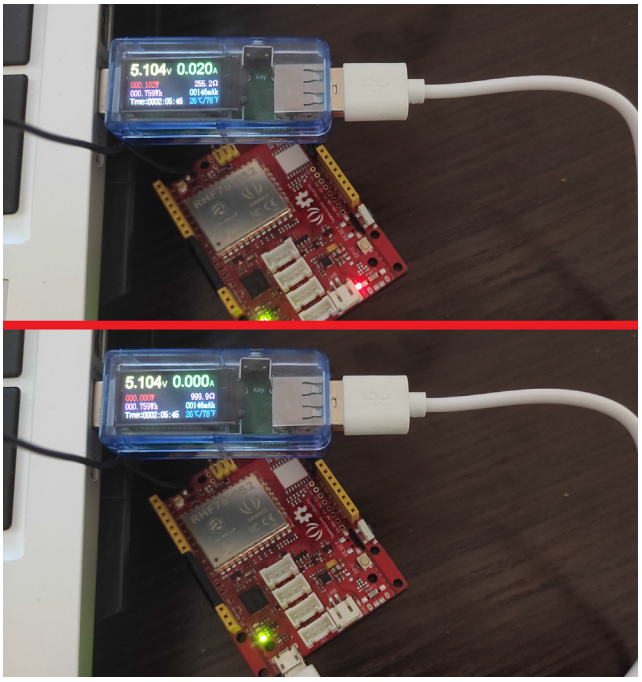

Fig. 11 Seeeduino energy consumption measurements. Top: sending condition; bottom: sleeping state.

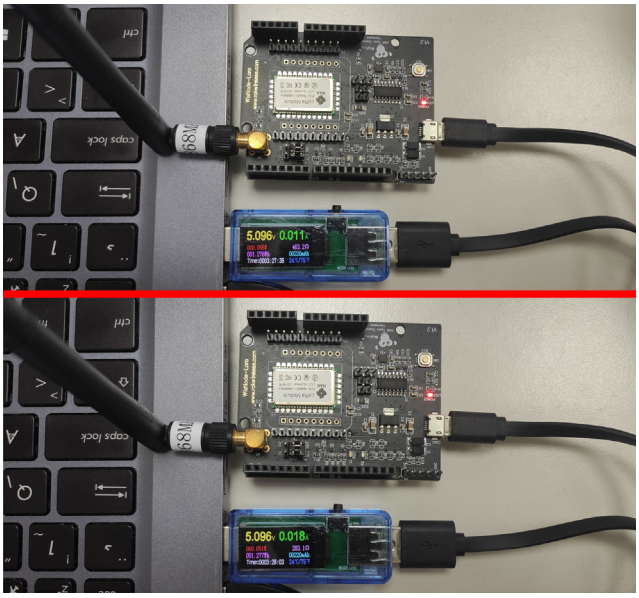

Fig. 12 WisNode energy consumption measurements. Top: sleeping state; bottom: sending condition.

7. Conclusion

This paper describes practical experiments that consider the energy efficiency and safety of LoRaWAN end devices (Seeeduino LoRaWAN node and RAK811 WisNode node).

After performing experiments related to energy efficiency, it can be concluded that the node's distance from the gateway significantly affects battery consumption, which was expected because the transmission of messages requires higher transmission power and longer transmission time. Experiments with different payload sizes and spectrum spreading factors showed that longer transmission time consumes more battery power.

When considering the security aspect, it is important to point out that ABP mode can be viewed as a more energy-efficient approach but at the expense of security. Namely, the ABP mode exchanges fewer messages due to the lack of join procedure. Due to the inability to refresh session keys (lack of no re-keying option) [29], ABP also poses an additional security challenge as it relies on counter values stored in memory. If there is a problem with keeping or reading these values, the end device will enter the out-of-sync state and become unusable. WisNode devices are more vulnerable due

Table 2 Summary table of experiments and comparisons of behavior of tested End Node devices. The X value in the table indicates better performance.

