network

MDPI

Article

Energy-Aware Security Adaptation for Low-Power IoT Applications

Miguel Rosendo † and **Jorge Granjal** *,†

Centre for Informatics and Systems, University of Coimbra, 3030-290 Coimbra, Portugal; mrosendo@student.dei.uc.pt

* Correspondence: jgranjal@dei.uc.pt

† These authors contributed equally to this work.

Abstract: The constant evolution in communication infrastructures will enable new Internet of Things (IoT) applications, particularly in areas that, up to today, have been mostly enabled by closed or proprietary technologies. Such applications will be enabled by a myriad of wireless communication technologies designed for all types of IoT devices, among which are the Long-Range Wide-Area Network (LoRaWAN) or other Low-power and Wide-Area Networks (LPWAN) communication technologies. This applies to many critical environments, such as industrial control and healthcare, where wireless communications are yet to be broadly adopted. Two fundamental requirements to effectively support upcoming critical IoT applications are those of energy management and security. We may note that those are, in fact, contradictory goals. On the one hand, many IoT devices depend on the usage of batteries while, on the other hand, adequate security mechanisms need to be in place to protect devices and communications from threats against their stability and security. With thismotivation in mind, we propose a solution to address the management, in tandem, of security and energy in LoRaWAN IoT communication environments. We propose and evaluate an architecture in the context of which adaptation logic is used to manage security and energy dynamically, with the goal of guaranteeing appropriate security, while promoting the lifetime of constrained sensing devices. The proposed solution was implemented and experimentally evaluated and was observed to successfully manage security and energy. Security and energy are managed in line with the requirements of the application at hand, the characteristics of the constrained sensing devices employed and the detection, as well as the threat, of particular types of attacks.

Keywords: IoT; LoRaWAN; LPWAN; end-to-end security; energy and security; dynamic security adaptation

check for updates

Citation: Rosendo, M.; Granjal, J. Energy-Aware Security Adaptation for Low-Power IoT Applications. *Network* **2022**, 2, 36–52. https://doi.org/10.3390/network2010003

Academic Editor: Youn-Hee Han

Received: 4 September 2021
Accepted: 4 January 2022
Published: 14 January 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

The ongoing evolution of the global communications infrastructure promises to provide support for pervasive wireless communications, which are employed by a myriad of applications, among which are mobile applications employing all kinds of sensing and actuation devices. In addition, diverse communication technologies will contribute to enabling new areas for the application of the IoT paradigm. Among such areas, we will encounter, for sure, the need to support strong security guarantees, while at the same time, guaranteeing appropriate lifetimes for the constrained and battery-powered devices employed. In this context, we may consider that security and energy will require a joint management, in order for many IoT applications to be viable. One class of IoT communication environments fitting this scenario is that of the Low-Power and Wide-Area Networks (LPWAN) [1], and in this context, the challenge is to address the support that these protocol classes offer to industrial applications with critical security and energy management requirements [2]. In this context, the Long-Range Wide-Area Network (LoRaWAN) open standard is a popular and effective LPWAN protocol. Figure 1 illustrates the general LoRaWAN communications architecture [3].

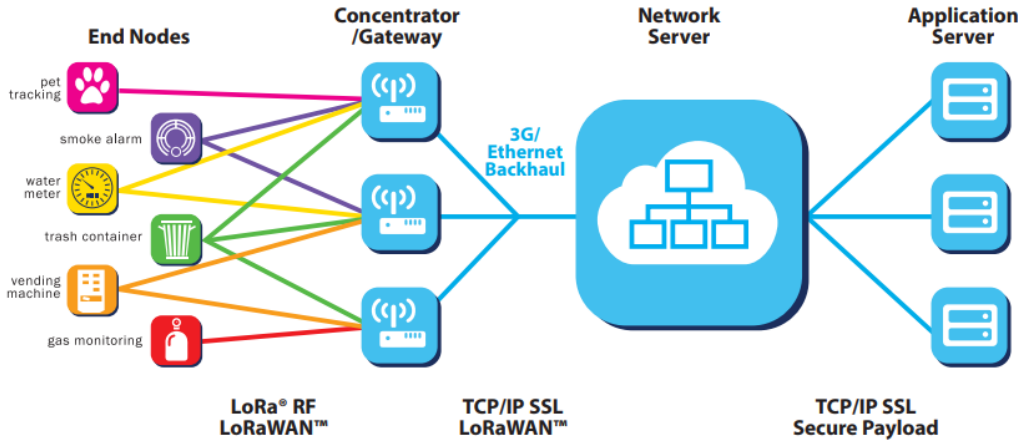

Figure 1. LoRaWAN architecture overview [3].

From the point of view of security, the employment of wireless communications between LoRaWAN sensing devices and gateways is particularly critical [4]. End-to-end security in this context is thus critical, in order to secure end-to-end communications against external threats. Our proposal in this article is in line with this concern, as we discuss later. On the other hand, critical applications call for the usage of appropriate energy management mechanisms, since many LoRaWAN sensing platforms are powered by batteries, and thus, a careful control of how much energy is expended to support security mechanisms is important, so as not to compromise the effective lifetime of the device in the context of the application at hand.

The LoRaWAN security scheme is composed of two levels of protection: at the network and application layers. Each uses one AES symmetric encryption key, which is obtained from the same primary key. The primary key is preconfigured and shared with each node by the network server, which is responsible for its generation during the setup phase. The network key is used to encrypt and to guarantee integrity for communications in the LoRaWAN infrastructure, while the application key ensures the confidentiality of the payload in the context of the application.

Devices in a LoRaWAN infrastructure may belong to three categories, namely A, B and C. For type A sensors, the device is responsible for initiating communications with the server, using dedicated uplink and downlink communication windows. This behavior allows the device to hibernate and to reduce energy consumption, and in consequence, this is the most efficient class regarding energy usage. Class B is similar to class A; however, in this case, the devices are synchronized with the server to schedule downlink communication windows, making it more costly in terms of battery usage. In Class C, a permanent open channel for bidirectional communication is maintained with the server, making it possible to receive a message at any time. We can thus observe that this configuration makes Class C devices the most expensive in terms of energy consumption.

