Internet of Things 12 (2020) 100303

Contents lists available at ScienceDirect

Internet of Things

journal homepage: www.elsevier.com/locate/iot

Review article

LoRaWAN security survey: Issues, threats and possible mitigation techniques

Hassan Noura[a], Tarif Hatoum[a], Ola Salman[b,*], Jean-Paul Yaacoub[b], Ali Chehab[b]

[a] Arab Open University, Department of Computer Studies, Lebanon
[b] American University of Beirut, Electrical and Computer Engineering, Lebanon

ARTICLE INFO

Article history:
Received 4 May 2020
Revised 23 July 2020
Accepted 29 September 2020
Available online 5 October 2020

Keywords:
LoRaWAN security
LoRaWAN applications
LoRaWAN security attacks

ABSTRACT

With the emergence of IoT, new communication technologies have been proposed to cope with the IoT large scale, and the "things" constraints in terms of power and processing resources. One of these technologies is LoRaWAN, which aims at providing very-long-range transmission with low power consumption. However, this technology suffers from several security and privacy vulnerabilities that could compromise availability, authentication, and privacy. In fact, security and privacy are major concerns in any IoT network. Consequently, effective countermeasures are highly needed to enable LoRaWAN's wide adoption in the IoT domain. In this paper, we review the LoRaWAN architecture, applications, and security concerns. In addition, we list several possible countermeasures to address the existing LoRaWAN vulnerabilities and thus, to prevent the related attacks.

1. Introduction

Recently, LoRaWAN has been proposed to enable IoT low data-rate applications that require to transmit and receive small amount of data with a range of few Kilometers. In fact, the data rate in LoRaWAN can vary between 0.3 and 50 kbps. The maximum achievable data rate depends on the receiver range and the environment interference level. The LoRaWAN technology is designed to be power efficient to cope with the power constraints of IoT devices. Also, LoRaWAN operates within unlicensed frequency bands, ranging between 868 MHZ and 900 MHZ, which compensates the licensing cost and makes this technology affordable. However, transmitting at low frequencies and long range is restricted in some regions. Consequently, the LoRa Alliance has defined different frequency plans that vary on a regional basis. Furthermore, in the same region, there might be a difference in the permitted transmission frequency bands based on countries' regulations. Generally, LoRaWAN uses a wide bandwidth which helps to resist both the interference and noise.

On the other hand, LoRaWAN uses the star topology as shown in Fig. 1. IoT devices are connected to LoRa End-devices (EDs), which are connected directly to one or many gateways (GWs). Each GW is connected to the network server (NS), which can be connected to one or many application servers (ASs). The connection between LoRaWAN EDs and GWs is a LoRaWAN wireless communication, while the connection between the GW and the AS is an IP connection [1].

Additionally, an ED can be connected to a LoRa GW by using one of the following activation techniques: Over-The-Air-Activation (OTAA), or Activation by Personalization (ABP). After the activation phase, the EDs start communicating their messages to the GW. However, these messages should be authenticated and encrypted by using a network key *NwKey* and

* Corresponding author.
E-mail address: oms15@mail.aub.edu (O. Salman).

https://doi.org/10.1016/j.iot.2020.100303
2542-6605/

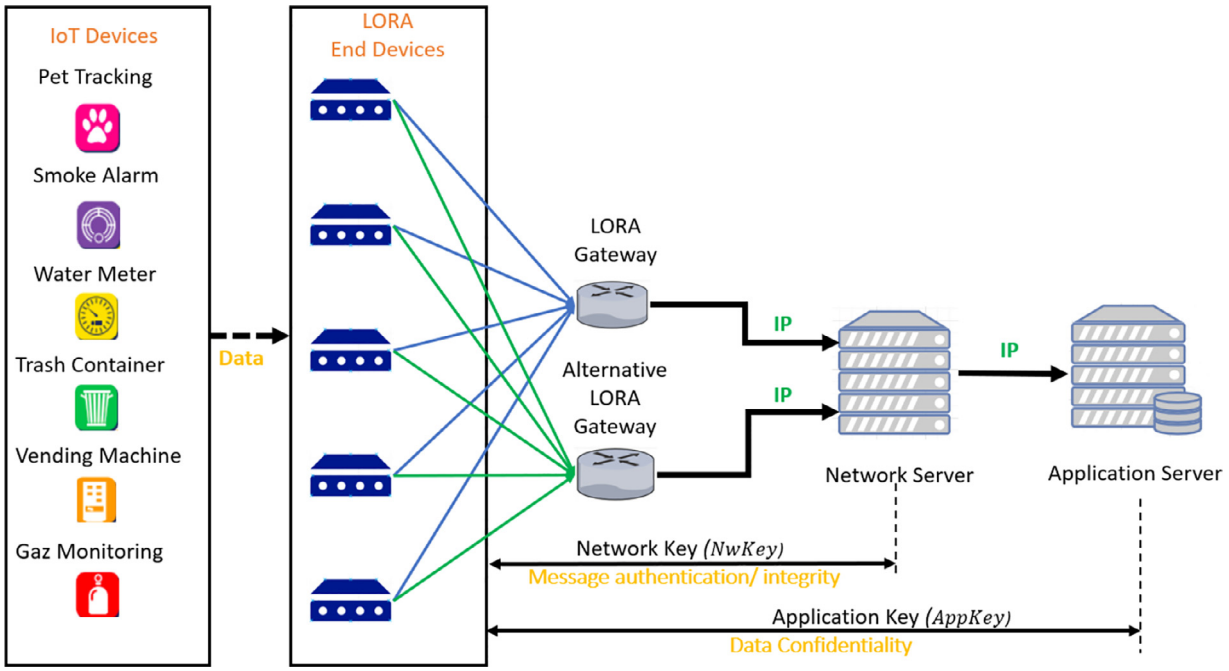


Fig. 1. LoRaWAN topology.

an application key *AppKey*. The NS knows all *NwKeys* of all EDs, while the application server knows all *AppKeys* of all EDs. However, the difference between the ABP and the OTAA activation modes is that in the ABP activation mode, all the secret keys and addresses are static and they are stored at the ED, and the EDs are directly connected to the NS without any request. While in the OTAA activation mode, the secret keys are dynamically generated.

1.1. Problem formulation

LoRaWAN suffers from some performance and security issues. Increasing the data rate while maintaining a high-security level is a major challenge and achieving the appropriate trade-off is not straightforward. In this paper, we review the different LoRaWAN security issues and explain the attack vector associated with the LoRaWAN vulnerabilities. On the other hand, we propose possible countermeasures to guard against the different security attacks.

1.2. Motivations

LoRaWAN exhibits security vulnerabilities that will be discussed in this paper. For example, the join procedure in the OTAA activation mode suffers from different security issues that may be exploited during the join-request and join-accept phases. Also, the ABP activation mode is plagued with its own security problems, especially that the keys are stored at the device level and hence, they can be compromised leading to attacks affecting the availability, authentication, confidentiality, integrity and the privacy of the LoRaWAN network.

1.3. Contributions

In this paper, we describe in details the LoRaWAN security vulnerabilities and the corresponding attacks. Also, we perform risk assessment of the different attacks, and we propose several security countermeasures that aim at enhancing the LoRaWAN security while maintaining a low overhead in terms of computation and communication. Moreover, we present most of the attacks related to LoRa devices and how these attacks could be mitigated.

1.4. Organization

The rest of this paper is organized as follows: in Section 2, we describe the LoRaWAN architecture with its different network layers as well as the different LoRa EDs classes, the supported operating systems, and the different applications. In Section 3, we perform a comparison between LoRaWAN and other similar technologies. In Section 4, the different security challenges facing LoRaWAN are presented. In Section 5, the security features of LoRa EDs activation modes are compared. Section 6 presents the different Cryptographic security features provided by LoRaWAN. In Section 7, the strength and

weakness of LoRaWAN security features are described in details. In Section 8, the LoRaWAN security attacks are reviewed along with the possible countermeasures. Section 9 discusses the new security features introduced by LoRaWAN v1.1. In Section 10, the risk assessment of LoRaWAN v1.1 is presented. In Section 11, we present our security recommendations. Finally, in Section 12, we conclude this paper.

2. LoRaWAN system: a comprehensive study

In this section, we describe the LoRaWAN architecture, the LoRaWAN network stack and its corresponding layers, and we present the different applications that can be enabled by LoRaWAN.

Note that Table 1 includes the notations that are frequently used in this paper.

2.1. LoRaWAN standards

LoRa was introduced as an LPWAN protocol that relies on the spread spectrum modulation techniques that derive from the Chirp Spread Spectrum (CSS) technology. In fact, the introduction of LoRa lacked the ability to define the upper networking layers. As a result, LoRaWAN, which is a cloud-based medium access control (MAC) layer protocol was introduced as a network layer (routing) protocol and a network's system architecture to manage the communication between LPWAN gateways and end-devices, communication frequencies, required powers/resources and data rates [2,3].

Moreover, to constantly and continuously enhance the performance and security of LoRaWAN, different versions were presented to ensure a much more secure, resilient, flexible and enhanced real-time long range communication for resource-constrained IoT devices. These versions are presented as follows:

2.1.1. LoRaWAN v1.0.1

LoRaWAN v1.0.1 was introduced (in February 2016) as a telecom technology to benefit both LoRa community and the IoT industry [4]. Its initial introduction ensured a secure and bi-directional communication, ensuring a close cooperation between Mobile Network Operators (MNO) and IoT domains [5]. However, LoRaWAN v1.0.1 is prone to many limitations and attacks which will be further discussed in this paper.

2.1.2. LoRaWAN v1.0.2

LoRaWAN v1.0.2 (introduced in July 2016) relies on using two distinct keys which are the network key *NwkSKey*, and the application key *AppSKey* [6], respectively, which are derived from a single root key [7]. *NwkSKey* is used to perform an integrity check, while the *AppSKey* is used to encrypt the payload until the application server [8]. However, LoRaWAN v1.0.2 is prone to various vulnerabilities (i.e reuse of frame counter/nonce values, and weak replay protection mechanism) and attacks (i.e replay, eavesdropping, ACK spoofing, DoS, etc), which are mitigated in the newest LoRaWAN v1.1.

2.1.3. LoRaWAN v1.1

LoRaWAN v1.1 was introduced (in October 2017) to overcome the main security vulnerabilities (weak key management, weak cryptography, weak authentication, etc.) and security attacks (eavesdropping, relay, ACK spoofing, packet modification, etc) that still lurked in the previous LoRaWAN versions (i.e v1.0.1 and 1.0.2) [9]. Thus, supporting handover roaming and enhancing security for battery-powered and resource-constrained IoT end-devices [10]. Among its specifications, this paper lists the following:

- **Several** *FCntDown***Counters:** are supported in the LoRaWAN v1.1, instead of being limited to a single *FCntDown* counter shared over all ports, in LoRaWAN v1.0.x (i.e 1.0.1 and 1.0.2).
- **Reset indication commands:** (i.e ResetInd, ResetConf) are only available to LoRaWAN v1.1 ABP devices. This MAC command is not available in LoRaWAN 1.0.x servers.
- **Rekey indication commands:** (i.e RekeyInd, RekeyConf) are only available to LoRaWAN v1.1 OTA devices. This MAC command is not available in LoRaWAN 1.0.x servers.
- **Device time commands:** (i.e DeviceTimeReq, DeviceTimeAns) are only available to LoRaWAN v1.1 activated devices. This MAC command is not available in LoRaWAN 1.0.x servers.
- **Application session key:** is only derived from the *AppKey* in LoRaWAN v1.1 network servers, whilst the device's *AppKey* is not used.
- ***NWKey* root key:** is responsible for deriving all the required keys for the provision of LoRaWAN v1.1 devices with the LoRaWAN v1.0.x back-end.
- **Device time request/answer command:** is used in LoRaWAN v1.1 instead of the traditional *BeaconTimingReq* & *BeaconTimingAns* MAC commands in previous LoRaWAN versions.

2.1.4. LoRaWAN v1.0.3

LoRaWAN v1.0.3 was introduced (in July 2018) as the newest version that outperforms and overcomes its predecessors (LoRaWAN v1.0.1 and LoRaWAN v1.0.2), whilst also overcome certain performance limitations found in LoRaWAN v1.1 [11]. For this reason, its various specifications [12] are presented as follows:

Table 1
Abbreviations and symbols.

ADR	Adaptive data rate
CCM	Counter with CMAC
CIA	Confidentiality, Integrity and Availability
CMAC or CBC-MAC	Cipher block chaining message authentication code
CTR	Counter
DoS	Denial of Service
DDoS	Distributed Denial of Service
IoT	Internet of Things
IV	Initialization vector
LoRa	Long-Range LPWAN technology
LoRaWAN	An application of LoRa
LPWAN	Low power wide area networks
MAC	Message Authentication Code
MIC	Message Integrity Code
Nonce	An arbitrary number that may only be used once
RFID	Radio Frequency Identification
ABP	Activation By Personalization
ACK	Acknowledgement
DL	Down-link
ED	End Device
GW	Gateway
MCS	Modulation and Coding Scheme
NS	Network Server
JS	Join Server
OTAA	Over-The-Air Activation
PHY	Physical Layer
SF	Spreading Factor
UL	Up-link
NwkKey	Network Key
NwkSKey	Network Session Key
AppKey	Application Key
AppSKey	Application Server Key
AS	Application Server
CSS	Chirp Spread Spectrum
DER	Data Extraction Rate
DevAddr	Device Address of ED
DevNonce	Device Nonce of ED
FCntDown	Frame Counter Down
FCntUp	Frame Counter Up
JoinNonce	Nonce of Joining server
MITM	Man-In-The-Middle attack
NFCntDwn	Network Frame Counter Down
RF	Radio Frequency
HMAC	Keyed-Hash Message Authentication Code
TLS	Transport Layer Security
NetID	Network Identifier
RxDelay	Delay between RX and TX
Join-request	Join request to attach the ED to the LoRa network
AppEUI	Application Identifier
DevEUI	Device Identifier
AES128—CMAC(K, M)	AES 128 uses CMAC operation mode with secret key K and a message M
CFList	Optional list for channel frequencies
MHDR	MAC Header
MType	Message Type
FHDR	Frame Header
FPort	Frame Port
FRMPayload	MAC Frame Payload Encryption
LoRaWAN-EN	LoRaWAN Enterprise Network
FNwk_SIntKey	Forwarding Network Session Integrity Key
SNwk_SIntKey	Serving Network Session Integrity Key
Nwk_SEncKey	Network Session Encryption Key
RF	Radio Frequency

- **Device time commands:** (i.e DeviceTimeReq, DeviceTimeAns) is a MAC command that is only available and activated on LoRaWAN v1.0.3 and not on LoRaWAN v1.0.1, nor LoRaWAN v1.0.2.
- **Beacon timing request/answer commands:** are also MAC commands that are only used LoRaWAN v1.0.3 version, with another command (*DeviceTimeReq* & *DeviceTimeAns*) which can be used by the device as a substitute.
- **Unicast/Multicast Support:** is added in LoRaWAN v1.0.3 to support class B devices.

Table 2
Differences between LoRaWAN classes: A, B, and C.

Class Type A	Class Type B	Class type C
Battery powered sensors	Battery powered actuators	Main powered actuators
Low latency	Low latency	No latency
Bidirectional with 1 UL+ 2DL Slot	Bidirectional with scheduled DL slots	Bidirectional with most of the time in listening mode
Unicast messages	Unicast and multicast messages	Unicast and multicast messages
Small payloads, long intervals	Small payloads, long intervals, periodic bacon from GW	Small payloads
ED initiates communication (UL)	Extra receive window	Server can initiate transmission at any time
Server communicates with ED (DL) during predetermined response windows	Server can initiate transmission at fixed intervals	ED is constantly receiving

- **Device time request command:** this MAC command was also introduced in LoRaWAN v1.0.3 as a new time synchronisation feature for class A and C devices.
- **Compatibility:** class B devices in LoRaWAN v1.0.3 are also compatible with LoRaWAN v1.1 class B devices.

2.2. LoRaWAN architecture

In this section, we describe the main components of the LoRaWAN architecture such as LoRa ED's, GW's, in addition to the network and application servers.

2.2.1. End-devices

The LoRa ED's are the devices that send data to the LoRa network. These devices could be sensors that measure pressure, velocity, humidity, temperature, vibration, etc. LoRa ED's can be divided into three classes: A, B, and C, based on the *MAC* layer operation [13–15].