Experiment	WisNode	Seeeduino
Experiment #1 - Energy Consumption with OTAA Payload Size 4 bytes; End Node Distance < 10 m		x
Experiment #1 - Energy Consumption with OTAA Payload Size 4 bytes; End Node Distance > 10 m	x	
Experiment #2 - Efficiency Consumption with OTAA Payload Size 4 bytes; End Node Distance = 10 m		x
Experiment #2 - Efficiency Consumption with ABP Payload Size 4 bytes; End Node Distance = 10 m	x	
Experiment #3 - Efficiency Consumption with ABP Payload Size 1 byte; End Node Distance = 10 m	x	
Experiment #3 - Efficiency Consumption with ABP Payload Size 51 byte; End Node Distance = 10 m	x	
Experiment #4 - Energy Consumption for Spread Factor SF = 7 Payload Size 8 bytes; End Node Distance = 10 m; ABP mode	x	
Experiment #4 - Energy Consumption for Spread Factor SF = 12 Payload Size 8 bytes; End Node Distance = 10 m; ABP mode		x
Experiment #5 - Security Keys Extraction from Memory		x
Experiment #6 - Battery Drain Testing	x	
Experiment #7 - Work in Sleep/Idle Time		x

to their exposure to physical memory attacks from this security aspect.

An alternative is to use an OTAA solution that establishes secure session keys to protect communication. However, in that case, more energy resources are consumed, so this communication is recommended for those applications that require a higher degree of security.

The main contribution of this paper is reflected in the practical testing of LoraWaN end devices to consider energy and safety limitations.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This research received funding from the Czech Ministry of Education, Youth and Sports within the grant reg. No. SP2022/5 conducted by VSB-Technical University of Ostrava.

References

[1] H.H.R. Sherazi, L.A. Grieco, L.A. Grieco, G. Boggia, Energy-efficient LoRaWAN for Industry 4.0 Applications (2020) 11.

[2] C. Delgado, J.M. Sanz, C. Blondia, J. Famaey, Batteryless LoRaWAN Communications Using Energy Harvesting: Modeling and Characterization, IEEE Internet Things J. 8 (4) (2021) 2694–2711, https://doi.org/10.1109/JIOT.2020.3019140, URL: https://ieeexplore.ieee.org/document/9174941/.

[3] L. Casals, B. Mir, R. Vidal, C. Gomez, Modeling the Energy Performance of LoRaWAN, Sensors 17 (10) (2017) 2364, https://doi.org/10.3390/s17102364, URL: http://www.mdpi.com/1424-8220/17/10/2364.

[4] J. Toussaint, N. El Rachkidy, A. Guitton, Performance analysis of the on-the-air activation in LoRaWAN, in: 2016 IEEE 7th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), IEEE, 2016, pp. 1–7. doi:10.1109/IEMCON.2016.7746082. URL: http://ieeexplore.ieee.org/document/7746082/

[5] S. Mathur, A. Sankar, P. Prasan, B. Iannucci, Energy Analysis of LoRaWAN Technology for Traffic Sensing Applications, in: Intelligent Transportation Systems (ITS), World Congress, 2017.

[6] M.N. Nafees, N. Saxena, P. Burnap, B.J. Choi, Impact of Energy Consumption Attacks on LoRaWAN-Enabled Devices in Industrial Context (2020) 4.

[7] X. Yang, E. Karampatzakis, C. Doerr, F. Kuipers, Security Vulnerabilities in LoRaWAN, in: 2018 IEEE/ACM Third International Conference on Internet-of-Things Design and Implementation (IoTDI), IEEE, 2018, pp. 129–140. doi:10.1109/IoTDI.2018.00022. URL: https://ieeexplore.ieee.org/document/8366983/.