Our focus in this article lies in the proposal of an architecture and algorithms to support the adaptation logic required for the dynamic management of security and energy in tandem, in the context of radio-frequency communications in an LPWAN network, as per the requirements and motivations previously discussed. In particular, the goal is to protect the application against external threats, with energy being dynamically managed to promote acceptable lifetime for the constrained devices employed.

The article proceeds as follows: We discuss related work in Section 2, and in Section 3, we present the proposed architecture for dynamic energy and security management in IoT. In Section 4, we discuss the proposed algorithm for dynamic security and energy management, a particularly relevant component of the proposed solution. Section 5 focuses on the experimental and analytic evaluation of the proposed architecture and controller, and finally, Section 6 concludes the article and discusses future work and research opportunities.

2. Related Work

As previously discussed, LoRaWAN is one particularly important LPWA communications technology and lays the ground for the implementation and evaluation of the architecture and controller proposed in this article. In regards to the management of security and energy, we note that LoRaWAN lacks the mechanisms to support its dynamic management and adaptation, given that three classes of end devices may be employed with fixed configurations in terms of energy [5]. In addition, the standard does not consider the possibility of adapting the level of security dynamically, and as such, the impact of security mechanisms on energy is predictable and unavoidable, irrespective of the devices' classes employed. We thus may observe the lack of dynamic adaptation strategies for security and energy, particularly strategies that may promote energy saving in the presence of attacks, even when employing class B devices, and that can also support extra protection by enabling security for end-to-end communications in the presence of relevant threats. Given the previous limitations, we are able to identify a number of research proposals over the years, with the goal of improving the security of LoRaWAN-based applications, as we proceed to discuss.

The architecture in [6] considers the usage of a Public-Key Infrastructure (PKI), employed to issue a certificate for all network entities supporting secure communications. Communications are secure in terms of integrity and confidentiality, while a set of other security-related mechanisms are proposed to secure communications in the network, such as VPN, firewalls and SSH. In [7], the authors consider the addition of a third element to the LoRaWAN architecture, which is independently managed and used to create and distribute keys among the participating devices. In this proposal, end devices receive only their own session key, and the NS is unaware of the various session keys, which are managed at the application level. In addition, this work proposes to change the AppKey in each session, in order to prepare for difficult attacks. Each device stores the last timestamp received and compares it against the timestamp value of the next message, in order to prevent against replay attacks.

Authors in [8] consider Blockchain to improve trust in the network server, as well as to contribute to improving the overall security of the network. Blockchain must be supported by the NS, frequently the only device with the required resources for this purpose, in contrast with end devices and gateways. In this proposal, a management component is added to the NS, which adds a hash of each transmitted message to the Blockchain. In [9], the authors propose to use Ephemeral Diffie–Hellman Over COSE (EDHOC) to renew the keys employed for encryption of communications between sensing devices. EDHOC is a symmetric key management protocol employed between end devices and is based on SIGMA-I and implements the Elliptic Curve Diffie–Hellman algorithm (ECDH) with P-256. This new mechanism is used to renew session keys as soon as the device is activated, adding flexibility to the entire key management process. Another approach considered in [10] proposes that key management can be performed through public keys without the use of certificates, thus implementing Certificateless Public-Key Cryptography (CL-PKC). This is a lightweight method in terms of the memory and computational resources required and is therefore considered a good choice for IoT networks, given the typical constraints of the devices employed. In [11], the authors address the formal evaluation of LoRaWAN security against known threats. The authors employ the Scyther tool to investigate and expose weaknesses in LoRaWAN security. Besides the enumeration of vulnerabilities, the authors also discuss possible approaches to each of the identified vulnerabilities.

In [12], the authors focus on jamming attacks against LoRaWAN environments. To mitigate this threat, the authors propose a LoRaWAN-based IDS module deployed on gateways. This component monitors join request traffic between end nodes, in order to detect jamming attacks. In [13], it is proposed to use two activation keys. This method is very similar to the one specified by the LoRa Alliance; however, it uses yet another preconfigured key to segregate the key management functions of the network and application layers. This approach uses a key (AppKey) to generate a session key between the end device and the

AS, and another key (NwkKey) to generate the session key to be used between the end device and the NS. This approach discusses the exposure of the network to attacks, due to the fact that the NS is able to access application-level session keys. A mechanism for updating session keys is also proposed to increase system security, using the session keys currently used at each layer to create new session keys.

A review of vulnerabilities, threats and common defense strategies in some LPWAN protocols, in particular LoRaWAN and SigFox, is performed in [14]. This work discusses critical aspects of security in LPWAN technologies and identifies attack vectors, while also proposing defense strategies to mitigate the identified risks in the context of relevant IoT technologies.

In [15], the authors propose a new method for multihop data routing, which employs Q-learning in the context of data transfers. The main goal is to improve the performance of the network regarding quality of service and energy consumption, a core concern when referring to IoT applications. The results of the experiment are compared against existing modules, and the new proposal is verified to be very efficient in terms of energy savings, with lower data interference and improved quality of service. The authors in [16] discuss the importance of managing energy and fast data transfers in wireless sensor networks (WSN) and propose a new data routing method optimizing the energy cost of the links. This strategy allows one to generate uniform energy consumption and promote faster data transfers. The evaluation considers two strategies: simulation, with exhaustive analysis of network lifetime, energy and data latency, and deployment over a test bed, which demonstrated that the proposed cognitive small world model is more efficient regarding energy balancing and consumption, data latency and network lifetime, when compared with different state-of-the-art approaches.

From our analysis on related work, we may observe that, although various proposals address security weaknesses in IoT LoRaWAN environments, there is a lack of proposals addressing the dynamic management of security and energy, as per the goals of our proposal in this article. As previously discussed, our main motivation is to design and evaluate a solution to address, in tandem, these two important and (from the point of view of the resources available in constrained IoT devices) conflicting aspects. As we address throughout our discussion in this article, the dynamic management of energy and security in the context of an IoT application may provide the required security to protect the network against threats being detected at the moment, while allowing for a more careful and intelligent management of energy during the life cycle of the sensing devices employed—a valuable resource in providing acceptable lifetime to the IoT application at hand.