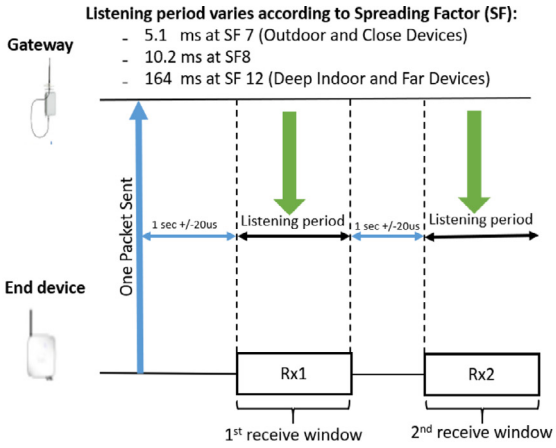
- **Class A:** The EDs in this class transmit an Up-Link (UL) message and then open 2 Down-Link (DL) receive windows, as illustrated in Fig. 2-a. The server can respond in one of the two opened receive windows. The operational mode of this class is the "most power-efficient mode".
- **Class B:** In this class, the EDs are bi-directional with scheduled receive slots as shown in Fig. 2-b. The ED opens an extra receive window at scheduled times, in addition to the two receive windows opened in Class A mode. Thus, the GW sends a time synchronization beacon to the ED to open a receive window at the scheduled time slot. This process permits the server to check if the ED is listening or not.
- **Class C:** The EDs in this class, as shown in Fig. 2-c, have almost continuous receive windows, which requires a high amount of energy compared to the other classes. However, the latency concerned with the data transmission is minimal between the NS and the ED.

Table 2 highlights the difference among the various LoRaWAN classes.
Currently, a variety of LoRa EDs exists. In the below, we list some of the most known LoRa EDs:

- **LoPy**: it is a triple bearer Micro-Python. In some cases, it functions as a single channel GW. It is python-based and works under the frequencies of 868 MHz, and 915 MHz. Moreover, it is a LoRa Wi-Fi/BLE ED [16].
- **Dragino Lora GPS HAT**: it is an extension module for the Raspberry Pi. It works in the frequencies of 868 MHZ, 433 MHZ, and 915 MHZ. It has a GPS that can switch "between the internal patch antenna and the external active antenna". It has low power consumption. It works under the environmental operational temperature which ranges between −40 °C and 85 °C [17].
- **RN2483**: it is a LoRa microchip with low power consumption. Its RF Communication Bit Rate (CBR) is up to 10,937 bps. It works in the environmental operational temperature which is between −40 °C and 85 °C [14].
- **Waspmote**: it is a hardware ED that works with extremely low power consumption. It is C/C++ based and it works under the frequencies of 868 MHz and 915 MHz. It works under the environmental operational temperature which is between −30 °C and 70 °C [18].
- **mDot**: it works under the frequencies of 868 MHz and 915 MHz and under the environmental operational temperature which is between −40 °C and 85 °C [19].
- **Adafruit Feather 32u4**: it is a 433 MHz radio module used for either 868MHz or 915MHz transmission/reception bands [20].

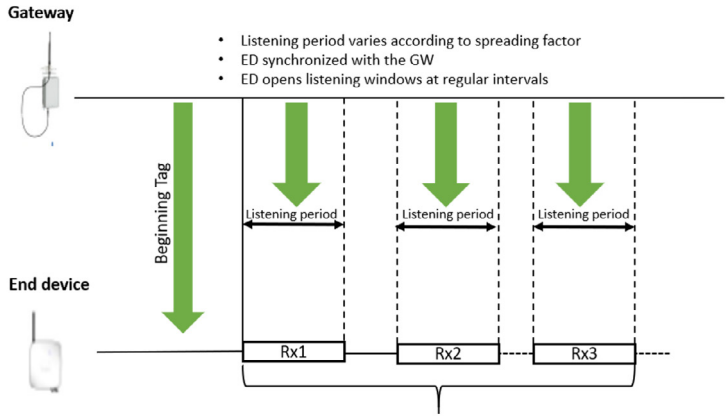
2.2.2. LoRa gateways

The LoRa GW is a bridge between the EDs and the LoRa network. Being the interface with a set of heterogeneous EDs, LoRa GWs support many communication protocols and can operate across all the layers of the OSI system. In some cases, the GW might be connected to a firewall in order to guard against well-known security attacks. Hence, in LoRa, the GW is responsible for connecting the EDs to the network server using the LoRaWAN protocols. They present the second layer in the LoRaWAN architecture. There are 3 types of LoRa GWs that are described next.

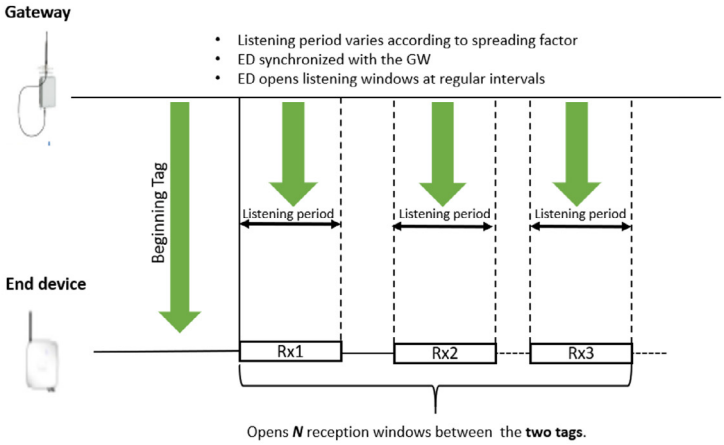


After **sending a packet**, ED opens **2 windows** for **DL reception** for acknowledgments, Mac commands, application commands, etc.

(a) Class A



(b) Class B



(c) Class C

Fig. 2. LoRaWAN class A (a), class B (b) and class C (c).

- **iC880A-SPI**: this GW receives packets from different nodes. It can handle communication on up to 8 channels. It may use a dynamic rate adaptation. As the distance between the nodes and the GW increases, a high spreader factor is used, which leads to a lower data rate. This GW presents a robust architecture with immunity against interference. Additionally, this GW has the ability to decode multiple Spread Factors (SF) [21].
- **Kerlink IoT**: this GW exhibits a high performance and it is highly reliable. It has a capacity of 8 frequency bands. It may contain on board GPS. It can receive packets from different nodes at the same time, which means that this GW can handle communication on up to 8 channels. Additionally, it may decode multiple SF [22].
- **WISE-3610**: it is a new GW technology which supports AC1200 dual-band Wi-Fi. It supports up to 500 nodes and a mini-PCIe module for 3G/LTE card [23].
- **Dragino LG01-S GW**: it is an open source single channel LoRa GW. It connects the LoRaWAN wireless network to an IP network through Wi-Fi or any other mobile technology (e.g. 3G). This GW can be used to send data over a long distance range with a low data rate. Hence, this GW can be adapted to different applications [24].
- **Cisco Wireless Gateway**: it presents a new GW technology designed by Cisco. The features of this gateway can be summarized as follows [14]:
  - Support of all LoRaWAN classes
  - Support of adaptive data rates
  - Channel diversity
  - Integrated GPS
  - Time synchronization
  - Support of geolocation
  - Frequency band: 863–870 MHz for Europe, Middle East, Africa and India, 902 - 928 MHz for Americas, Asia and Pacific
  - Support of 4G/LTE backhaul

2.2.3. Network server

The LoRa NS has the role of managing the LoRa network. The NS receives the up-link frames that are forwarded by the GWs, and sends them to the corresponding AS. It plays a role in handling the MAC layer component and in scheduling the down-link data transmission. Also, it eliminates any packet duplication, and serves in scheduling the acknowledgments, and in adapting the data rates. Moreover, it ensures a high level of security by applying encryption at several layers such as: the encryption by the *NwKey* at the network layer, and the encryption by the *AppKey* at the application level, and the encryption at ED level using the device key [25].

2.2.4. Join server

The JS has the role of joining the EDs to the network and authenticating them. Hence, the two join server keys are the *JSIntKey* and the *JSEncKey*. The *JSIntKey* is used for the "*MIC* of the rejoin request message and the join accept answer", while the *JSEncKey* is used in encrypting the join accept message [25].

2.2.5. Application servers

The ED communicates with the AS through a code or program running on the ED. The AS functions as a remote entity that controls the EDs actions, and collects all the information about the connected EDs [25].

2.3. LoRaWAN network stack

The LoRaWAN protocol stack is divided into four layers as shown in Fig. 3. These layers present different frames and headers as shown in Fig. 4 [26]. These layers are listed and described in the following [14,27,28]:

2.3.1. RF layer

The signals are transmitted by the LoRaWAN modules and devices at the RF layer. The radio interface defines the signal power, its frequency, its modulation waveform, and its supported RF protocol. Hence, the most important frequencies that are being used in different regions are as follows: in Europe: 868 MHz and 433 MHz, in North America: 915 MHz, and in Asia: 433 MHz [29].

2.3.2. Physical layer

This layer is used to define the modulation scheme and the corresponding Time Symbol (TS), as shown in Fig. 5. The structure of the frame at the physical layer is as follows [30]:

LoRaWAN messages have a physical *PHY* header which contains a preamble, the payload header, the data, in addition to *CRC* [27]. The introduced preamble defines the packet modulation scheme and its duration is 12.25 *TS*. The *PHY* header and the *CRC* header have together a size of 20-bits and they are encoded with a reliable code rate. In addition, the payload at the *PHY* layer is encoded with the selected code rate.

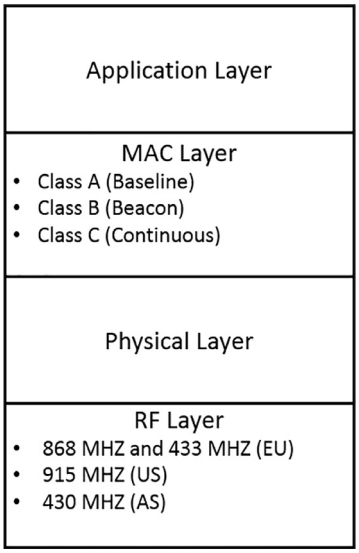


Fig. 3. LoRaWAN protocol stack.

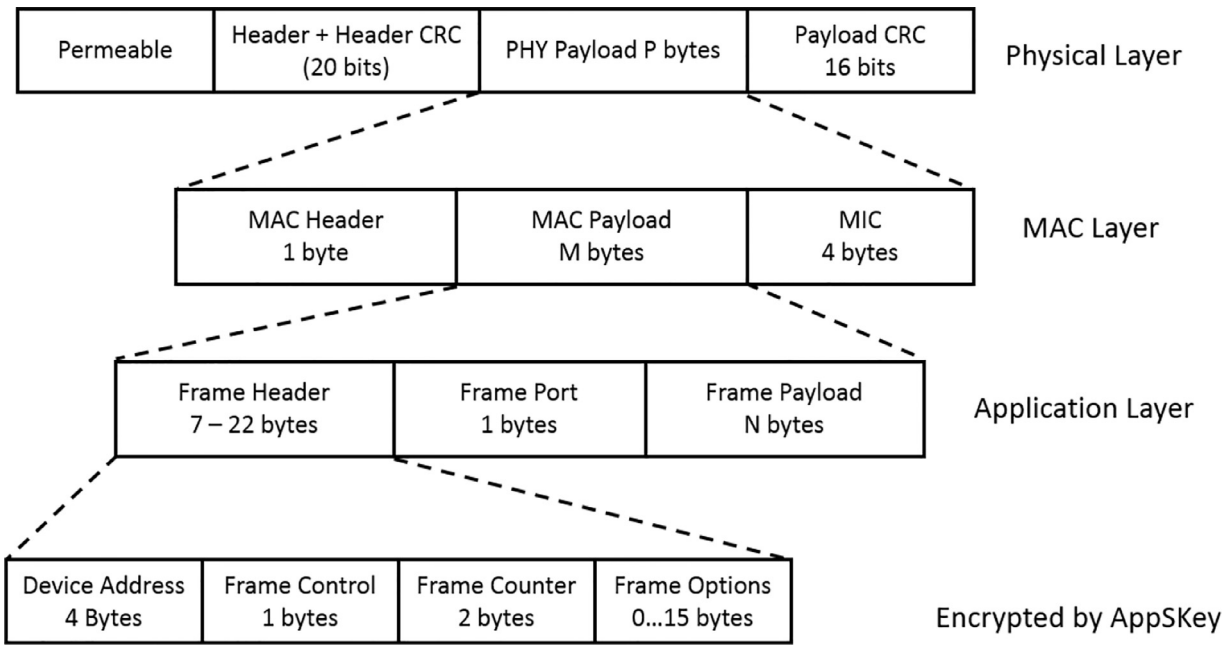


Fig. 4. LoRaWAN frame structure.

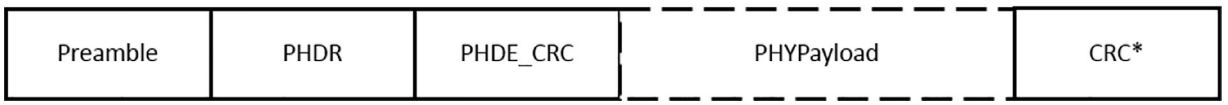


Fig. 5. Physical layer frame.

Fig. 6. MAC layer frame.

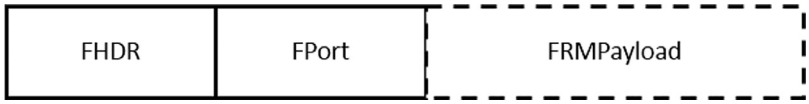


Fig. 7. Application layer frame.

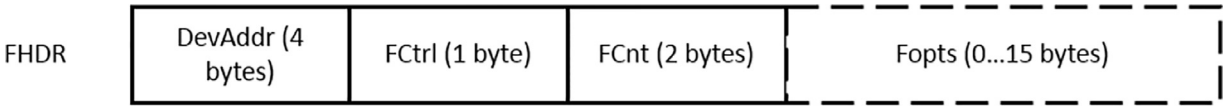


Fig. 8. Frame header.

2.3.3. MAC layer

The MAC frame, as shown in Fig. 6, is divided into three blocks as described in [27,30]:

- ***MAC* header** : it specifies the type of the message and the major frame format version that is related to the encoded LoRaWAN Layer.
- ***MAC* payload** : contains user's data, in addition to the application header information (e.g. the message type and LoRaWAN version information). Also, it contains The frame header *FHDR*, the frame port *FPort*, and the *FRMPayload*.
- ***MIC*** : the *MIC* value is used to validate the MAC header and payload integrity, and authentication towards preventing any unauthorized message modification. The *NwkSKey* is used as a message authentication secret session key, while the data confidentiality of the MAC payload is ensured by using the *AppSKey* with the AES-128 algorithm.

Note that in a join procedure, the *MAC* payload is changed, and the join-accept and the join-request messages are used. On the other hand, in LoRaWAN, the *CRC* at the physical layer is used to detect any channel error in the received messages. Meanwhile, the *MIC* at the MAC layer is employed to validate the data integrity and source authentication of the received messages.

2.3.4. Application layer

The application frame header, as illustrated in Fig. 7, contains the following fields:

- Frame header *FHDR*: contains the 4 elements, as shown in Fig. 8, that can be described as follows:
  - The device address *DevAddr*: it consists of 32 bits that identify the ED. Hence, the *DevAddrs* is assigned by the NS to the ED.
  - The frame control *FCtrl*: it contains 1 byte used for controlling the network information such as the data rates used for up-link transmission, and the message acknowledgment.
  - the frame counter *FCnt*: this counter keeps track of the count of the up-link and down-link data messages sent/received by the NW.
  - The frame options *FOpts*: it transports the MAC commands, which are found on the "piggybacked into data frames".
- Frame Port *FPort*: it is determined depending on the application type. The frame port is used to determine if the frame contains MAC commands alone (when it is set to 0) or application specific data. It is used to determine if the *FRMPayload* contains any *MAC* command alone (set to 0) or any application-specific data. The value of the *FPort* ranges from 1 to 223, while the 244 port number is dedicated to LoRaWAN MAC layer test protocol.
- Frame payload: if this frame contains a payload, then the *FRMPayload* must be encrypted before calculating the *MIC* using AES-128.

In Fig. 9, the different layers frames and headers are illustrated with the fields sizes.

2.4. LoRaWAN applications

The LoRaWAN technology enables many IoT low-data-rate applications [31–33]. The IoT applications that might benefit from LoRaWAN are listed and described in the following:

- **Smart Homes and Buildings:** The LoRaWAN technology can be used in smart homes, given that it supports a wide range of wireless communications. Consequently, the home sensors can reliably be used to track the assets, detect fire, optimize the building security by means of smart door locks, manage water overflow, manage energy consumption, and predict maintenance [34].
- **Smart Cities:** In smart cities, the need for high range transmissions is key and as such, LoRaWAN, being cost and power efficient, is a good candidate technology [35]. LoRaWAN can enable several smart cities applications like smart street lighting, smart parking, smart water leakage detection, smart waste management, and smart bus schedule signs.
- **Smart Health-care:** LoRaWAN technology can be used for many smart health-care applications, and more specifically for monitoring applications [36]. This is due to its low cost, its power efficiency, and its good performance at long distances.