[8] E. Aras, G.S. Ramachandran, P. Lawrence, D. Hughes, Exploring the Security Vulnerabilities of LoRa, in: 2017 3rd IEEE International Conference on Cybernetics (CYBCONF), IEEE, 2017, pp. 1–6. doi:10.1109/CYBConf.2017.7985777. URL: http://ieeexplore.ieee.org/document/7985777/

[9] B. Oniga, V. Dadarlat, E. De Poorter, A. Munteanu, Analysis, design and implementation of secure LoRaWAN sensor networks, in: 2017 13th IEEE International Conference on Intelligent Computer Communication and Processing (ICCP), IEEE, 2017, pp. 421–428. doi:10.1109/ICCP.2017.8117042. URL: http://ieeexplore.ieee.org/document/8117042/

[10] S. Tomasin, S. Zulian, L. Vangelista, Security Analysis of LoRaWAN Join Procedure for Internet of Things Networks, in: 2017 IEEE Wireless Communications and Networking Conference Workshops (WCNCW), IEEE, 2017, pp. 1–6. doi:10.1109/WCNCW.2017.7919091. URL: http://ieeexplore.ieee.org/document/7919091/

[11] W.-J. Sung, H.-G. Ahn, J.-B. Kim, S.-G. Choi, Protecting end-device from replay attack on LoRaWAN, in: 2018 20th International Conference on Advanced Communication Technology (ICACT), IEEE, 2018, pp. 167–171. doi:10.23919/ICACT.2018.8323684. URL: https://ieeexplore.ieee.org/document/8323684/

[12] L. Casals, B. Mir, R. Vidal, C. Gomez, Modeling the energy performance of LoRaWAN, Sensors (Switzerland) 17 (10). doi:10.3390/s17102364.

[13] J. Petäjäjärvi, K. Mikhaylov, R. Yasmin, M. Hämäläinen, J. Iinatti, Evaluation of LoRa LPWAN Technology for Indoor Remote Health and Wellbeing Monitoring, Int. J. Wireless Inf. Networks 24 (2) (2017) 153–165, https://doi.org/10.1007/s10776-017-0341-8.

[14] B. Kim, K.I. Hwang, Cooperative downlink listening for low-power long-range wide-area network, Sustainability (Switzerland) 9 (4). doi:10.3390/su9040627.

[15] T. Bouguera, J.F. Diouris, J.J. Chaillout, R. Jaouadi, G. Andrieux, Energy consumption model for sensor nodes based