3. An Architecture for Dynamic Energy and Security Management in IoT

The architecture in the context of which we design and evaluate the proposed energy and security adaptation strategy is illustrated in Figure 2. This architecture extends the LoRaWAN operational model by adding new elements considered to be indispensable for the management of security and energy in tandem, as per our goals. One important component is the Network Server (NS), and in this case, it preserves the base functionality of a LoRaWAN network server. The NS is responsible for filtering information from the sensors and forwarding the respective payloads to the Application Server (AS). As illustrated, the dynamic safety controller is added in the context of this component of the architecture. An IDS module is added at the Gateway to assist the controller, in particular to allow the assessment, from the analysis of the communications with sensing devices, of the state of the network in relation to intrusions and external attacks. We can also observe that a component for security management is added to the various entities of the architecture, which is responsible for sending information about the security scheme currently being used, as well as for receiving information important for the enforcement of a new security scheme (as decided by the controller). In Figure 2, we also illustrate the data flow for the variables

used by the controller, as well as the distribution of the new security scheme, illustrating in general what happens in the system in each periodic run of the controller logic.

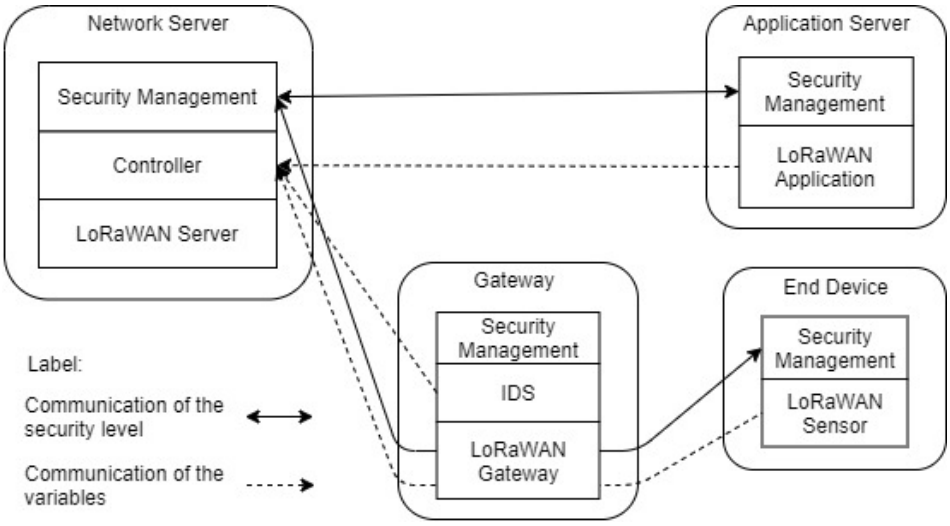

Figure 2. System view of the proposed architecture.

In Figures 3 and 4, we illustrate a functional and protocol view of the proposed architecture, particularly the data flows, the origin, the destination and the type of information exchanged between the various entities of the architecture illustrated in Figure 2.

In Figure 3, we illustrate the process of setting up the controller. In this process, the security levels defined in the controller are shared with the remaining components of the architecture, which make use of this information, in particular, the sensing devices and the AS. The attacks and the security profile tables are also shared, which are required to decode the attack reports transmitted by the IDS component of the architecture and adapt accordingly.

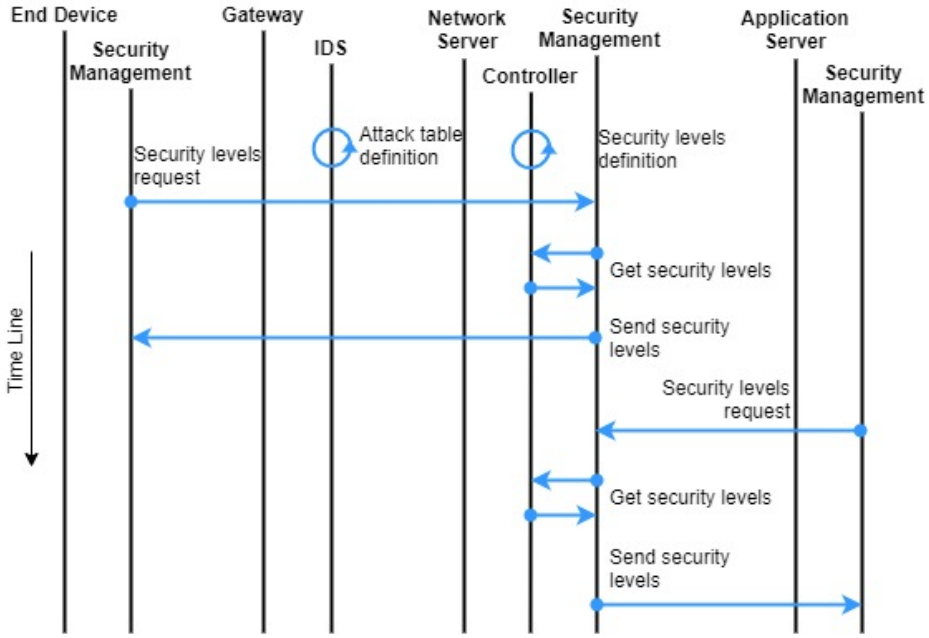

Figure 3. Setup phase of the proposed architecture (protocol view).

In Figure 4, we illustrate the process of exchanging the necessary information for the operation of the controller, in each run of the algorithm. Each sensing device part of the architecture outputs its battery level (in percentage), and the IDS transmits the security

assessment made at that moment from its analysis of the communications environment. The minimum and initial security levels, as well as the preference threshold (which expresses the preference given to energy saving over security), as well as the security profile table, are transmitted by the AS, since all variables depend on system preferences. Such preferences influence the behavior of the controller and may be expressed via application configuration, either on setup or modified at run time. We must note that this is implementation specific and out of the scope of our implementation and evaluation process. In each run (one per minute), the controller gathers all the required information, analyzes it and reacts to the IDS report and end-node status (battery level), adapting the security level as appropriate. Afterward, the new security scheme is transmitted to the sensor, NS and AS entities of the architecture, using encryption at the network and application layers provided by LoRaWAN.