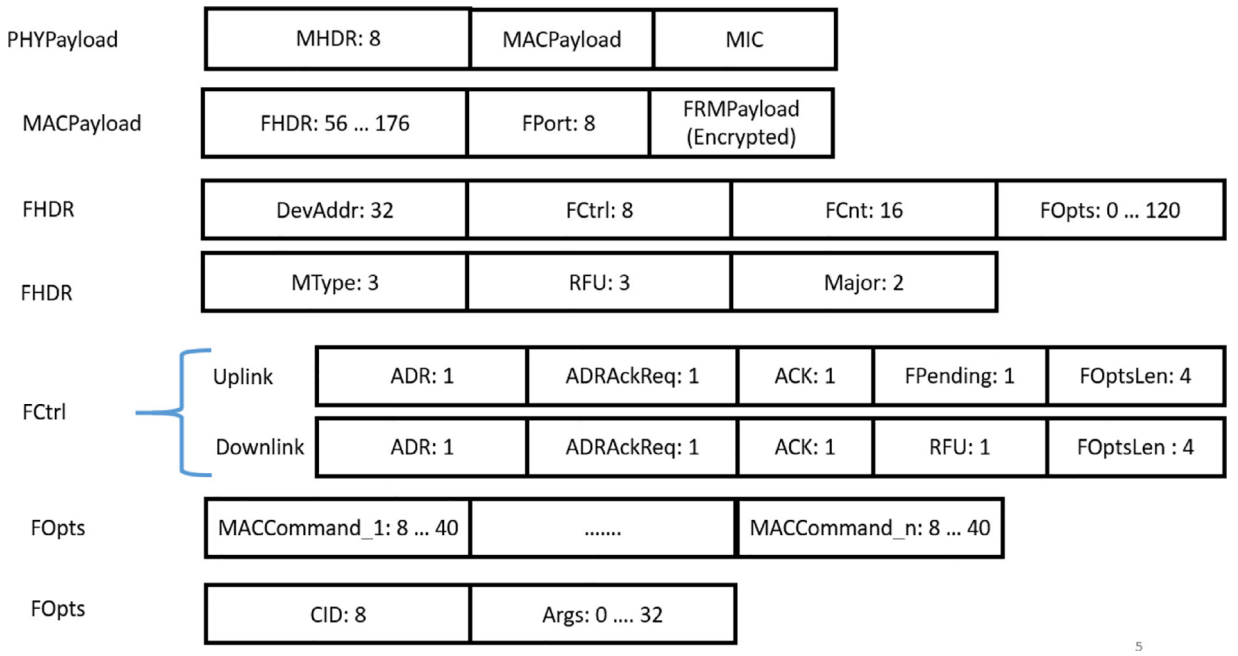


Fig. 9. LoRaWAN frame format (field sizes are in bit).

This technology can be used for tracking Alzheimer patients, locating the dementia patients, smart wearable for kids and seniors, smart medical fridge, and detecting people's falls [37].

- **Smart Agriculture System:** In the smart agriculture domain, LoRaWAN can be used for measuring the environmental conditions via low-cost sensors that send data from the farms to the cloud for enhancing the management operations. Hence, this can be used in smart cattle ranching, ingestible cattle health tracker, soil moisture monitoring, and smart soil sensors [38].
- **Smart Transportation System:** LoRaWAN can help also in enabling smart transportation because of its low power consumption, whilst the GW can be mounted on a bus or a train. This GW, connected to the cloud, can collect traffic data for analytical purposes. This plays an essential role in enhancing traffic management by avoiding traffic congestion and minimizing traveling time. Hence, collecting information about the traffic can be done for future planning, advanced traveler information system, advanced vehicle control system, and advanced commercial vehicles operations system [39].
- **Smart Parking:** Nowadays, most people use the car as the main transportation tool which results in parking management issues. Thus, the integration of LoRaWAN in smart parking helps in alleviating this issue by detecting free areas. The smart parking sensor technology is used for data collection and has a low battery consumption that can last for 10 years.
- **Smart Metering:** Smart metering aims at measuring and recording the electrical power consumption. The recorded data can be used for power consumption monitoring and billing. Also, this application presents many advantages such as making the bills more accurate, monitoring the nodes, GWs, and clouds.
- **Smart Factory:** The smart factory is based on using acoustic emission sensors that enable many applications including monitoring the factory safety, inspecting the production, ensuring the quality of the production, ensuring a non-stoppable production, etc. A smart factory offers many benefits such as low maintenance cost, managing the production remotely, and optimizing the production level.
- **Drones/UAVs:** LoRaWAN can also be used in the drone/Unmanned Aerial Vehicle (UAV) domain [40] as an intelligent transportation [41] and communication system [42] (i.e UAV to Everything (U-2-X)) [43], especially as a secondary telemetry communication system for drone delivery [44] and as a long distance communication with low power [45].
- **Cyber-Physical Systems:** LoRaWAN was also deployed in different cyber-physical systems and domains alike [46] to sort power consumption especially in real-time resource constrained devices [47], especially in the indoor/outdoor Industrial IoT domain [48], and part of Industry 4.0 [49].

Finally, we summarize all the LoRaWAN Characteristics in Fig. 10.

3. LoRaWAN versus other technologies

There are many LoPWAN technologies that are mainly used in IoT. Table 3 shows the difference between LoRaWAN and other technologies.

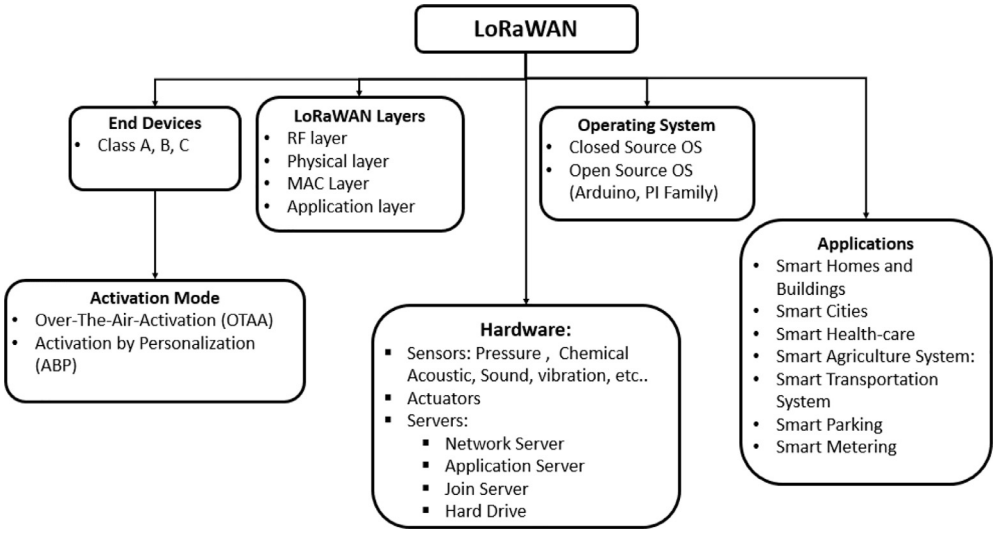


Fig. 10. LoRaWAN characteristics.

Table 3
LoRaWAN vs other technologies [50,51].

Features	LoRaWAN	SIGFOX	LTE-Cat 1	LTE-M	NB - LTE
Modulation	SS chip	UNB/GFSK/BPSK	OFDMA	OFDMA	OFDMA
Rx bandwidth	500–125 KHz	100 Hz	20 MHz	20–1.4 MHz	200 KHz
Data rate	290bps → 50Kbps	100 bit/sec 12/8 bytes Max	10 Mbit/sec	200 kbps to 1 Mbps	Average 20K bit/sec
Max. # Msgs/day	Unlimited	UL: 140 msgs/day	Unlimited	Unlimited	Unlimited
Max output power	20 dBm	20 dBm	23 - 46 dBm	23/30 dBm	20 dBm
Link budget	154 dB	151 dB	130 dB+	146 dB	150 dB
Battery lifetime - 2000 mAh	105 months	90 months 18		months	
Power efficiency	Very high	Very High	Low	Medium	Medium to high
Interference immunity	Very high	Low	Medium	Medium	Low
Coexistence	Yes	No	Yes	Yes	No
Security	Yes	No	Yes	Yes	Yes
Mobility/localization	Yes	Limited mobility, no localization	Mobility	Mobility	Limited mobility, no localization

In the following, we compare LoRaWAN against two competitive LoPWAN technologies, NB-IoT and DASH7.

3.1. NB-IoT

NB-IoT (Narrow Band IoT) is a LoPWAN technology mainly used by 3GPP cellular systems. NB-IoT is characterized with very low complexity and high throughput. This technology has been integrated also into the LTE system with some enhancements such as extending the battery lifetime, and reducing the device cost. This optimization was intended also to ensure better performance by introducing new measurements related to channel quality. However, NB-IoT uses the same frequency as LTE and uses the QPSK modulation [25].

3.2. DASH7

It is a new open source technology, which compiles with ISO/IEC under the frequency band 433 MHZ. it consists of 7 OSI layers known as the DASH7A protocol. It has a range of 2 KM with low latency and low battery consumption [25].

In Table 4, we compare the topology of LoRaWAN, DASH7, and NB-IoT. We can conclude that adding the haystack into LoRaWAN technology can solve certain limitations [25,52] towards making LoRaWAN more efficient and better secure.

However, with the rise of a new technology, such as the Wireless Smart Utility Network (Wi-SUN-IEEE 802.15.4 g standard) and its quick adoption into the IoT domain, due to its network support of star and mesh topologies, along hybrid star/mesh deployments, proved to be challenging for LoRaWAN especially in terms of high data rates and low latency. As a result, a quick comparison has been made and summarized in the following table Table 5 between the new leading technologies such as the NB-IoT (Low Power WAN (LPAN)) [53–55], LoRaWAN [56,57] and Wi-SUN [58–60].

Table 4
LoRaWAN vs DASH7 and NB-IoT [25].

Specification	LoRaWAN technology support	DASH7 technology support	NB-IoT
Standard	LoRa alliance	ISO/IEC 18000-7	3gpp (release 2015)
Operational frequencies	Unlicensed ISM band 868, 915 MHz	Unlicensed ISM band 433.92, 868, 915 MHz	Licensed same as LTE bands
Data rate (kbps)	0.3–50 (Europe) 0.9 -100 (US)	13, 55, 200 (16, 8, 4 channels)	50
Identity header size	4 bytes	2–8 bytes	like LTE
Payload size	51–222 bytes	256 bytes (Max)	UL: 125 bytes DL: 85 bytes
Addressing	UL: Broadcast and DL: Unicast	Unicast - Broadcast and multicast - Anycast	UL: Unicast DL: Unicast and broadcast
Topology	Star	Star	Star

Table 5
NB-IoT, LoRaWAN and Wi-SUN Comparison.

Characteristics	Standards		
Type	**NB-IoT**	**LoRaWAN**	**Wi-SUN**
Acronyms	**Narrow-Band IoT**	**Long range WAN**	**Wireless smart ubiquitous network**
Range	**Less than 10 km**	**15 km**	**Less than 4 Km**
Technology	**5G**	**Gateway**	**4G**
Latency	**1.5–10 secs**	**1–16 secs**	**0.02–10 secs**
Data rate	**10–26 Kbps**	**27–63 kbps**	**50–300 Kbps**
Coverage	**Wide Area Network**	**Wide Area Network**	**Field Area Network**
Modulation	**UNB/GSK/BPSK**	**Chirp SS**	**FSK/OQPSK/OFDM**
Application	**IoT domain**	**IoT/Non-IoT domain**	**IoT/Non-IoT domain**
Security	**Low**	**High**	**High**
Cost effective	**Yes**	**Yes**	**Yes**
Complex	**No**	**No**	**Depends on hardware**

4. LoRaWAN challenges

In practice, LoRaWAN suffers from different challenges and limitations as described in [61–64] and discussed below:

- **Compatibility:** The connectivity between the heterogeneous devices, which are fabricated by different manufacturers, is one of the major challenges that are facing LoRaWAN. Hence, one essential goal in the IoT domain is to ensure the compatibility among the different devices across the network.
- **Efficient energy management:** The energy consumption is another challenge facing the LoRa devices since most of these devices (class A and B) are battery powered. Replacing the battery can be challenging when the device is not easily reachable, in addition to the associated cost since the battery cost is sometimes higher than the device's cost. Thus, the solution has to be efficient in a way to increase the battery lifetime.
- **Complexity:** This is another challenge that faces LoRaWAN. For example, adding a new communication capability to the LoRaWAN architecture is challenging and involves tedious tasks. As such, an abstraction layer is needed to make it easier to integrate new features into the LoRaWAN architecture.
- **Fast-paced development:** The IoT paradigm is rapidly evolving with continuous development. Every day, new technologies and devices are connected to such a network, which can result in many unknown issues. Hence, keeping up-to-date with the LoRaWAN technology serves the purpose of having it widely adopted as an IoT framework.
- **Security:** There are some security issues within LoRaWAN, which makes it vulnerable to several attacks such as replay attack, bit-flipping attack, etc. Hence, developing lightweight security solutions is key for LoRaWAN tiny devices that are constrained in terms of power, processing, and storage resources. However, achieving a high security level comes at the cost of increasing the required processing time and power and consequently an overhead in terms of latency and computation.
- **Resilience to physical attacks and natural disasters:** The IoT devices are small in size and may or may not support a physical protection layer. Thus, the movable devices can be easily stolen and the fixed ones can be destroyed. On the other hand, these devices should also be protected against any natural disaster that may occur [63].
- **Privacy protection:** IoT devices might contain sensitive information that must be protected. However, many threats are still compromising private IoT data, and thus efficient data security schemes need to be adopted [63].

.

5. LoRaWAN security features

In this section, the LoRaWAN security features are listed and described. The two types of EDs activation modes are described first, followed by LoRaWAN key management and then, the message authentication-encryption scheme (CCM operation mode, see Fig. 16).

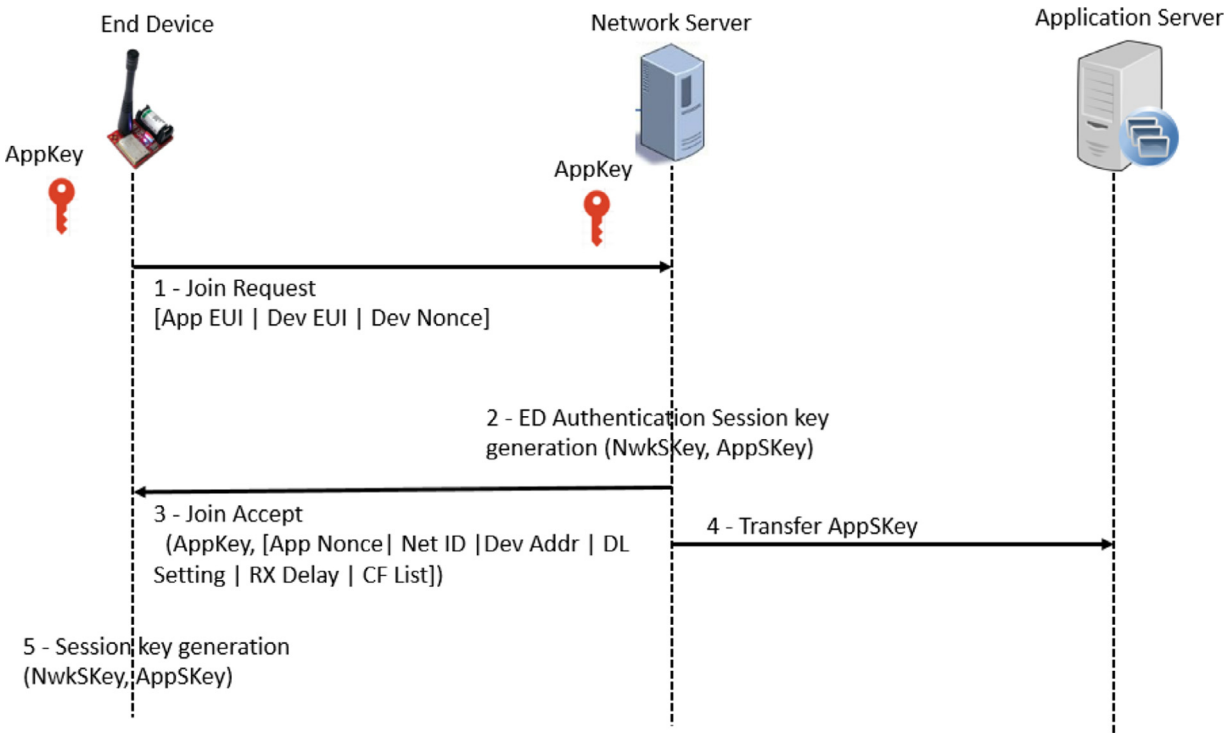


Fig. 11. OTAA: Over The Air Activation.