on LoRa and LoRaWAN, Sensors (Switzerland) 18 (7) (2018) 1–23, https://doi.org/10.3390/s18072104.
[16] H.H.R. Sherazi, L.A. Grieco, M.A. Imran, G. Boggia, Energy-Efficient LoRaWAN for Industry 4.0 Applications, IEEE Trans. Industr. Inf. 17 (2) (2021) 891–902, https://doi.org/10.1109/TII.2020.2984549.
[17] M.A. Khan, M.M. Nasralla, M.M. Umar, Z. Iqbal, G.U. Rehman, M.S. Sarfraz, N. Choudhury, A Survey on the Noncooperative Environment in Smart Nodes-Based Ad Hoc Networks: Motivations and Solutions, Security and Communication Networks (2021), https://doi.org/10.1155/2021/9921826.
[18] H. Noura, T. Hatoum, O. Salman, J.-P. Yaacoub, A. Chehab, LoRaWAN security survey: Issues, threats and possible mitigation techniques, Internet of Things 12 (2020) 100303, https://doi.org/10.1016/j.iot.2020.100303, URL: https://linkinghub.elsevier.com/retrieve/pii/S2542660520301359.
[19] M. Eldefrawy, I. Butun, N. Pereira, M. Gidlund, Formal security analysis of LoRaWAN, Comput. Netw. 148 (2019) 328–339, https://doi.org/10.1016/j.comnet.2018.11.017.
[20] X. Yang, E. Karampatzakis, C. Doerr, F. Kuipers, Security Vulnerabilities in LoRaWAN, in: 2018 IEEE/ACM Third International Conference on Internet-of-Things Design and Implementation (IoTDI), IEEE, 2018, pp. 129–140. doi:10.1109/IoTDI.2018.00022. URL: https://ieeexplore.ieee.org/document/8366983/.
[21] I. Butun, N. Pereira, M. Gidlund, Security risk analysis of LoRaWAN and future directions, Future Internet 11 (1) (2018) 1–22, https://doi.org/10.3390/fi11010003.
[22] C. Pham, A. Bounceur, L. Clavier, U. Noreen, M. Ehsan, Radio channel access challenges in LoRa low-power wide-area networks (2020) 38. URL: doi: 10.1016/B978-0-12-818880-4.00004-1.
[23] M. Capuzzo, D. Magrin, A. Zanella, Confirmed Traffic in LoRaWAN: Pitfalls and Countermeasures To cite this version: HAL Id: hal-01832534 Confirmed Traffic in LoRaWAN: Pitfalls and Countermeasures, 2018 17th Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net) (2018) 1–7.
[24] P. Seneviratne, Beginning LoRa Radio Networks with Arduino: Build Long Range, Low Power Wireless IoT Networks, Technology in Action, Apress, 2019.
[25] A. Hazarika, S. Poddar, M.M. Nasralla, H. Rahaman, Area and energy efficient shift and accumulator unit for object detection in IoT applications, Alexandria Engineering Journal 61 (1) (2022) 795–809, https://doi.org/10.1016/j.aej.2021.04.099.
[26] L. Alliance, White paper: A technical overview of LoRa and LoRaWAN (2015).
[27] A. Yegin, T. Kramp, P. Dufour, LoRaWAN protocol: specifications, security, and capabilities (2020) 27. URL: doi: 10.1016/B978-0-12-818880-4.00003-X.
[28] Sornin, Nicolas, Luis, Miguel, Eirich, Thomas, Kramp, Thorsten, O. Hersent, LoRaWAN Specification, Tech. rep., LoRa alliance. (2015). URL: https://osch.oss-cn-shanghai.aliyuncs.com/blogContentFileSnapshot/1556464676588.pdf
[29] Butun, Ismail, Pereira, Nuno, Gidlund, Mikael, Security risk analysis of lorawan and future directions, Future Internet 11 (1). doi:10.3390/fi11010003. URL: https://www.mdpi.com/1999-5903/11/1/3.
[30] H. Noura, T. Hatoum, O. Salman, J.-P. Yaacoub, A. Chehab, LoRaWAN security survey: Issues, threats and possible mitigation techniques, Internet of Things 12 (2020) 100303, https://doi.org/10.1016/j.iot.2020.100303, URL: https://linkinghub.elsevier.com/retrieve/pii/S2542660520301359.
[31] Phui San Cheong, Johan Bergs, Chris Hawinkel, Jeroen Famaey, Comparison of LoRaWAN classes and their power consumption, 2017 IEEE Symposium on Communications and Vehicular Technology, SCVT 2017 12 (2017) 1–6, https://doi.org/10.1109/SCVT.2017.8240313.

List of Acronyms

IoT: Internet of Things.
LoRaWAN: Long Range Wide Area Network.
LoRa: Long Range.
SF: Spreading Factor.
OTAA: Over-The-Air Activation.
GW: Gateway
DoS: Denial of Service.
DR: Dynamic Range.
VPN: Virtual Private Network.
UART: Universal Asynchronous Receiver Transmitter.
ADR: Adaptive Data Rate.
SPI: Serial Peripheral Interface.
RSSI: Received Signal Strength Indicator.
ABP: Activation By Personalization.
CR: Coding Rate.
BW: Bandwidth
FEC: Forward Error Correction.
ToA: Time on Air.
LPWANs: Low Power Wide Area Networks.
FSK: Frequency Shift Keying.
NS: Network Server.
AppEUI: Application Identifier.
DevEUI: Device Identifier.
AppKey: Application Key.