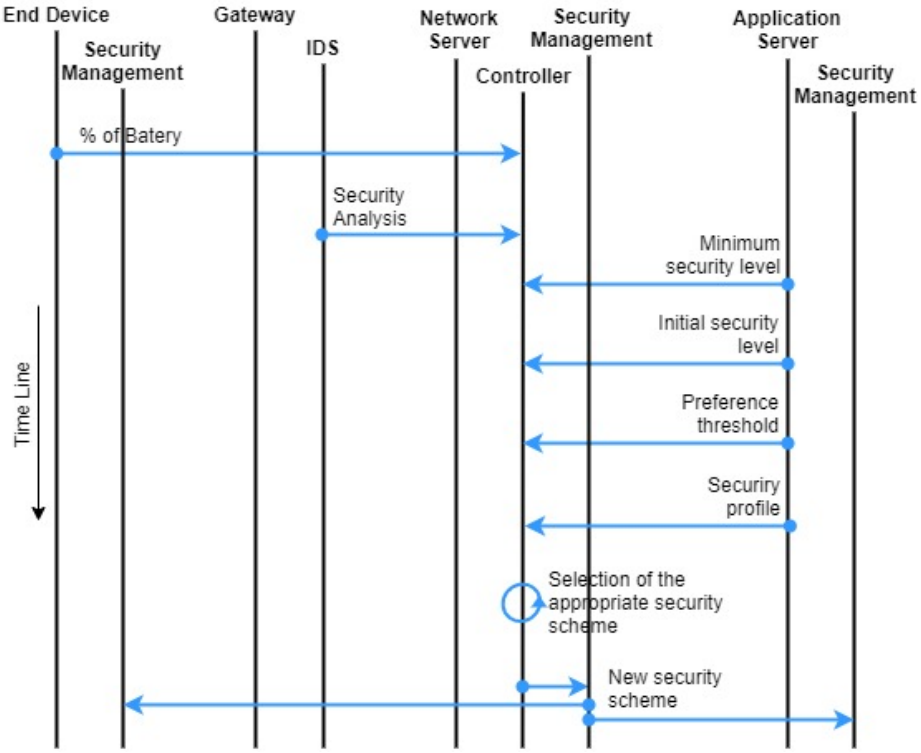

Figure 4. Operational phase the proposed architecture (protocol view).

We need also to consider which LoRaWAN device classes are able to support the proposed architecture. From our previous description in Section 1, we may note that Class B and C devices will have no problem using this approach, as they have synchronous or continuous downlink opportunities to transmit. Thus, server-to-device communications will easily allow the transmission of the new security scheme. Regarding Class A devices, in this case, the device is responsible for initiating the conversation, and there are always two downlink windows. Therefore, we may observe that this new architecture is also suitable for Class A devices.

As previously discussed, the main motivation behind our proposal is to secure the network against external attacks, by adapting security and energy dynamically, in order to deal with relevant threats, while trying to extend the battery life of constrained devices as feasible. The design of this proposal takes into account the different preferences that each application may have, which may be specific to the application itself or the environment supporting that particular application. Therefore, in order to make an adequate management of security and energy for the application, several profiles describing energy over security preferences were considered and are discussed next. The various profiles express different approaches and priorities, in what respects compromises between increasing

security and promoting savings in energy. While some applications may require higher security in the presence of attacks, even at the expense of energy, for others, the lifetime of the devices may be more important. Overall, the approach described next addresses the dynamic management of energy and security in light of the requirements of the applications, as well as of the operational conditions of the devices and the presence (or absence) of attacks, as we discuss next.

4. Dynamic Security and Energy Adaptation

The dynamic security controller is a strategic component of the proposed architecture and was developed with the goal of addressing the challenge of dynamically managing energy and security, in the context of IoT constrained sensing devices. The controller is designed to run periodically, either synchronously or asynchronously, depending on the requirements of the application at hand. The controller allows the modification of the security scheme currently active, thus outputting the indication of a new security scheme that best suits the system at that particular moment, in order to guarantee a more articulated management between security and energy consumption in the context of the application. Thus, one important motivation is to extend battery life, while without jeopardizing the security of the system. In the presence of an attack, the system may increase the security level currently in use, providing additional security against threats known by the controller. We start by describing the main requirements considered for the development of the controller.

4.1. Security and Functional Requirements

The scheme proposed to secure communications against external attacks is calculated by the controller, from information on the state of the network, as previously discussed. For this purpose, various items of information are collected from several sources and used as inputs to the controller:

- Percentage of energy remaining in the battery of the sensing device;
- Initial security level determined for the network at hand (configured in the setup phase), as defined in Table 1;
- Minimum security level allowed by the application for this network, as defined in Table 1;
- Preference threshold, which represents the importance that the application puts on the device's battery longevity (a percentage value);
- Security assessment performed by the IDS module, which reports that a particular attack has been detected. Attacks are identified in a numerical scale, from 1 up to a configured value, with each index representing a particular type of attack;
- Security profile, expressed as a table that matches each index for an external attack reported by IDS with the impact that this attack has on the system at hand (expressed as a percentage). The setup of this table is also conducted in the context of the configuration of the application;

It is important to note that both the initial and minimum security levels considered for the application are from Table 1 and defined for the application at hand during its setup. As can be observed, the various levels assure different and complementary security properties for the application. This strategy allows one to express the preference and requirements of different applications, regarding the relative importance attributed to security and energy savings.

In each run of the controller's logic and from the various inputs, the controller outputs the best security scheme to apply to the system at that particular instant in time. Each security level combines a certain level of confidentiality (using symmetric Advanced Encryption System (AES) encryption) with integrity, in this case supported by the Secure Hash Algorithms (SHA), as identified in Table 1.

Table 1. Security profiles considered for the proposed architecture.

Level	Security Scheme		Energy Saving	Security Level
	Confidentiality	Integrity		
1	-	-	Very Strong	Weak
2	-	SHA256	Strong	Moderate
3	AES128	-	Strong	Moderate
4	AES128	SHA256	Moderate	Strong
5	AES256	SHA512	Weak	Very Strong

The usage of different security profiles allows one to employ the controller's logic with applications requiring different security and energy preferences and management approaches. Next, we discuss the algorithm developed to support the dynamic management of security an energy, in the context of the proposed architecture.