5.1. LoRaWAN end-devices activation modes

A node can connect to the LoRaWAN network by one of the two activation modes, which are: OTAA and ABP. The message is encrypted and signed by using *NwkSKey* and *AppSKey*, which are known by the NS and the ED. However, the generation of the session keys differs between these two activation modes [27,65–67].

5.1.1. Over-The-Air-Activation (OTAA)

The Over-the-Air Activation consists of "Join-request" and "Join-accept" messages between an ED and a server. First, the ED starts the join procedure by sending a join-request message to the NS. This message includes the *DevEUI*, *AppEUI*, and *DevNonce*. The *AppKey*, which is an AES-128 root key, is provided to the NS and the EDs as illustrated in Fig. 11 [67,68].

The *AppEUI* represents the application identifier, and *DevEUI* is the global unique device identifier. The *DevNonce*, which is a random sequence number starting by 0 when the ED is turned on, and it gets incremented with every ED join-request. Hence, the *DevNonce* value must never be reused for a given *AppEUI* value. The *MIC* value is calculated by the ED, and the *AppKey* is shared between the ED and the NS. Note that the join-request message is not encrypted, however, it is signed with the 4 bytes MIC according to the following equation:

$$mac = AES_{cmac,\ 128}(AppKey, MHDR||AppEUI||DevEUI||DevNonce), \tag{1}$$

where $MIC = mac[0 \ldots 3]$.

The join-request message is sent to the NS, which checks if the ED is authenticated by checking the *MIC*, using *DevEUI* (unique for each device) and *AppEUI* (unique for each person who owns this device). Then, it forwards it to the corresponding AS. Hence, if the message is rejected, then no response will be received by the ED, and the process will be terminated by the NS, and the join process will fail. If the process is accepted, then a join-accept message will be sent from the NS to the ED. The join-accept message consists of 3 bytes: 1) The *AppNonce*, which is a unique random identifier generated by NS; 2) The *DevAddr*, which is the device address assigned by the NS to the ED; and 3) The *NetID*, which is a network identifier used in separating the addresses of different geographic LoRaWAN networks. Hence, the *DLSetting* is related to the down-link configuration values. The *RxDelay* refers to the delay between the 2 processes, the transmission and reception processes. The *CFList* is related to the channel frequencies. The message is signed and encrypted using the *AppKey*. Note that 2 keys are generated by the ED during the join-accept message, which are the *NwkSKey* and the *AppSKey*, as shown below:

5.1.2. Activation by Personalization (ABP)

In the ABP activation mode, there is no join procedure. The ED does not have the *DevEUI*, *AppEUI*, and *AppKey*, which are used in the join procedure. Thus, in ABP, the four session keys, the $FNwk_SIntKey$, $SNwk_SIntKey$, $Nwk_SEncKey$ and App_SKey

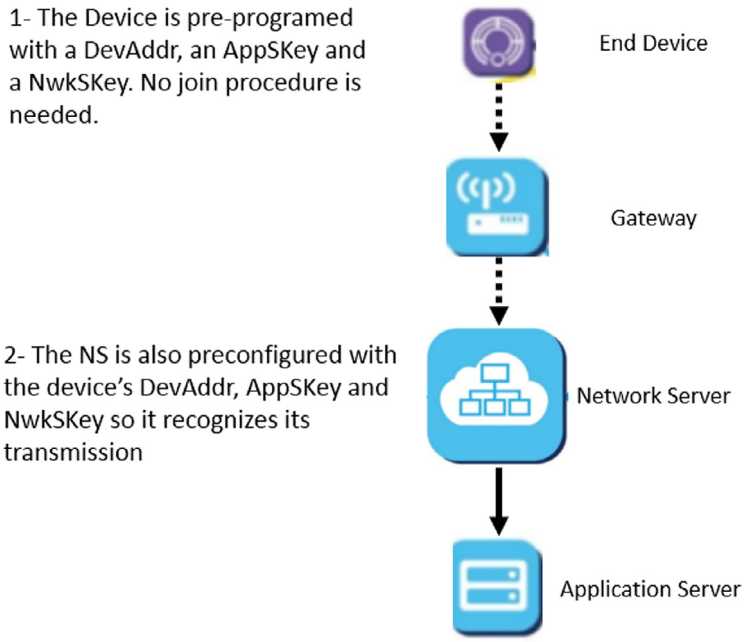


Fig. 12. ABP: Activation By Personalization.

are stored inside the ED. Each ED should have its unique preset keys, $FNwk_SIntKey$, $SNwk_SIntKey$, $Nwk_SEncKey$ and App_SKey. Thus, if any ED is compromised, it should not affect other EDs. The ED transmits the *ResetIndMAC* command in *FOpt* for all the up-link messages when it accesses the network for the first time, and continues until a *Resetconfcommand* is received from the network. Thus, when the ED is reset, it should use the predefined default configuration. Then, the ED will directly communicate with the server by sending the message that is signed and encrypted as shown in Fig. 12.

5.1.3. Difference between the OTAA and ABP

The OTAA is more secure than ABP, due to the dynamic creation of the session keys *NwkSKey* and *AppSKey*. The keys in ABP are predefined and stored inside each ED and NS. In ABP, there is no need for join-request and join-accept messages, which allows the message to be sent directly to the NS.

5.2. LoRaWAN key management

There are 3 essential keys in the LoRaWAN security scheme, the *AppKey* (root key) and the two session keys, *NwkSKey* and *AppSKey*. In the following, we describe the generation and exchange of these keys.

5.2.1. LoRaWAN key generation

The *AppKey* is a 16-byte unique key, which is allocated by the application owner to the EDs. In OTAA, as shown in Fig. 13, the 2 session keys are generated by the *AppKey* using the *AppNonce* related to the NS, and the *DevNonce* related to the ED [27,69]. The session keys are re-generated each time the ED rejoins the network or when it is reset. However, in ABP, the 2 session keys are unique and stored at the ED before any data transmission and they will not be changed if the ED is being reset [27,70,71].

$$NwkSKey = AES128_{Encr}(AppKey, 001||AppNonce||NetID||DevNonce||pad_{16}) \tag{2}$$

$$AppSKey = AES128_{Encr}(AppKey, 002||AppNonce||NetID||DevNonce||pad_{16}) \tag{3}$$

5.2.2. LoRaWAN key exchange

The ED sends a join-request message that contains the *DevNonce* to the NS. The NS checks if the ED is allowed or denied to join the network, and then sends a join-accept message to the ED that includes a new *AppNonce*. Both the ED and the NS will generate the session keys using the *AppNonce* and the *DevNonce*. Hence, the keys are not transmitted over the air and only the 2 nonces are transmitted. Moreover, the encrypted *AppKey* in addition to *DevNonce* and *AppNonce* will also be generated [27,69]. Fig. 14 shows how the session keys are exchanged in OTAA.

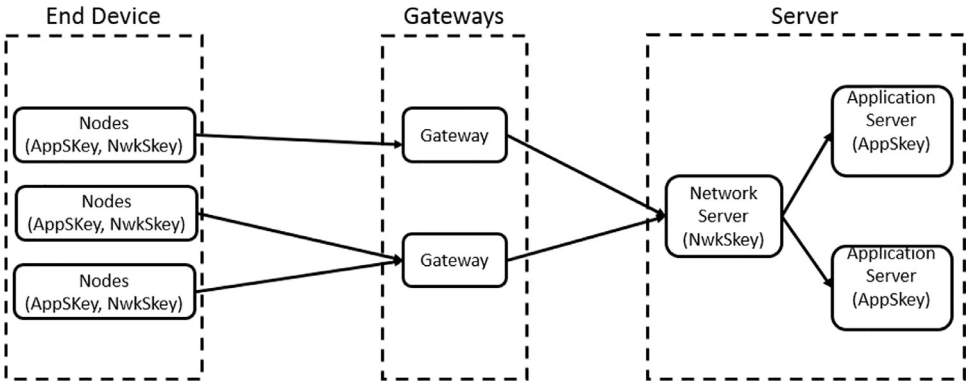


Fig. 13. Usage of session keys in LoRaWAN.

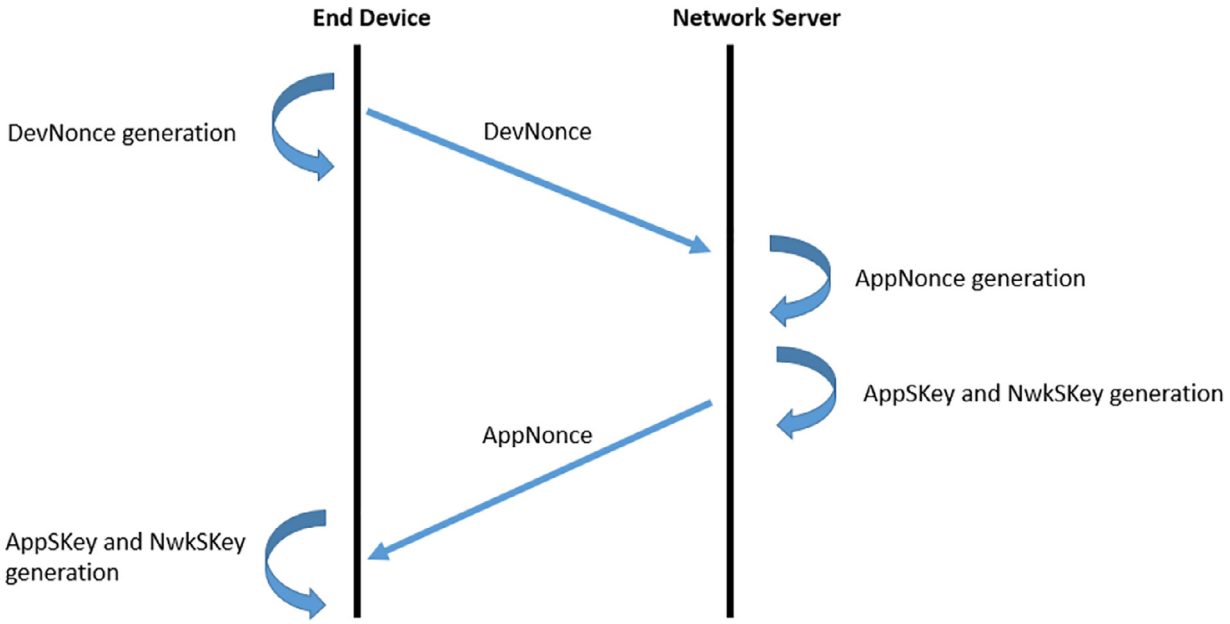


Fig. 14. Exchange of session key in OTAA.

5.3. LoRaWAN data confidentiality

In LoRaWAN, the data is encrypted using AES-128 in *CTR* mode. However, the *NwkSKeys* will be used when the *FPort* packet is set to 0, and otherwise, the *AppSKey* will be used. Hence, in all LoRaWAN sessions, the sent and received counters (*FCntUp* and *FCntDown*) are not repeated, and they are provided by the EDs and the NS. Algorithm 2, shown below, describes

Algorithm 2 LoRaWAN Authentication Encryption Algorithm (LAEA).

procedure LAEA(FRMPayload, K)
  $nb \leftarrow \lceil len(FRMPayload)/16 \rceil$
  for $i \leftarrow 1 \text{to} nb$ **do**
    $B_i \leftarrow getBlock(FRMPayload, i)$
    $A_i \leftarrow (0x01||(0x00 * 4)||Dir||DevAddr||FCntUp\ or\ FCntDown||0x00||i)$
    $S_i \leftarrow AES128_{Encr}(K,\ Ai)$
    $C_i \leftarrow S_i \oplus B_i$
  $C \leftarrow C_1||C_2||\ldots||C_k$

the generation of the encryption and decryption keys [27,69]. At the end, the *FRMPayload* is XOR-ed with the key-stream to encrypt or decrypt the data, and *FCNTUp* and *FPort* are sent unencrypted [27].

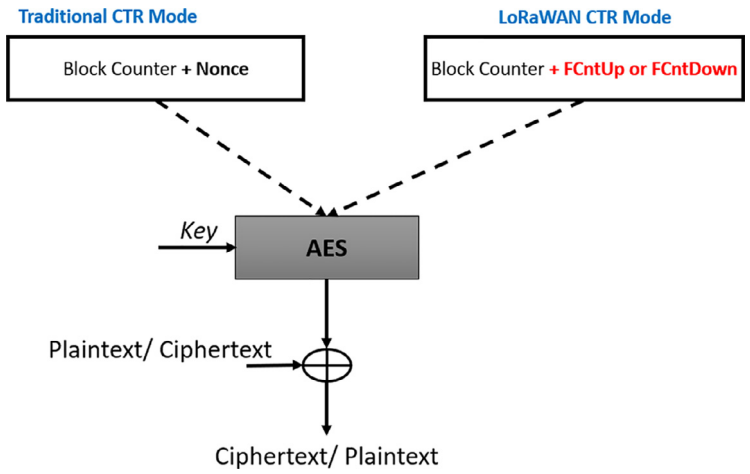


Fig. 15. LoRaWAN cipher mode vs traditional counter mode.

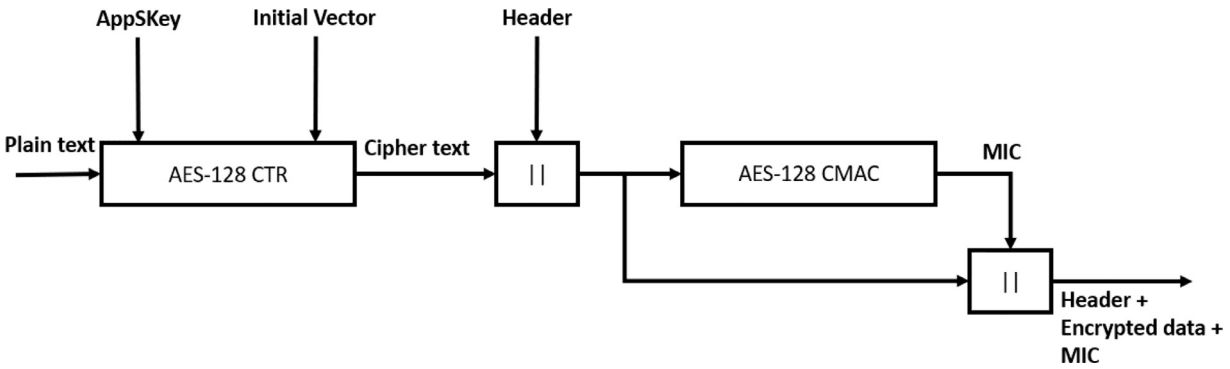


Fig. 16. LoRaWAN authentication-encryption process.

Fig. 15 shows that the *CTR* mode and the LoRaWAN cipher mode are similar, except for some details. In *CTR* mode, the block counter contains a nonce. In LoRaWAN, there is an *FCntUp* and an *FCntDown*, which are the message counters that are increased with each message. However, if the *FCntUp* and the *FCntDown* are not repetitive, then both modes are similar [69].

5.4. Securing key management

Recently, different security solutions were presented by different authors to ensure a secure key management scheme for LoRaWAN. In [72], Han & Wang reviewed LoRaWAN 1.1 security whilst LoRaWAN security was enhanced by presenting a root key update scheme that makes the LoRaWAN security keys' cryptoanalysis more difficult and strengthen the security of session key derivation. The experimental results indicate that this scheme generates a high degree of randomness with the highest computing efficiency compared to hash-based KDF. In [73], Sanchez-Iborra et al. evaluated LoRaWAN's security vulnerabilities and presented different alternative schemes mainly the Ephemeral Diffie-Hellman Over COSE (EDHOC) due to its flexibility in updating session keys, low computational cost and limited message exchange, and the Static Context Header Compression (SCHC) algorithm to enable the IPv6 communication between the LoRaWAN's end-nodes and external networks. In [74], Donmez et al. presented assisted mode as a key management scheme to improve LoRaWAN's security and address to its end-devices vulnerability. Assisted mode delegates the management of lifetime keys to a master device instead of end-devices. This allows the use of cheaper and less-complex medical end-devices. In [75], Xing et al. presented an improved secure key management scheme for LoRa communication system to ensure a remote distribution and update of the static key. A Hierarchical Deterministic (HD) wallet for key management was used by the device and server, whilst ensuring a key agreement between them using an Elliptic Curve Diffie-Hellman (ECDH) key exchange algorithm. Experimental results indicate the achievement of trade-off between security, computational cost, and communication overhead. In [76], Chen et al. examined the LoRaWAN v1.1 security framework, and presented a new key generation scheme for LoRaWAN based on Rabbit cipher. This scheme seems to be faster than the original Rabbit algorithm and the AES which is recommended by LoRaWAN v1.1. The experimental results show that this scheme achieves a high computing efficiency and also meets the security key randomness requirement. In fact, further comparison can be seen in Table 6.