4.2. Controller Logic

The verification of the preference threshold defined at the application level is the first step considered by the controller logic algorithm. Depending on this value, an appropriate calculation flow is taken using the impact percentage. Next, the energy remaining in the device's battery is verified, and an attenuation is applied to the previous calculation.

The battery percentage is divided into three intervals in the controller's logic: [100%, 75%],]75%, 25%[and [25%, 0%]. These intervals are chosen to reflect three representative phases in the typical discharge curve of an alkaline battery, as discussed in [17]. Since one of the goals of our proposal is to extend the lifetime of sensing devices by promoting energy savings, the three considered intervals allow the algorithm to react differently in such situations, with respect to the security profile chosen. In particular, this means that the algorithm, depending on the battery level at each moment, can reduce the security scheme to be suggested in three attenuation levels: least attenuation, medium attenuation and heavy attenuation, respectively.

In our implementation and evaluation, we considered two versions of the algorithm. The first version, identified subsequently as *Logic*, is illustrated by Algorithm 1. In the other version, identified subsequently as *Delta Logic*, we modify the algorithm by adding, between lines 10 and 11, an instruction to store a given number of previously transmitted security assessments (detection of an attack by an IDS module in the network) in a buffer. In this version, when a new security assessment is received from the IDS, it is stored in a buffer, and the algorithm considers instead the highest recorded assessment present in that buffer to perform the necessary calculations. This version of the algorithm arises from the need to absorb temporary anomalies that may exist in the security assessment performed by the IDS, thus allowing to filter false positives and smooth the behavior of the controller when appropriate. We proceed to discuss our initial approach to evaluate the behavior of the described algorithm.

Algorithm 1 Controller logic.

```
Declare main variables
Declare auxiliary variables
Read initial configuration from text file
Allocate required memory
Initialize the result files
Write initial configuration in the result files
Apply the energy cost of the initial level
while Battery percentage decrease is less than 5% do
    Calculate new battery percentage
    Get security assessment from IDS
    Calculate security level to be suggested
    Apply attenuation depending on the battery level
    Remove energy cost of the new security
    Save the data in the corresponding file
    Increase cycle number counter
end while
Display number of cycles
Free up space and close the files
```

5. Analytical Evaluation of the Controller

For the evaluation of the controller in the context of the proposed architecture, we considered two complementary approaches. For both approaches, it was necessary to specify the requirements of the application at hand, which are translated to specific inputs of the controller. In this context, the table of attacks was established as shown in Table 2, identifying four types of relevant attacks. As already discussed, the indexes in this table are used by the IDS to report the detection of a particular type of attack.

When the Controller receives a notification about an attack reported by the IDS module, it is necessary to translate the index identifying the attack to a percentage of impact that the detected attack potentially represents for the application at hand. For this purpose, the security profile was defined as shown in Table 3, with Index 0 corresponding to the absence of attacks, and with the attacks considered identified as illustrated in Table 2.

Table 2. External attacks considered (evaluation).

Index	Attack
1	Spoofing
2	Hijacking
3	Denial of Service
4	Man in the Middle

Table 3. Security profile (impact of attacks).

Index	Impact Value
0	0%
1	40%
2	90%
3	20%
4	80%

Again, we note that this is configurable for the application at hand, given that, in practice, different attacks may represent different risks and have different impacts to different applications.

In fact, a particular type of external attack can have a different impact depending on the purpose of each LoRaWAN application, and one motivation of our strategy is to ensure that the proposed architecture is as inclusive as possible of LoRaWAN applications with different requirements and goals.

5.1. Analytical Evaluation of Controller Behavior

We start by considering the two previously described approaches against a set of scenarios and configuration settings. The focus in our evaluation is on the impact of security in the energy available in sensing devices, given that it determines the lifetime of the device and, in consequence, of the application itself.

5.1.1. Evaluation Setup

To test the behavior of the proposed controller, we consider various scenarios combined with different configuration setups. We also assume that each application-layer message requires 51 bytes, and a rate of two messages per run. The preference threshold, the minimum level and the initial level correspond to the variables changed in the initial settings of each test, and Table 4 illustrates the corresponding values.

Table 4. Initial settings considered for evaluation.

ID	Preference Threshold	Minimum Level	Initial Level
1	80%	2	4
2	20%		
3	80%	2	3
4	20%		
5	80%	1	4
6	20%		

The scenarios considered in our implementation and evaluation of the proposed adaptation logic are as follows:

- A scenario with random attacks, both in what regards the type of attack, the time of its occurrence and its duration;
- Two scenarios with 20% of high risk attacks, namely with 80% and 90% of impact;
- A scenario with 20% of medium impact attacks, namely with 40%;
- A scenario with 20% of attacks with low impact, in this case with 20%.

With regard to the evaluation study, the controller was implemented in the C language for each approach described in Section 4.2. The goal of our evaluation is to determine its effectiveness in implementing energy-aware security adaptation, as per the goals previously discussed. For this study, we consider the energy consumption impact of the AES and SHA algorithms [18], illustrated in Table 5. As for the battery capacity of the sensor, we consider the usage of an AA LR6 alkaline battery with a capacity of 1.8 Ah and a voltage of 1.5 V, which corresponds to 2.7 Wh of energy generated and 9720 Joules. It is important to note that this setup still does not take into account any LoRa/LoRaWAN configuration regarding energy consumption, only the energy cost of the cryptography process.

Table 5. Energy cost for the various security levels.

Level	Security Scheme		Energy Cost (microJ/B)
	Confidentiality	Integrity	
1	-	-	0
2	-	SHA256	0.043
3	AES128	-	0.245
4	AES128	SHA256	0.288
5	AES258	SHA512	0.375

Each algorithm was used to test all the considered scenarios, in particular considering the six combinations of initial configurations presented in Table 4, in total accounting for 60 tests. Given the high amount of time required to perform each test and considering the typical discharging curve of an alkaline battery [17], measurements were made only in three battery intervals: from 100% to 95%, 75% to 70% and from 25% to 20%. We therefore consider the behavior of the algorithm in the three distinct and representative phases of the lifetime of a battery. The final value corresponds to an extrapolation of the values obtained in these three intervals. Next, we proceed with a discussion on the results obtained in our evaluation tests.

5.1.2. Results and Analysis

In Table 6, we identify the values obtained in each of the tests, representing the number of cycles that could be performed for a given test approach, scenario and initial configuration. The results presented are extrapolated from the measured values, using expression (1).

$$5 \times Firstinterval + 10 \times Secondinterval + 5 \times Thirdinterval \quad (1)$$

The different weight attributed to each of the three intervals is equivalent to the number of times that each interval incorporates 5% of battery; that is, between 100% and 75% or between 25% and 0% there are 5 intervals of 5%, and between 75% and 25% there are 10 intervals of 5%. The results obtained are compared with the number of cycles that can be obtained using a fixed security scheme, helping to understand how far the goals of the proposed algorithm (saving battery and extending the life of the sensors) have been achieved.

For the scenario with a fixed security scheme, we consider security level 4, as defined in Table 5, where we can also obtain the energy cost per Byte inherent to its usage. With the same conditions used for the tests previously presented, we obtain an energy cost of 0.073152 mJ per execution. We consequently note that the value allows a total of 132,874,016 cycles to be achieved.

Table 7 identifies the percentage comparison between the values obtained in the tests and the value previously computed. Positive percentage values correspond to a gain in relation to the number of cycles performed with a fixed security scheme, while negative values correspond to a loss. When analyzing the results illustrated in Table 7, we observe that 55 of the 58 tests have resulted in higher values than intended, proving that the dynamic security controller has a positive contribution to save energy. The three results (obtained with the random scenario, *Delta Logic* algorithm and with the initial configurations of 2, 4 and 6) that were below expectations and result in a negative percentage correspond to tests in which preference was for security enforcing over saving energy, resulting in an increase in power usage to allow for increasing the security level (enabling a higher security configuration). The two blank spaces correspond to tests that are impossible to perform since, given the conditions, no security scheme would be used.

Table 6. Evaluation results (number of cycles that occurred in each test).

		Initial Configuration					
Scenario	**Algorithm**	**1**	**2**	**3**	**4**	**5**	**6**
Random	Logic	306,698,115	208,301,610	306,698,150	208,301,610	379,677,515	224,068,215
	Delta Logic	148,947,695	117,738,690	148,947,770	117,738,710	149,992,340	117,797,305
High 90%	Logic	398,231,705	362,629,505	398,231,705	362,629,475	623,803,770	539,511,990
	Delta Logic	157,904,815	137,406,450	157,904,925	137,406,470	165,193,705	142,853,115
High 80%	Logic	468,039,540	390,846,515	468,039,545	390,846,515	1,124,969,280	604,671,550
	Delta Logic	254,661,730	153,298,930	254,661,750	153,298,950	297,894,645	160,122,900
Medium 40%	Logic	889,946,800	553,140,925	889,946,820	553,140,930	3,337,300,455	1,660,158,380
	Delta Logic	889,946,660	360,221,480	889,946,780	360,221,720	883,662,985	439,623,500
Low 20%	Logic	17,798,937,782	17,798,937,782	17,798,937,858	17,798,937,858		88,994,688,908
	Delta Logic	17,798,937,782	17,798,937,782	17,798,937,858	17,798,937,858		88,994,688,908

Table 7. Test results compared against the scenario with fixed security.

		Initial Configuration					
Scenario	**Algorithm**	**1**	**2**	**3**	**4**	**5**	**6**
Random	Logic	306,698,115 131%	208,301,610 57%	306,698,150 131%	208,301,610 57%	379,677,515 186%	224,068,215 69%
	Delta Logic	148,947,695 12%	117,738,690 −11%	148,947,770 12%	117,738,710 −11%	149,992,340 13%	117,797,305 −11%
High 90%	Logic	398,231,705 200%	362,629,505 173%	398,231,705 200%	362,629,475 173%	623,803,770 369%	539,511,990 306%
	Delta Logic	157,904,815 19%	137,406,450 3%	157,904,925 19%	137,406,470 3%	165,193,705 24%	142,853,115 8%
High 80%	Logic	468,039,540 252%	390,846,515 194%	468,039,545 252%	390,846,515 194%	1,124,969,280 747%	604,671,550 355%
	Delta Logic	254,661,730 92%	153,298,930 15%	254,661,750 92%	153,298,950 15%	297,894,645 124%	160,122,900 21%
Medium 40%	Logic	889,946,800 570%	553,140,925 316%	889,946,820 570%	553,140,930 316%	3,337,300,455 2412%	1,660,158,380 1149%
	Delta Logic	889,946,660 570%	360,221,480 171%	889,946,780 570%	360,221,720 171%	883,662,985 565%	439,623,500 231%
Low 20%	Logic	17,798,937,782 13295%	17,798,937,782 13295%	17,798,937,858 13295%	17,798,937,858 13295%		88,994,688,908 66877%
	Delta Logic	17,798,937,782 13295%	17,798,937,782 13295%	17,798,937,858 13295%	17,79,8937,858 13295%		88,994,688,908 66877%

5.2. Analytical Evaluation in the Context of LoRaWAN

The second evaluation we consider in our study aims to present results in the context of a practical LoRaWAN application. We now consider also the energy costs of transmissions, discussed in [19]. The values we are considering in this evaluation were obtained with a LoRaWAN Class A device. This class is the most relevant for our analysis since it is the most

efficient regarding energy usage. Ref. [19] also compares how different spreading factors affect power consumption in LoRaWAN, with SF = 12 being the most expensive in terms of energy usage and the worst case in power consumption. For that reason, the chosen results for this evaluation consider a SF = 12, given that it provides support for a worst-case scenario analysis of the behavior of the algorithm. Transmission costs in terms of energy are thus used together with the cost of the various security levels decided by the controller. We start by addressing the configuration of the scenarios considered in this evaluation.