Table 6
Comparison between different key management solutions.

Year	Authors	Reference	Solution	Description
2018	Han & Wang	[72]	Root key update scheme	High randomness degree
2018	Sanchez-Iborra et al.	[73]	Lightweight & authenticated key management	EDHOC update session keys & SCHC for IPv6
2019	Donmez et al.	[74]	Delegated key management	Master device manages lifetime keys
2019	Xing et al.	[75]	Improved secure key management	HD wallet for key management, ECDH for key agreement
2019	Chen et al.	[76]	Fast session key generation	High computing efficiency & key randomness

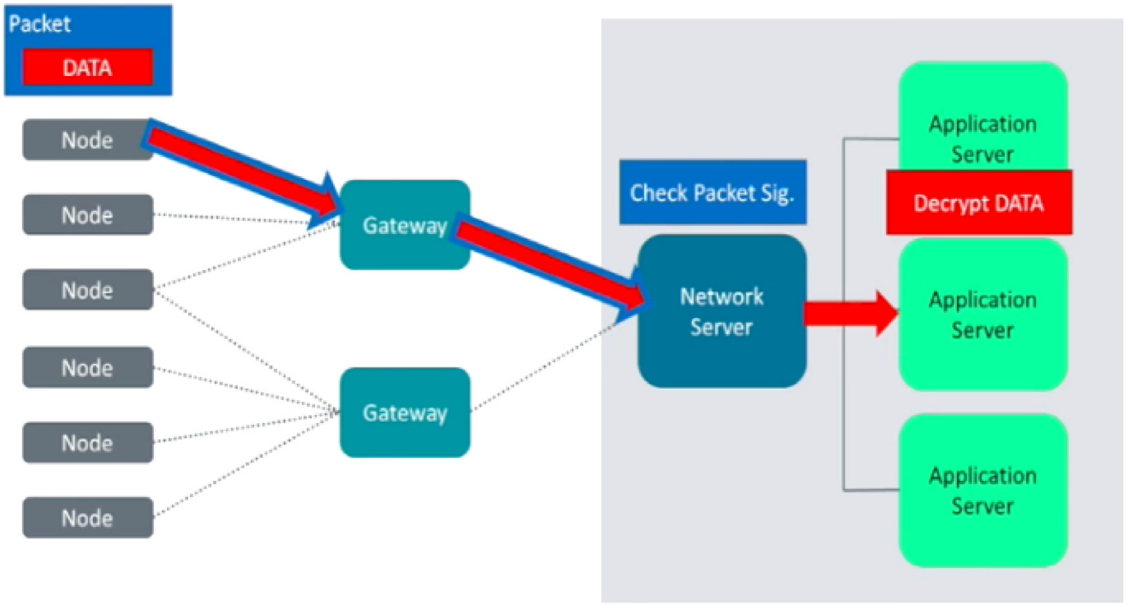


Fig. 17. Join message signing.

5.5. LoRaWAN data integrity and source authentication

MIC is used for supporting the message integrity check. At this phase, the *MAC* payload of the messages, which are sent and/or received, are signed to prevent the manipulation of the cipher text, *DevAddr, FCntUp, FCntDown*, etc. The *MIC* calculation is illustrated in Algorithm 3 [27,69]:

Algorithm 3 LoRaWAN Message Signing (LMS).

procedure LMS
  $msg \leftarrow MHDR||FHDR||FPort||FRMPayload$
  $B0 \leftarrow (0x49||4*0x00||Dir||DevAddr||FCntUporFCntDown||0x00||len(msg))$
  $MAC \leftarrow AES128_cmac(NwkSKey, B0||msg)$
  $MIC \leftarrow mac[0 \rightarrow 3]$

The join-request and accept messages are signed, as shown in Fig. 17, and a *MIC* is generated for each one. The *MIC* is obtained from the computed *MAC* and it represents the four bytes of *MAC*. For the join-request message, the *MAC* is produced as follows:

$$MAC = AES_{cmac,\ 128}(AppKey, MHDR||AppEUI||DevEUI||DevNonce) \quad (4)$$

While, for the join-accept message, *MAC* is derived as below:

$$MAC = AES_{cmac,\ 128}(AppKey, MHDR||AppNonce||NetID||DevAddr||DLSettings||RxDelay||CFList) \quad (5)$$

At the network server, the packet signature is checked as shown in Fig. 17.

5.6. MAC-then-encrypt or encrypt-then-MAC

The choice as to which operation (MAC or encryption) comes first depends on the required security level. In LoRaWAN, we can distinguish between two types of messages: the join-accept message, and the normal data messages. Hence, the

MAC-then-encrypt approach is applied to the join-accept message. However, for the data messages, the encrypt-then-MAC approach is adopted. The difference between the two approaches is the confirmation of the cipher-text's integrity. If we sign-then-encrypt, the cipher text's integrity will not be ensured since the attacker can modify the cipher-text without being noticed. However, the encrypt-then-MAC approach will ensure the cipher-text's integrity since any modification of the cipher-text will be detected at the receiver when checking the *MAC*.

6. LoRaWAN cryptographic features

Some of the cryptographic features that are supported by LoRaWAN are listed below [27,77,78]:

6.1. Security session context

In LoRaWAN, the session context is made up of the application and the network sessions; the network session is established by either the NS or the ED and a lot of information is exchanged in the network session such as the *NetSKeys*, *Framecounters*, and *DevAddr*. However, the application session is established between the ED and the AS and information such as the *AppSKey* and Frame counters are exchanged. Thus, the session keys will not change in a given session. Instead, only the frames counters will be incremented and they will not be reused in other sessions. If the old context (after processing the ED request) is dropped by the NS, this will trigger the context switch (such as reply attack, join request, etc...). As a result, there will be an issue in the communication between the ED and the NS.

6.2. Use of optimized AES implementation

LoRaWAN uses the AES 128-bit for encryption with the *CCM**, the *CMAC* and the *ECB* modes. Thus, using AES with the *CCM* mode will ensure the authentication and the confidentiality of the message. The *CCM* ensures the integrity and the confidentiality of the message. However, LoRaWAN uses the *CMAC* mode for authentication followed by encryption. Hence, the *CCM** mode cannot be used in LoRaWAN for join messages. During the join-accept message, the *MIC* is encrypted along with the message. Thus, the ED checks the received message for integrity.

6.3. Security keys and key derivation

In OTAA activation mode, the key derivation occurs at the ED and the JS. The keys are dynamic since they are based on the following nonces: *DevNonce, JoinNonce, RJcount*0, and the *RJcount*1, which change for each session. However, in the ABP activation mode, the keys are static and stored in the EDs during the manufacturing process. Thus, the ED will use any of the session keys due to the lack of the join procedure.

6.4. Counters and nonces

These parameters are used for defending against the replay attack. When talking about a session counter, we should keep in mind that a security context switch will be added and it generates new session keys; the old session counter will be reset. Hence, the frame counter that consists of the *FCntUp*, the *NFCntDown*, and the *AFCntDown*, helps in tracking the up-link and down-link messages. The *FCntUp* is used for counting the up-link data messages and it is incremented with every up-link message. The *NFCntDown* is used for counting the down-link data messages carrying the *MAC* commands. The *AFCntDown* is used for counting the down-link data messages carrying the application data. However, the *FCntUp* is sent in the FCnt up-link header.

The *NFCntDown* and *AFCntDown* are sent in the *FCnt* down-link header and the network element will synchronize the *FCnt* up-link with the value received. Also, the receiving ED synchronizes the corresponding local counter with the received value. Thus, if a message is authenticated, the ED synchronization occurs and the local counter will be synchronized with the received value. However, if the message is not authenticated, the synchronization will not occur and the local counter will not be modified. As mentioned above, in OTAA, the counters are reset at every session while in ABP, the counters are not reset. Moreover, the *DevNonce* and *JoinNonce* counters are static.

The *RJcount*0 and *RJcount*1 counters are 16-bit counters. The *RJcount*0 is incremented with every join-request, and it is reset to 0 each time a Join-accept is successfully processed. Moreover, *RJcount*0 must be different for each session, whereas the *RJcount*1 is a fixed counter and remains for the ED lifetime; it will never be reset and the counter will increase with every join-request message.

7. Strengths and weaknesses of LoRaWAN security features

From a security perspective, LoRaWAN exhibits many strengths and some weaknesses as described in the following subsections [79].

7.1. LoRaWAN security strengths & enhancements

LoRaWAN is usually used with devices having hardware constraints such as sensors that send data to the LoRaWAN network forwarding to the corresponding AS. The LoRaWAN technology presents many advantages such as the ease of use, the cost efficiency, and the high level of security. Some of the important security features that LoRaWAN provides are listed as below:

1. **OTAA-provisioning:** In this phase, a dynamic communication is established between the ED and the NS for the negation of the keys and the certificates at each session. In this activation mode, the session keys are generated during the join procedure, which reduces the risk of attacks.
2. **Dynamically activated devices use the application key (*AppKey*):** In OTAA mode, the *AppKey* is used for generating the 2 session keys, the *AppSKey* and the *NwkSKey*. Initially, the session keys are set to the default *AppKey*, which is used during the join procedure for activating all the devices on the network. Hence, it is important to perceive a new way to customize *AppKey* for every device.
3. **Having a secure hardware element:** the security credentials are stored at the device level. Moreover, by using a well-secured hardware, the security level against physical attacks will be enhanced. Consequently, the possibilities of extracting the secret key will be prevented (e.g. reverse engineering).
4. **To prevent replay attacks:** activating the up-link and down-link counters on the NS will help in defending against replay attacks. Note that an attacker will not have the ability to decrypt a message due to the session keys that are generated in a dynamic way. If a replay attack occurs, the MIC will fail to check the message integrity and the attack will be detected: when activating a device, the 2 frame counters, *FCntUp* and *FCntDown*, will start at 0. For every up-link message, the *FCntUp* will be increased by 1, and for every down-link message, the *FCntDown* will be increased by 1. Thus, any message will be denied if the ED or the NS received a transmitted message with any frame counter value equals to or less than the previous one.

7.2. LoRaWAN security weaknesses

LoRaWAN technology has many advantages, but it presents also several security vulnerabilities, which are common to many other wireless technologies. Some of the weaknesses that are facing the LoRaWAN technologies are:

1. The encrypted message and the secret key have the same length.
2. If an attack occurs and the session keys are compromised, it would be very hard to change the AES keys across all devices or nodes.
3. LoRaWAN technology is based on 3 keys: the *AppKey*, the *AppSKey* and the *NwkSKey* that an attacker would try to compromise to disclose the message content. Hence, there are risks 1) when transmitting the message, 2) during the key management cycle, 3) during the generation of keys at every session, and 4) when storing these keys at the device level. Thus, accessing the device keys will maximize the attacker's chances to succeed in compromising the other keys. In this context, side channel attacks aim at accessing such keys from stored memory data. Besides, any attack related to the NS key storage will affect the entire system since the attacker can intercept the messages through many techniques.
4. Identification and connection issues: In fact, the GW sends periodically a beacon to the NS. Consequently, the attacker can employ these beacons to perform several attacks. For example, if the attacker compromises the GW, then, he can send the beacons at a higher rate than the original one. However, some manufacturers have their own approaches to defend against this attack, such as the "retro-fitting security elements", but this solution has several drawbacks related to operational cost, time, and complexity.

7.3. Security requirements:

Before discussing the threats associated with LoRaWAN, we list first the security requirements of IoT applications [63]:

1. **Confidentiality**: protecting the data by preventing unauthorized users from accessing private information.
2. **Integrity**: preventing unauthorized users from modifying the data that is being exchanged among IoT devices.
3. **Availability**: ensuring that the resources and information are available when requested by authorized users. In IoT, new attacks are performed on the physical layer to compromise the availability of IoT devices (e.g. depletion-of-battery attack).
4. **Authenticity**: ensuring that the transferred data is genuine by authenticating the parties involved in the transmission.

Typically, the attacker goals are:

1. compromising the network security properties (i.e. availability, confidentiality, authenticity, etc.),
2. compromising the network security assets (i.e. keys, nonces, counters, etc.).

It is important to note that securing the LoRaWAN components is mandatory since it affects its wide adoption, in addition to defending against the possibility of compromising these devices to initiate other types of attacks (e.g.Mirai attack) [6].

Table 7
LoRaWAN security elements classification.

Element	Primary or secondary asset	Confidentiality	Integrity
NwkSKey	Primary	Yes	Yes
AppSKey	Primary	Yes	Yes
AppKey	Secondary	Yes	Yes
DevNonce	Secondary	No	Yes
AppNonce	Secondary	Yes	Yes
FrmPayload	Primary	Yes	Yes
DevAddr	Secondary	No	Yes
Fcnt	Secondary	No	Yes
ACK	Secondary	No	Yes
MAC commands	Secondary	Yes	Yes

Table 7 summarizes the LoRaWAN security elements in terms of confidentiality and integrity. For example, the confidentiality of *AppNonce* is guaranteed while it is not for the *DevNonce*. These nonces (*AppNonce* and *DevNonce*) are used to generate the session keys. We can see in the table that the integrity of all elements is protected due to the LoRaWAN integrity method.

Next, we describe the security elements that are listed in the table. As an example, in Table 7, if a primary security element such as the *NwkSKey* is compromised, then the whole LoRaWAN network is exposed since the key can be used to decrypt the communicated messages. However, if the attacker compromises a secondary key such as the *AppKey*, he will need other elements to decrypt the messages.

1. *NwkSKey*: it is used by the NS to check the signature of the message, and it is generated during the ED activation. Hence, if the confidentiality of the *NwkSKey* is compromised, then the attacker can use it to send unauthorized messages, which will pass the signature checking at the NS. However, if the *NwkSKey* integrity is compromised either at the ED or the NS, then all the communication sessions will be compromised, but not the signature checking at the NS.
2. *AppSkey*: it is used by the AS to decrypt the messages and it is generated during the ED activation. Hence, if the confidentiality of the *AppSKey* is compromised, then the attacker will have the ability to decrypt all the messages, and thus the confidentiality of the whole system will be compromised. However, if the integrity of the *AppSKey* is compromised, then the EDs and the AS will lose the ability to decrypt the messages and the attack is detected.
3. *AppKey*: it is used in the ED activation for deriving both session keys, *AppSKey* and *NwkSKey*. In fact, the *AppKey* is allocated at the ED and the NS before the activation process. Hence, compromising the confidentiality of *AppKey* allows the attacker to perform the join-request reply attack, allowing malicious devices to join the network. However, if the integrity is compromised, the devices are unable to join the network when OTAA is used.
4. *DevNonce*: this nonce is generated at the ED. During the OTAA activation, it will be transmitted to the GW and to the NS, and then both nonces (*DevNonce* and *AppNonce*) are encrypted using the *AppKey* to generate the 2 session keys. The *DevNonce* is transmitted in plain text and an attacker cannot make use of it without the *AppKey*. However, it is important to protect the integrity of *DevNonce* since it is used to derive other keys.
5. *AppNonce*: it is used to generate session keys with *DevNonce* and *AppKey*. Hence, the confidentiality, as shown in the table above, is protected, and thus identifying the *AppKey* by the attacker can lead to compromise the whole network by computing the session keys (*NwkSKey* and *AppSkey*). However, if the integrity of the *AppNonce* is compromised, the generated session keys are modified leading to an invalid communication session.
6. *FrmPayload*: it contains sensitive data that will be transmitted to the NS. Hence, if its confidentiality is compromised, the attacker breaches the user privacy. However, if its integrity is compromised, the transmitted data will not be trusted.
7. *DevAddr*: it is the ED identifier and it is transmitted in plain text. Hence, compromising its integrity interrupts the communication between the EDs and the NS.
8. *Fcnt*: it represents the counter value in both the NS and the EDs and it is transmitted in plain text. Hence, compromising its integrity leads to de-synchron-ization between the ED and the NS. Thus, if the counter is modified, this will result in a replay attack.
9. ACK: it is used for acknowledging received messages and it is transmitted in plain text. Hence, compromising its integrity will lead to disabling the ACK functionality.
10. MAC commands: it can be sent in *FOpts* or in *FrmPayload*, and some commands are confidential such as the radio settings management commands. Otherwise, if the confidentiality of the MAC commands is compromised, the attacker can get access to sensitive information such as the radio transmission parameters. Also, the integrity of the MAC commands must be well protected.