5.2.1. Configuration of the Test Scenarios

To carry out this analytical evaluation, it was necessary to define the scenarios to be tested, their initial configurations and the logic implemented in each evaluation. We tested two different scenarios: the first where no attacks on the system are recorded, and another where there are constant attacks against communications, each with a high impact of 80%. The preferential factor, which represents the importance attributed to battery savings by the application, was tested with values of 80% (which gives priority to battery savings over security) and 20% (the opposite scenario), and the minimum level of security for levels 1 and 2. The size of each message is 51 Bytes, the maximum payload size used for LoRaWAN configurations where SF is 12. The transmission rate considered is of one message every half a minute. The battery characteristics are also maintained, considering the same type of battery (with 9720 Joules available) and applicable discharge curve.

The value of the transmission energy cost is of 0.59 mJ per bit [19] and equals a total of 6.136 mJ, for values measured for a LoRa SX1272 transmitter, which is common in IoT sensors as those employed in LoRaWAN environments. The values used for the energy cost of security are the same as those used in the previous tests, illustrated in Table 5. In this evaluation, the scenario in which the security of the system does not change was also analyzed, and in this case, we chose the fourth security scheme from Table 5, with the objective of comparing the remaining results with the currently most common usage scenario of security in a LoRaWAN network. Please note that the preference threshold and minimum security level settings are not applicable here, as these variables are only used by the Controller.

5.2.2. Results and Analysis

In Table 8, we present the results obtained for each of the tests performed, considering the energy impact of data transmission and security, as previously discussed. Each combination of scenario and initial configuration setup was analyzed to determine the battery percentage intervals where each security level would be suggested, presented in the fifth and sixth columns of Table 8. With this division, it was possible to calculate the different energy cost over time for the life of the battery and thus obtain the time of execution of each test.

Table 8. Results for the analytical evaluation of the Controller in the context of LoRaWAN, + (number of cycles and total time for tests).

Scenarios	Initial Settings			Battery Range	Security Level	Energy Cost of Security (mJ)	Energy Cost of Transmission (mJ)	Total Number of Executions		Total Time of Execution (minutes)	Converted Total Time of Execution
	Preference Threshold	Minimum Level of Security	Initial Level								
Fixed Security	Not Applicable	Not Applicable	4	100–0%	4	0.019	6136	1,610,165		789,647	559 d 2 h 2 m
Constant attacks with 80% impact each	80%	1		100–59%	4	0.019		660,168	1,580,534	790,267	559 d 12 h 47 m
				59–22%	3	0.016		596,060			
				22–2%	2	0.002		322,923			
				2–0%	1	0		32,303			
		2		100–59%	4	0.019		660,168	1,580,523	323,704	559 d 12 h 41.5 m
				59–22%	3	0.016		596,060			
				22–0%	2	0.002		355,215			
	20%	1		100–74%	5	0.024		418,259	1,579,088	789,544	559 d 0 h 15 m
				74–21%	4	0.019		853,388			
				21–0%	3	0.016		338,304			
		2		100–74%	5	0.048		418,259	1,579,088	789,544	559 d 0 h 15 m
				74–21%	4	0.019		853,388			
				21–0%	3	0.016		338,304			
Without attacks	Indifferent	1		100–0%	1	0		1,615,155		807,577	560 d 19 h 37 m
		2		100–0%	2	0.002		1,614,618		807,308	560 d 15 h 9 m

Table 9 presents a comparison, in percentage value, between the values obtained in the tests made for the different scenarios and an environment in which the security scheme employed is fixed. The positive percentage values represent a gain in the lifetime of the sensor, and the negative values represent a loss in the execution time. It is possible to notice that only two tests out of six revealed negative percentage values, and in this situation, we may consider that the controller plays an important role in the dynamic adaptation of energy and security, in the context of a LoRaWAN infrastructure. This is because, on the one hand, the proposed architecture is very helpful in saving energy, in situations where the conditions of the network allow it, given the limited or no existence of external attacks. On the other hand, when the network is put to the test given the severity of the intrusion, this algorithm allows the infrastructure to achieve the necessary additional protection, something that would not be possible in a LoRaWAN network using a fixed security scheme.

Table 9. Lifetime of a sensor taking into account the energy cost of security and message transmission.

Scenario	Initial Configuration		Total Number of Executed Cycles	Total Time of Execution (minutes)	Comparison
	Preference Threshold	Minimum Level of Security			
Fixed Security	Not applicable	Not applicable	1,579,294	789,647	-
Impact of 80%	80%	1	1,580,534	790,267	0.08%
		2	1,580,523	790,262	0.08%
	20%	1	1,579,088	789,544	−0.01%
		2	1,579,088	789,544	−0.01%
No attack	Indifferent	1	1,584,094	792,047	0.31%
		2	1,583,578	791,789	0.28%

Observing the initial configurations and the scenario of the two tests, it is noticeable why these negative values were obtained, since in this case, we have a scenario with constant attacks and consequently with a high impact in data protection. In addition, the preference threshold is chosen to be 20%, since in this case, data security is preferred over savings in energy. This results in choosing a higher security level and, in consequence, increasing the energy required to support the new configuration. The different values obtained for the same scenario result from the application of different conditions and algorithms, reinforcing the existence of a differentiation in the treatment of security and energy savings, which depends on the controller's input values. The larger gain is always recorded with a preference threshold factor of 80% and a minimum security level of 1, which means that the algorithm chooses to save battery over security and can decide to lower or even turn security off, according to what is most appropriate given the current conditions of the network and the energy remaining in the battery of the device. On the other hand, the loss (or the lowest gain) is recorded when the preference threshold is equal to 20% and the minimum level of security is 2, which represents a controller approach focused on security instead of saving energy. In this situation, a security scheme is employed even if no attacks on the system are detected. There are some factors that, in practice, can influence the gains obtained by using the controller, such as the size of each message, the frequency of message transmission and the available energy in the device's battery.

6. Conclusions and Future Work

In this article, we addressed the proposal of an architecture with the goal of supporting the dynamic adaptation of energy and security, particularly focusing on IoT applications supported by devices powered by batteries. An important component of our proposal

resides on the implementation of the logic required for the dynamic management and adaptation of security and energy. The proposal was implemented and evaluated experimentally and analytically, in light of the applications and communication environments with particular requirements and configurations. Our evaluation allowed us to perceive the advantage behind the usage a dynamic adaptation approach, since this allows one to intelligently manage two factors: being able to save battery and consequently extend the lifetime of the device when no external threats are detected, while being able to increase the security of the system if the environment is more hostile and attacks against the normal operation of the application are detected.

We observed there is a general lack of proposals in this area, particularly in what respects the dynamic management of security and energy in IoT LoRaWAN constrained environments. In fact, most of the existing proposals focus on security-related aspects such as key management, authentication and intrusion detection. For sure, our work leaves room for new assessments to be made and for further work to be conducted with the goal of extending the implementation of the proposed architecture. For example, an experimental evaluation of a complete implementation of the architecture in a real environment can be performed, allowing to test the proposed approach against complementary system requirements. Other approaches may be to consider different architectural approaches for the placement of critical components of the architecture, as well as to develop and evaluate other approaches for the logic behind the adaptation performed by the controller, in light of particular classes of attacks.

Author Contributions: M.R. and J.G. contributed equally to the paper except for the development of the experimental scenario which was made by M.R. All authors have read and agreed to the published version of the manuscript.

Funding: This work is funded by national funds through the FCT-Foundation for Science and Technology, I.P., within the scope of the project CISUC-UID/CEC/00326/2020 and by European Social Fund, through the Regional Operational Program Centro 2020.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Chaudhari, B.S.; Zennaro, M.; Borkar, S. LPWAN technologies: Emerging application characteristics, requirements, and design considerations. *Future Internet* **2020**, *12*, 46. [CrossRef]
2. Hellaoui, H.; Koudil, M.; Bouabdallah, A. Energy-efficient mechanisms in security of the internet of things: A survey. *Comput. Netw.* **2017**, *127*, 173–189. [CrossRef]
3. Alliance, L. *A Technical Overview of LoRa ® and LoRaWAN ™ What Is It*? Technical Report; LoRa Alliance: Vancouver, BC, Canada, 2015.
4. Noura, H.; Hatoum, T.; Salman, O.; Yaacoub, J.P.; Chehab, A. LoRaWAN security survey: Issues, threats and possible mitigation techniques. *Internet Things* **2020**, 100303. [CrossRef]
5. de Carvalho Silva, J.; Rodrigues, J.J.; Alberti, A.M.; Solic, P.; Aquino, A.L. LoRaWAN—A low power WAN protocol for Internet of Things: A review and opportunities. In Proceedings of the 2017 2nd International Multidisciplinary Conference on Computer and Energy Science (SpliTech), Split, Croatia, 12–14 July 2017; pp. 1–6.
6. Oniga, B.; Dadarlat, V.; De Poorter, E.; Munteanu, A. A secure LoRaWAN sensor network architecture. In Proceedings of the 2017 IEEE SENSORS, Glasgow, UK, 29 October–1 November 2017; pp. 1–3.
7. Naoui, S.; Elhdhili, M.E.; Saidane, L.A. Trusted third party based key management for enhancing LoRaWAN security. In Proceedings of the 2017 IEEE/ACS 14th International Conference on Computer Systems and Applications (AICCSA), Hammamet, Tunisia, 30 October–3 November 2017; pp. 1306–1313.
8. Lin, J.; Shen, Z.; Miao, C. Using blockchain technology to build trust in sharing LoRaWAN IoT. In Proceedings of the 2nd International Conference on Crowd Science and Engineering, Beijing, China, 6–9 July 2017; pp. 38–43.
9. Sanchez-Iborra, R.; Sánchez-Gómez, J.; Pérez, S.; Fernández, P.J.; Santa, J.; Hernández-Ramos, J.L.; Skarmeta, A.F. Enhancing lorawan security through a lightweight and authenticated key management approach. *Sensors* **2018**, *18*, 1833. [CrossRef] [PubMed]

10. Tiwari, S.; Patel, H.B.; Shrimali, B. A Survey on Certificate-Less Public Key Encryption for Authentication in a Smart IoT-Based LoRaWAN. *IOSR J. Eng.* **2018**, *8*, 22–28.
11. Eldefrawy, M.; Butun, I.; Pereira, N.; Gidlund, M. Formal security analysis of LoRaWAN. *Comput. Netw.* **2019**, *148*, 328–339. [CrossRef]
12. Danish, S.M.; Nasir, A.; Qureshi, H.K.; Ashfaq, A.B.; Mumtaz, S.; Rodriguez, J. Network Intrusion Detection System for Jamming Attack in LoRaWAN Join Procedure. In Proceedings of the 2018 IEEE International Conference on Communications (ICC), Kansas City, MO, USA, 20–24 May 2018. [CrossRef]
13. Kim, J.; Song, J. A dual key-based activation scheme for secure LoRaWAN. *Wirel. Commun. Mob. Comput.* **2017**, *2017*, 6590713. [CrossRef]
14. Torres, N.; Pinto, P.; Lopes, S.I. Security Vulnerabilities in LPWANs—An Attack Vector Analysis for the IoT Ecosystem. *Appl. Sci.* **2021**, *11*, 3176. [CrossRef]
15. Pandey, O.J.; Yuvaraj, T.; Paul, J.K.; Nguyen, H.H.; Gundepudi, K.; Shukla, M.K. *Improving Energy Efficiency and QoS of LPWANs for IoT Using Q-Learning Based Data Routing*; IEEE: Piscataway, NJ, USA, 2021.
16. Pandey, O.J.; Hegde, R.M. Low-Latency and Energy-Balance Data Transmission Over Cognitive Small World WSN. *IEEE Trans. Veh. Technol.* **2018**, *67*, 8. [CrossRef]
17. Mikhaylov, K.; Tervonen, J.; Fadeev, D. Development of energy efficiency aware applications using commercial low power embedded systems. In *Embedded Systems-Theory and Design Methodology*; IntechOpen: London, UK, 2012; pp. 407–430.
18. Banerjee, U. *Energy-Efficient Protocols and Hardware Architectures for Transport Layer Security*; Technical Report; Massachusetts Institute of Technology: Cambridge, MA, USA, 2017.
19. Bouguera, T.; Diouris, J.F.; Chaillout, J.J.; Jaouadi, R.; Andrieux, G. Energy Consumption Model for Sensor Nodes Based on LoRa and LoRaWAN. *Sensors* **2018**, *18*, 2104. [CrossRef] [PubMed]