7.4. LoRaWAN security vulnerabilities and threats:

LoRa EDs are increasingly being deployed for different IoT applications, which were described in Section 2.4. However, attackers are trying to find and exploit any possible vulnerability in these EDs, in addition to the GW, NS, and AS. Recent research studies concerning the LoRaWAN vulnerabilities are discussed in the following sections. According to the reviewed

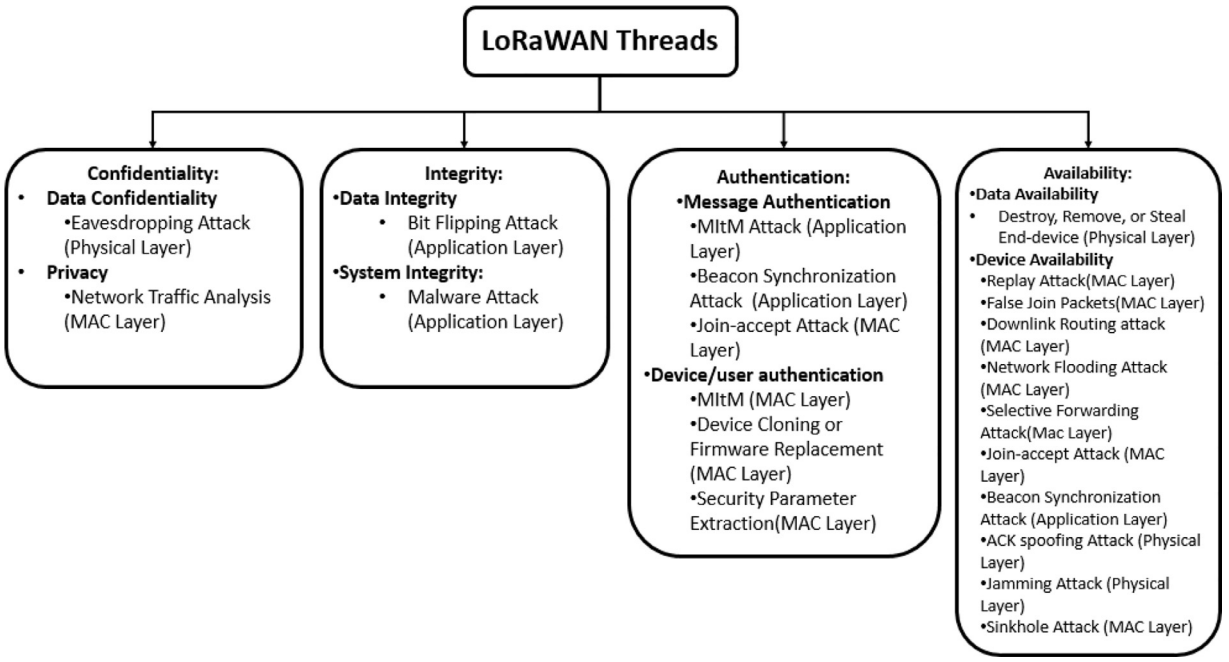


Fig. 18. LoRaWAN vulnerability.

work, the EDs suffer from availability, authentication, and integrity attacks in the OTAA activation mode. However, in the ABP activation mode, confidentiality and replay attacks (availability threats) are added to the OTAA threats, as illustrated in Fig. 18.

7.5. Discussion

LoRaWAN 1.1 was released in order to enhance the LoRaWAN security by fixing the existing vulnerabilities, while supporting backward compatibility with older LoRaWAN versions. The changes affect all the LoRaWAN components, including the ED, NS, GW, and AS. This version aims at enhancing the communication between the ED and the GW, and between the GW and the NS. In addition, it introduces countermeasures to known attacks. For example, in LoRaWAN 1.1, the *FCntUp* is encrypted and a confirmation message is added in such a way that the ED can identify easily the confirmed message sent by the NS. Moreover, the new LoRaWAN technology separates the NS, JS, and the AS and adds new Keys such as *NwkKey, NwkSEncKey, SNwkSIntKey, FNwkSIntKey*, and *AppSKey*. As such, the new keys, related to handover roaming, have an important role in enhancing LoRaWAN security.

Also, in LoRaWAN 1.1, the JS is managed by a third party and not by the NS. Hence, the introduced enhancements can be summarized as follows: all the keys that are related to the ED are managed by the JS, the NS is responsible for message transmission, and the AS is responsible of manipulating the message which prevents the network operator from reading the message content.

8. Existing LoRaWAN attacks and proposed countermeasures

In this section, we present the different kinds of threats and vulnerabilities in LoRaWAN. In addition, we propose new countermeasures to prevent some of the existing vulnerabilities that are not considered in the literature.

8.1. Authentication attacks

In this section, we present the LoRaWAN authentication attacks, along with proposed countermeasures.

8.1.1. Man-in-the-middle attack

The scenario of MITM attack between the ED and the GW [80] is described below and illustrated in Fig. 19:

1. The device sends a message
2. The attacker intercepts it
3. The attacker changes the payload
4. Using the compromised *NwkSKey*, the attacker signs the message with a valid MIC

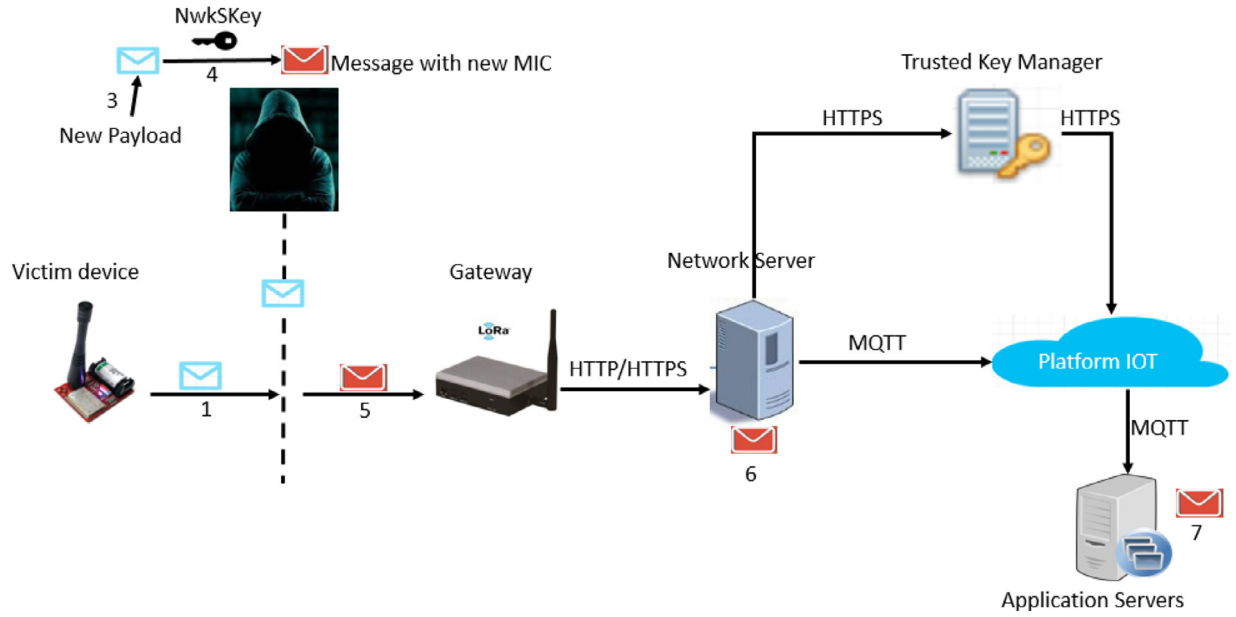


Fig. 19. Man in the middle attack technique.

5. The attacker sends the modified message to the GW
6. The NS checks the message's MIC with the *NwkSKey* and validates it before forwarding it to the IoT platform
7. The AS receives the modified message.

Taking into consideration that the attacker knows the *AppKey*, and after compromising the *AppSKey*, the attacker captures the join-procedure messages (join-request and join-accept) and computes the *NwkSKey*, which can be obtained by:

$$NwkSKey = AES128_{Encr}(AppKey, 0x01||AppNonce||NetID||DevNonce||pad16) \quad (6)$$

Once the attacker obtains the *NwkSKey*, he uses it to conduct the MITM attack. The attacker intercepts the messages, which are communicated between the ED and the NS. Consequently, he can modify the communicated messages or inject new ones. In fact, this is done before the integrity checking phase. However, the *NwkSKey* will be used for generating a new MIC for the new and modified messages. Hence, this type of attacks can be done either between the ED and the GW, or between the GW and the NS. Fig. 19 shows how this type of attacks occurs [65,80,81].

To prevent this attack, we propose to relate the *MIC* with the obtained $h(AppKey)$ in a non-linear and non-invertible manner. The proposed solution consists of sending $h(MICXORh(AppKey))$ instead of $h(AppKey)$. The illegal GW cannot modify and masquerade the legal device since the MIC is related to $h(AppKey)$ in a non-invertible manner. Indeed, any modification will result in a different MIC that cannot be isolated from $h(AppKey)$. Therefore, conducting this attack will become very difficult. The NS uses the stored $h(AppKey)$ and the *MIC* from the "join-request" message to validate the ED request.

8.1.2. Security parameter extraction & device cloning or firmware replacement

In this case, the attacker has a direct physical access to the ED. The attacker can either steal the reused keys or change them by intruding the ED firmware. In the case of security parameter extraction, the authenticity of the sent messages is affected. However, in the case of device cloning or firmware replacement, the authenticity, confidentiality and the integrity of the keys are compromised. Thus, the ED should be well protected against any attack related to the firmware [82].

8.2. Availability attacks

In this part, a set of LoRaWAN availability attacks are presented and analyzed. In addition, a set of possible countermeasures are proposed towards preventing these attacks.

8.2.1. Sinkhole attacks

In this type of attacks, the main goal of a malicious node is to direct the network traffic to a specific node. Thus, the attacker will promote a specific route in the network and infect the network nodes to utilize this route as shown in Fig. 20. This attack can be harmful when combined with other attacks, and it affects the system's availability. Using an IDPS can detect and prevent such attacks [63].

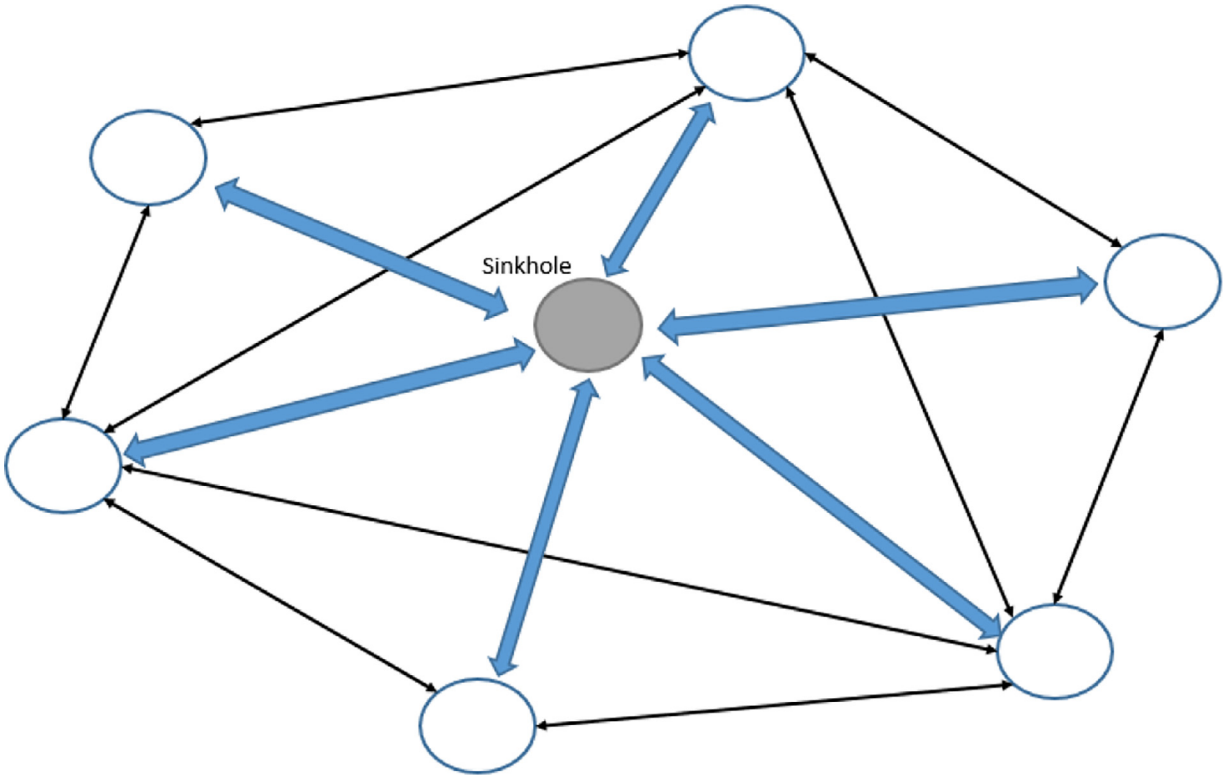


Fig. 20. Sinkhole attack topology.

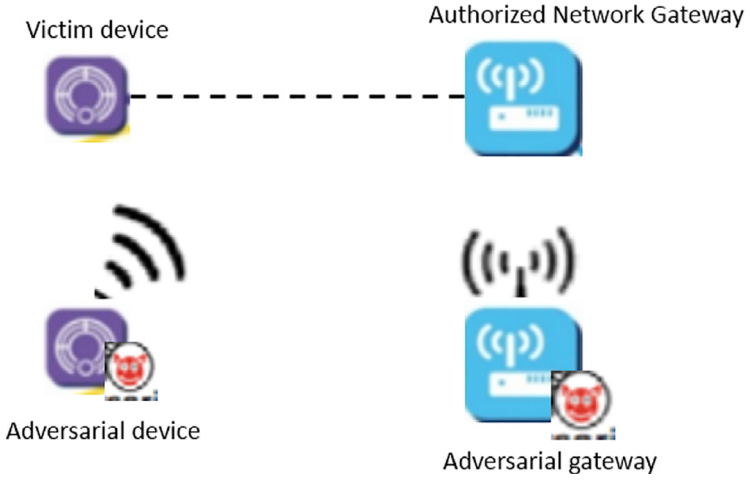


Fig. 21. Replay attack topology [69].

8.2.2. Replay attack

As shown in Figs. 21 and 22, the attacker uses the selective RF jamming technique to jam the OTAA join procedure signals. First, an ED sends a join-request message to the NS (join-request 1) with the *DevNonce*1. In this case, the attacker captured the join-request message 1 and jams the channel. This will prevent the ED from receiving the join-accept message. After waiting for the timeout of the joint-accept message, the ED will try to join the network again by sending a join-request message 2 with a *DevNonce*2. Similarly, the attacker jams this message and replays the joint-request 1 message. Checking the *DevNonce*1, the NS will accept the join-request since it has not been used before and the NS, JS, and the ED are de-synchronized with relation to the *DevNonce* parameter. However, this type of attacks is specific to LoRaWAN since there is a limited communication quota. Hence, every ED can daily transmit a maximum number of packets which is 14 packets consisting of 12-Byte payload. However, in ABP activation mode, a static key built into the ED is used [83]. Since the key is

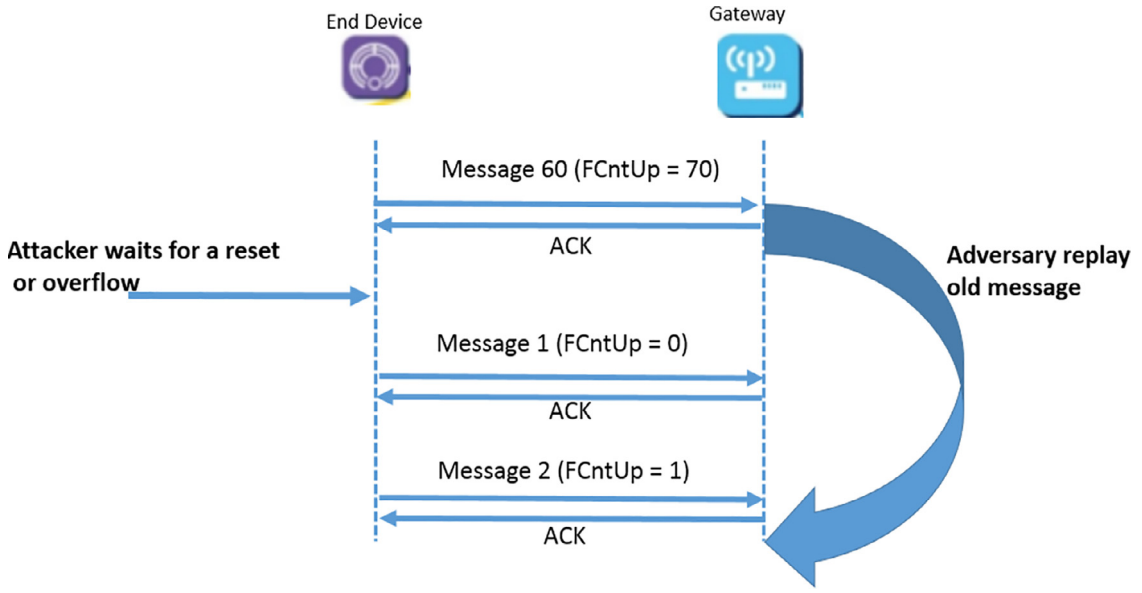


Fig. 22. Replay attack technique.

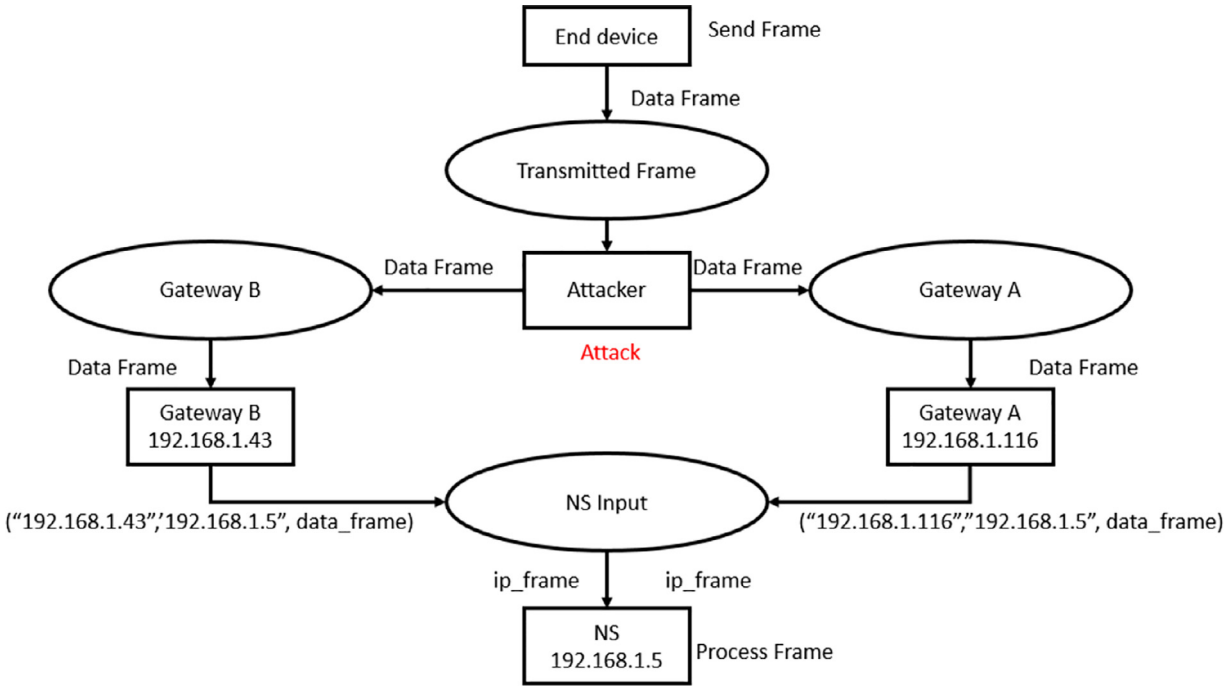


Fig. 23. Down-link routing attack process.

predefined, the attacker can do a replay attack when the counter overflows to 0. Thus, the attacker will monitor and store the up-link messages, and waits until the counter is reset. Then, the attacker will reply with a malicious message in which he will keep replaying. This will flood the ED with a high number of messages leading to a DoS attack [65,69,84,85].

In fact, the RF jamming is hard to detect and avoid. However, to prevent such attacks, the ED should again go through the activation procedure in order to obtain new session keys (change the keys periodically) or by adding the time-stamp or a counter to the message header.

8.2.3. Down-link routing attack

As per Fig. 23, the attack scenario consists of an ED, 2 GWs, an NS, and the attacker. Here, the ED will send an up-link message at a specific time interval to the authenticated GW. Simultaneously, the attacker eavesdrops on the transmission

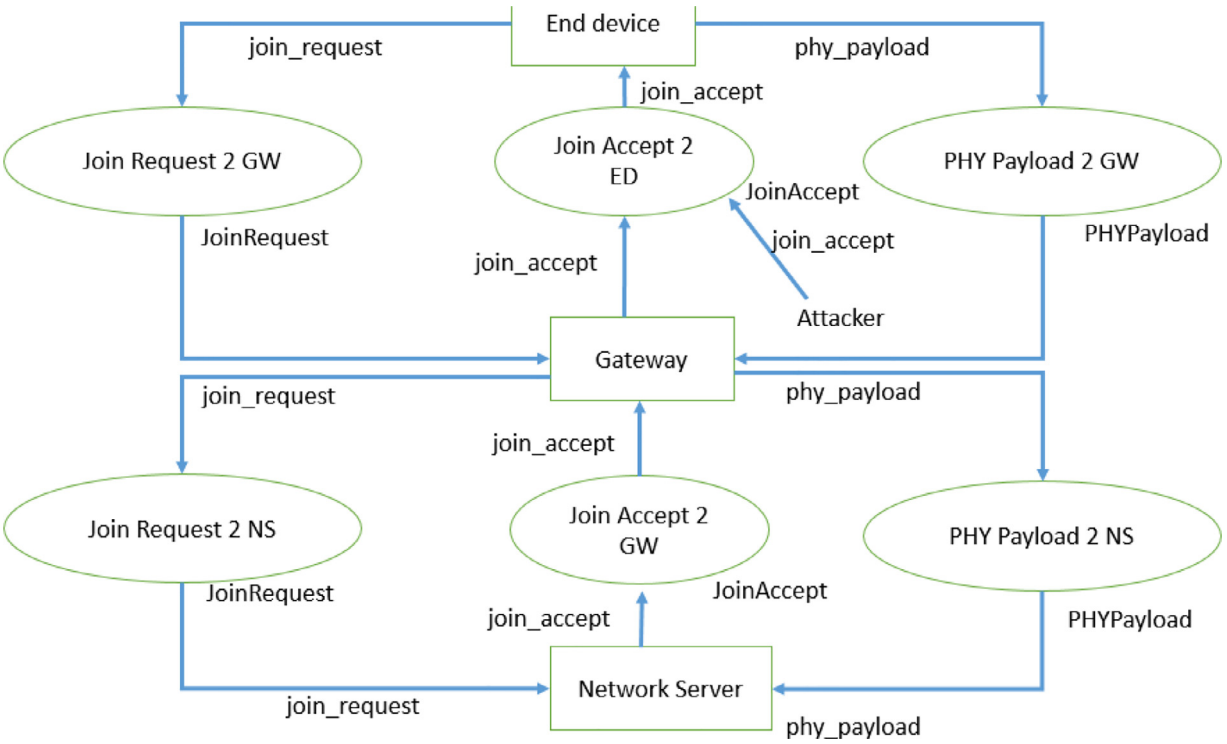


Fig. 24. Join-Accept replay attack process [86].

channel between the ED and the authenticated GW and replays the eavesdropped traffic to a different network through the compromised GW. The NS will validate the replayed packet and update the down-link routing path for the GW [86,87].

Now, the ED will send 2 up-link packets, which are captured by the attacker and replayed to the compromised GW. Consequently, the 2 GWs will receive the same up-link packets and forward them to the NS, which will receive 4 up-link packets. Hence, the order will be depending on the transmission time and the used network speed. However, the NS will accept only the first non duplicated.

8.2.4. Join-accept replay attack

This attack scenario consists of an ED, a GW, an NS, and the attacker as shown in Fig. 24 [86,87]. First, the ED will join the network using OTAA and sends an up-link packet which includes the application data with the security context. The role of the GW is to forward the traffic between the NS and the EDs.

The NS will receive the join-request message forwarded by the GW, and if accepted, the NS will reply by a join-accept message to the corresponding ED. Hence, the attacker will replay a join-accept message to the ED before receiving the authenticated one from the NS. Thus, this attack will prevent the further transmitted packets from being accepted, and the communication will fail between the NS and the nodes until a new OTAA communication connection is established [86,87].

As a countermeasure, We propose to use a pseudo-random nonce generation to authenticate the join-accept message at the ED. The nonce can be derived from these parameters: *AppKey, AppNonce, NetID, DevNonce*, and *counterJoin – Request*.

8.2.5. Beacon synchronization attack

In LoRaWAN, the class B beacon is not secured as shown in Fig. 25. In fact, the attacker can compromise the GW and send fake beacons to the EDs. Consequently, the EDs will open several unconfirmed received windows, which increases the collisions between the transmitted packets. Hence, using a key at the GW to authenticate the transmission will solve this issue [86,87].

As per Fig. 25, this attack scenario consists of 3 main elements: the GW, the ED, and the attacker. The GW broadcasts a beacon at a specific time interval. Simultaneously, the attacker broadcasts a malicious beacon with a high signal strength in order to replace the authenticated one. The broadcasted beacon contains a time stamp reference. The ED will receive the beacon and processes it during the reception window (Br) calculation for ensuring that this beacon is the only one to process. To prevent this attack, it is important to use *MIC* instead of the *PHY CRC*, which is responsible for the integrity check value and for the beacon frame authentication.

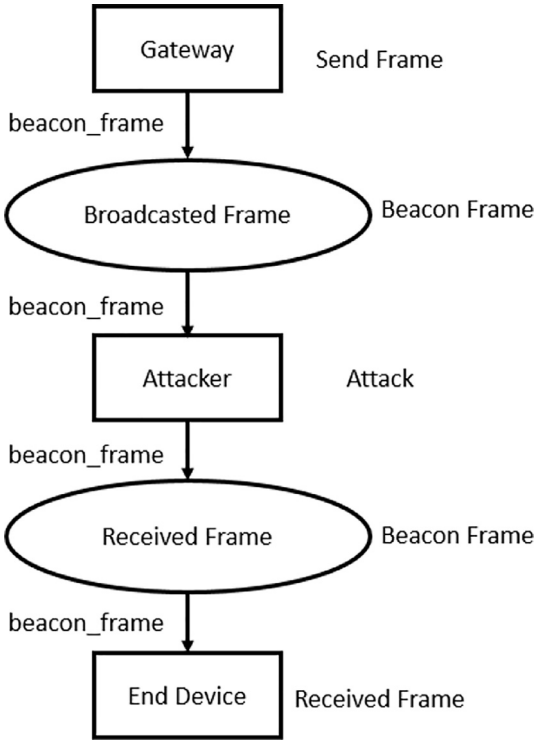


Fig. 25. Beacon synchronization attack.

8.2.6. ACK spoofing attack

The ACK spoofing attack is due to the lack of association between the data being confirmed and the acknowledgment. This allows the attacker to employ any down-link ACK message to confirm any up-link message coming from the same ED. Given that the attacker is capable of preventing any down-link frame reception, the replay protection method will not defend against this type of attacks [65,69].

Fig. 26 illustrates the ACK spoofing attack where the GW is compromised and plays a role in preventing some messages from being transmitted and received. Thus, the GW will block the ACK from reaching the ED. In this case, the ED will send another message *M*2, but the message will not be transferred successfully to the NS since the GW responds by using the first ACK in order to trick the ED to think that the second message is being transferred successfully to the NS. To prevent this attack, a *MIC* can be used to check the message integrity between the AS and the NS.

8.2.7. Destroy, remove, or steal ED

In LoRaWAN, the generated session keys are allocated to each device during their manufacturing process. Thus, when the attacker compromises a single session key, this will not compromise the traffic transferred all over the network, but it will compromise the availability, confidentiality, and the integrity of the stored data [82].

8.2.8. False join packets

In LoRaWAN, the two parameters *JoinEUI* and *DevEUI* might be used to perform the false join attack [82]. These two parameters are important for the join-procedure and they are protected by the *MIC* with another parameter, the *JoinNonce*. Hence, this type of attack is classified as unlikely to happen and when it does, it will affect the network availability.

8.2.9. Network flooding attack

In LoRaWAN, the ED can be used to perform an attack against the whole LoRaWAN network; it can flood the network with packets compromising the network availability. To resist such an attack, airtime restrictions should be added [82].

8.2.10. Selective forwarding attack

The attacker will perform this attack by forwarding selective packets compromising the network availability. Thus, using the IDPS will detect and prevent such an attack. [82].

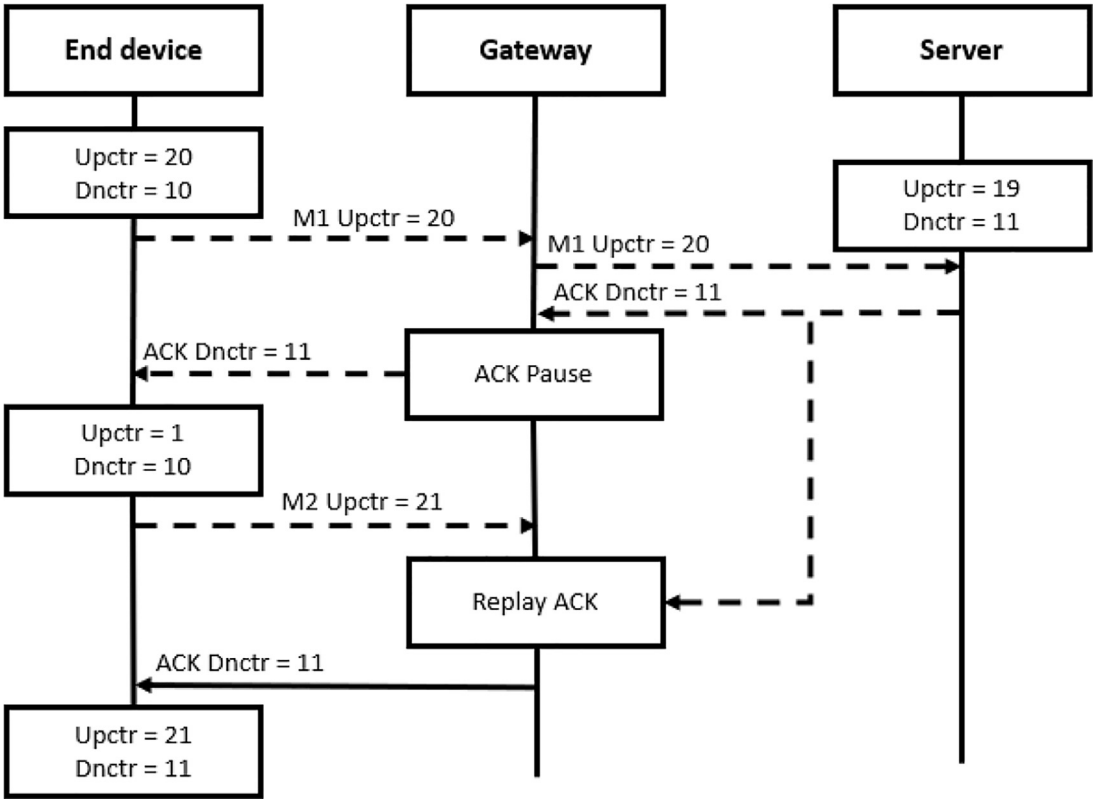


Fig. 26. ACK spoofing attack [69].

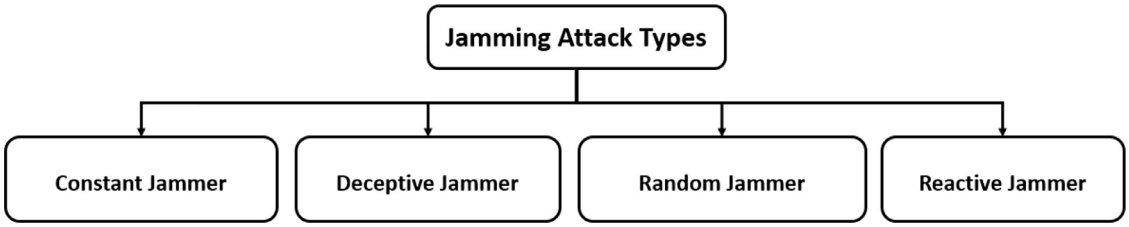


Fig. 27. Types of jamming attacks.

8.2.11. Jamming attack

Jamming is one of the most challenging issues facing wireless technologies and more specifically in the IoT domain. The jamming attack consists of transmitting a radio signal at the same frequency (carrier) of an ongoing radio transmission to disrupt it. In this context, an attacker can jam legal LoRaWAN radio transmissions by "using commercial-off the-self LoRa hardware". Considering LoRa devices, implementing the CSS modulation, the attacker takes advantage of the robust LoRaWAN transmission model in order to jam the LoRa network. For example, the attacker can use the Arduino platform and a LoRa radio modulation to jam the transmission channel at specific frequencies in order to corrupt the legal transmitted messages. [88,89]. Jamming attacks, illustrated in Fig. 27, are classified into four types [63,90] as follows:

1. **Constant Jammer:** this type of attack is performed by transmitting bits constantly on a specific channel. In this case, the attacker does not care if this channel is idle or not. Additionally, the attacker is not aware of the target channel protocol, bit rate, etc. This type of attack can be implemented using two types of wireless devices, a wireless wave generator and an ordinary wireless device that continuously transmit the data.
2. **Deceptive Jammer:** In this case, the attacker aims to continuously send packets into the target channel, forcing a node to function in a receiving mode because the channel is not idle to send new packets. The difference between this jammer and the constant one is that the attacker here is aware of the tract channel protocol. consequently, in this case, the jammer transmits legitimate packets at a high rate.
3. **Random Jammer:** this type of attack consists of non-continuous signal transmission, switching between transmitting and sleeping state. Hence, this type of jamming is based on two parameters, the *U* and the *td*, which could be random

or fixed. The *td* parameter represents the duration of the attack. The *U* parameter represents the sleep time. The attack starts by emitting a continuous signal for a *td* time and then, the transmission is turned off for the next *U* time. This attack consumes more energy than the constant jamming due to the use of a high powered battery or solar energy.

4. **Reactive Jammer:** this type of jamming attack is different from the two previous ones. The two previous jamming attacks block the channels without taking into consideration if the channels are busy or idle. Thus, the reactive jamming attack is based on monitoring the channel to check if it is idle or not. If the channel is idle, the jammer will not transmit any signal. Otherwise, the attacker transmits the attack signal for compromising the availability of the channel.

Jamming attacks can be prevented by detecting the devices that have abnormal behaviors. This can be done while the attack is running since all the malicious communicating devices will be detected and dropped from the network. In addition, to preserve the ED availability, the network administrator can switch the transmission to another frequency band.

8.3. Confidentiality attacks

In this section, the LoRaWAN confidentiality attacks are presented. In addition, their corresponding countermeasures are included.

8.3.1. Eavesdropping

LoRaWAN uses AES-128 in a counter mode to ensure the message confidentiality. The packet counter is being used as an input. In the ABP activation mode, the network and application keys are static and only the counters are updated during the session. However, when the counter overflows, the counter value will be reset and consequently, the same key-stream will be produced. In other words, when the counter value is repeated, the AES will reproduce the same key-stream since AES-CTR mode is a stream cipher [65,91,92].

The cipher-text is transmitted over the air and it is known. To get the new plain-text from the received cipher-text for a specific counter, the attacker should know or chose the previous plain-text $P1$. This is possible if an attacker employs chosen or known plain-text or cipher-text attacks [6,93]. Therefore, previous plain-texts $P1$ can be known or chosen by attackers and consequently, $P2$ can be recovered. In this case, the confidentiality of next communication messages is broken.

Since the confidentiality key is static and the counter can be reset, the same key-stream K is produced between two reset periods. This leads to a partial confidentiality issue since if two collected cipher-texts ($C1 = P1 \oplus K$ and $C2 = P2 \oplus K$) with the same key-stream block are XOR-ed ($\oplus$), their corresponding output will be equal to their corresponding plain blocks $P1$ and $P2$ as follows:

$$C1 \oplus C2 = P1 \oplus P2 \tag{7}$$

Hence, to prevent such attack, we can use the nonce instead of the counter value, since the nonce is generated by a secure pseudo-random number generator, which would reduce the collision chances and prevent the device from resetting or starting up using the same value every time. Additionally, using a re-key on any reset makes the attack very hard since the attacker is working on collecting information about the messages and the session keys. Thus, updating the network and the application session keys can prevent this attack. Therefore, when the counter value reaches the maximum value, the ED will be reset and will activate the update key derivation process. This scheme is similar to the one employed in the OTAA.

8.3.2. Network traffic analysis attack: privacy attacks

Traffic analysis is the process of capturing and analyzing the packets that are being transmitted through the GW. Many software tools can be used for analyzing the traffic such as Wireshark, Tcpdump, Kismet, Scapy, etc. However, these tools are divided into two main categories:

- **The sniffer:** used for capturing the transmitted packets.
- **The protocol analyzer:** used for decoding and analyzing the captured packets.

Thus, the main goal of this attack is to gather information about the transmitted data affecting the confidentially and the privacy of the system. To prevent data confidentiality attack, an encryption mechanism can be implemented to secure the transmitted packets. Besides, using variable and different identifications for each session, traffic analysis attacks will be more difficult.

8.4. Integrity attack

In the following, the bit flipping attack is described. In addition, we propose a security countermeasure to prevent it.

8.4.1. Bit flipping attack

The integrity of LoRaWAN messages could be targeted via the "bit flipping attack" as shown in Fig. 28, whereby the attacker can compromise the transport layer between the AS and the NS, which is an insecure channel.

As shown in Fig. 29, the attacker can change a specific field in the ciphertext without decrypting it. This is feasible in some encryption modes where both the plaintext and the ciphertext have the same bit order. In this case, the attacker can

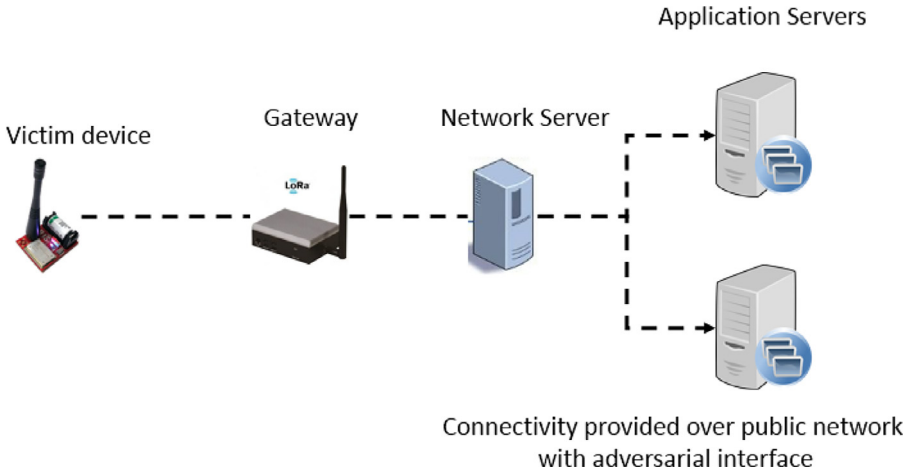


Fig. 28. Bit flipping attack [94].

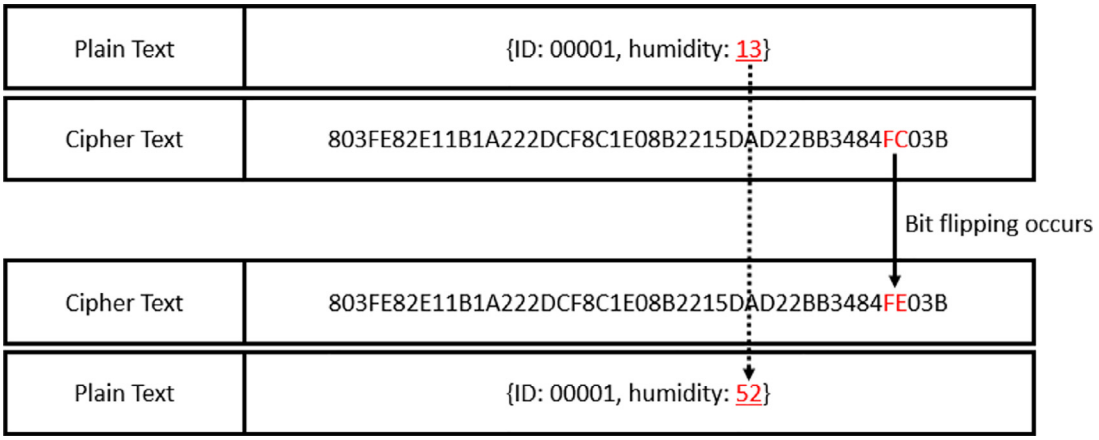


Fig. 29. Bit flipping attack mechanism.

Algorithm 1 Key generation process.

1: **procedure** KEY_GENERATION
2: $NwkSKey \leftarrow AES128_{Encr}(AppKey, 0x01||AppNonce||NetID||DevNonce||pad16)$
3: $AppSKey \leftarrow AES128_{Encr}(AppKey, 0x02||AppNonce||NetID||DevNonce||pad16)$
4: $mac \leftarrow AES128_{c}mac(AppKey, MHDR||AppNonce||NetID||DevAddr||RFU||RxDelay||CFList)$
5: $MIC \leftarrow mac[0 \rightarrow 3]$

modulate the bits in the ciphertext at the same position of the plaintext. This is possible since the ED in LoRaWAN uses the CTR mode as a block cipher mode for encrypting the messages, which is essentially a stream cipher mode based on XOR operation. The below equation shows the crypto logical encryption algorithm of the payload frame, where *si* is the counter block and *K* is the key used for encryption [95].

$$\begin{aligned} &Si = aes128encrypt(K, Ai),\ fori = 1 \ldots k \\ &S = S1|S2|\ldots|Sk \\ &S = S1|S2|\ldots|Sk \end{aligned} \tag{8}$$

This attack can be prevented by running the *MIC* for integrity checking at the AS, instead of checking it at the NS, or by removing the *CRC* and adding the *MIC* for integrity checking.

In Table 8, we summarize the LoRaWAN attacks, threats, and the proposed countermeasures.

Table 8
LoRaWAN threats, vulnerabilities, and countermeasures.

Type of threat	Type of attack	Attack description	Proposed countermeasure
Authentication	Man-In-The-Middle	Exposure of the secret keys will occur when capturing transmitted messages between the servers	Use *MIC* with the obtained hash $h(AppKey)$ in a nonlinear non-invertible manner.
Availability	Sinkhole	Transmission of certain traffic to a specific node and using it for attacking other nodes leading to DoS or DDoS	Use IDS/IPS for detecting and preventing such attacks
Availability	Replay	Using RF jamming to jam the message which results in DoS	Go through activation procedure for generating new session keys or use the timestamp in the message header
Availability	Down-link routing	Eavesdropping on the 2 up-link intervals at a specific time stamp. The attacker will send them through 2 GW's resulting in 4 up-link packets, from which the NS accepts only the first non duplicated	GW should be authenticated
Availability	Join-Accept replay	The attacker replays the join-accept messages leading to failure in the communication between the nodes and the NS	Use a pseudo-random nonce generation to authentication join-accept
Availability	Beacon synchronization	The EDs dismiss the down-link reception windows leading to DoS	Use MIC instead of the PHY CRC
Availability	ACK spoofing	The attacker captures a down-link ACK message and uses it for confirming another up-link message	Using *MIC* that is responsible for integrity checking on both the AS and NS
Availability	Jamming	Using malicious entities to jam the network and flood it which results in DoS and damaging the messages which are transmitted over the jamming frequency	Switching to another frequency
Confidentiality	Eavesdropping	The attacker applies a chosen or known plain text $P1$ which results in recovering plain text $P2$ leading to confidentiality issue	Use the nonce instead of the counter value
Confidentiality	Network traffic analysis	The attacker can configure a GW in order to receive the transmitted packets and get access to confidential data	Using variable and different identifications for each session
Integrity	Bit flipping	The attacker compromises the transport layer between the AS and the NS and introduces modifications to the communicated messages	Run *MIC* for integrity checking at the AS instead of checking it at the NS, or remove *CRC* and add *MIC* instead

9. LoRaWAN v1.1 security problems and benefits

Despite its early introduction and rapid adoption to overcome the limitations of previous versions (i.e v1.0 and v1.0.2), LoRaWAN v1.1 [10] is already suffering from different security problems. These problems will be discussed in terms of security issues, challenges and threats, respectively. However, security benefits will also be highlighted.

9.1. Security challenges & issues

Despite the promising advantages that LoRaWAN v1.1 offers, especially in terms of communication and availability, except that it seems to be suffering from different security challenges and issues. Burton et al. were among the first to present and discuss them [82].

- **Insufficient Gateways:** using a small number of gateways may result into the interruption of the availability of communications between the End Devices (ED) and the server.
- **Lack of Re-Keying Mechanisms:** which prevent any re-keying on-demand/periodic attempt(s) of the root keys during the Over-The-Air Activation (OTAA) sessions.
- **Undefined Session Duration:** which would result into further ambiguity without knowing how long a session can last, especially since higher protocol layers' session times will not be defined in a given specification document.
- **Lack of Tamper-Resistant EDs:** which allows their easy physical destruction and/or capture, especially with the absence of any physical protection.
- **Root Key Exposure:** especially when AES-128 root keys are not safely and securely stored, tracked and periodically updated would result into revealing and retrieving the sensitive information stored in the exploited ED.
- **Improper Session Termination Protocols:** usually result into allowing a given attacker to intercept the half-closed session and retrieve vital information about both ends due to the session not being permanently closed.
- **Lack of Integrity Protection:** unlike the end-to-end encryption of application payloads, the integrity relies on a hop-to-hop protective manner, which allows any malicious Network Server (NS) to eavesdrop and alter any message content of the transmitted data. Thus, targeting both integrity and confidentiality.

Table 9
LoRaWAN: v1.0.2 Vs. v1.1.

LoRaWAN			
v1.0.2		v1.1	
Vulnerabilities	Consequences	Solutions	Description
Constant reuse of frame counter values	Affects ABP and OTAA due to devices reset or counter overflow	Introduces ED re-keying explicit mechanism	New session key established, frame counters stored in non-volatile memory
Reuse of the same nonce values	Lack of full prevention of their reuse	Use of counter-nonces	Prevent nonce values' reuse & regeneration
Resource constrained EDs	Inability to prevent replay attacks	Tracks last observed Join-Nonce value	Discards replayed join-accept message
Only tracks the most recently used *N* DevNonce values	Weak replay protection mechanism	Tracks the last *DevNonce* per ED	Eliminates the replayed join-request message
Lack of acknowledgment association mechanisms	Acknowledgments are Unassociated with confirmed data messages	Adds a *ConfFCnt* in MIC calculation of data messages	*MIC* check fails if the captured *Ack* is replayed
Lack of available association mechanisms	Inability to associate join-accept messages with requests	Adds a *DevNonce* in *MIC* calculations of join-accept messages	*MIC* check fails if the *MIC* is calculated using different *DevNonce*
Inability to confirm security session contexts	ED/NS disassociated & using different security context	Introduces RekeyInd/RekeyConf MAC commands	ED/NS verifies each other's security contexts
No End-to-End integrity protection	Application data integrity unprotected during NS/AS transport	Applies End-to-End integrity protection	Appliance varies per application's decision

- **Compatibility Issues:** having LoRaWAN v1.1 version being compatible with other versions of LoRaWAN (i.e v1.0 and v1.0.2), leaves it exposed to similar security risks, threats, vulnerability and challenges.

9.2. Security threats

Security threats surrounding LoRaWAN v1.0.2 seems to be catching up with its newer version of v1.1, especially in terms of attacks/cyber-attacks targeting the LoRaWAN communication domain. Donmez et al. were among the first to discuss and highlight this issue in [77]. These attacks and represented and discussed briefly, aside the bit-flipping, denial-of-service and network flooding attacks.

- **Network Traffic Attacks:** target the physical layer once a rogue gateway is set to exploit known vulnerabilities to either conduct DoS attacks or impersonation attacks.
- **Physical Attacks:** include physical destruction, damage or hijacking the end-devices in hopes to retrieve the single root key to compromise a stored data's device.
- **Firmware Attacks:** network nodes are still not fully/partially protected from any firmware change(s) which can result into end-devices being prone to a firmware replacement with their key materials being reused in case of an attacker (whistleblower) gained a physical access to the device. This may lead to an illegal privilege escalation or performing replay attacks.
- **Routing Attacks:** still impose a serious threat especially if they are classed as selective forwarding attacks, like sink-hole/blackhole attacks, blocking/overwhelming network nodes to intercept network traffic through false advertising and rogue modification of routing information.
- **Self-Replay Attacks:** threaten the availability of LoRaWAN v1.1 communications, since the network servers packets can be intercepted before the end-device is able to receive them. This can also leave these end-devices prone to jamming attacks.
- **False-Data Attacks:** are still among the major security threats since attackers can still intercept incoming/outgoing network communications and modify them to inject false/malicious data. Allowing attackers to hide their malicious activities in a covert way.

9.3. Security benefits

Despite the security problems that LoRaWAN v1.1 suffer from, LoRaWAN v1.1 overcomes the vulnerabilities that LoRaWAN v1.0.2 suffer from. Donmez et al [77] highlighted the main LoRaWAN v1.0.2 vulnerabilities and their suitable solutions in LoraWAN v1.1. We summarize them in the following table Table 9.

Table 10
Qualitative risk analysis.

Kind	Threat	Needed attacker capabilities	Motivation of the attacker	Likelihood	Impact	Risk
Availability	Sinkhole (alone)	Basic	Low	Likely	High	Minor
	Replay (ABP)	Basic	High	Likely	High	Critical
	Down-link Routing (OTAA)	Moderate	High	Possible	Medium	Major
	Join-Accept Replay (OTAA)	Moderate	High	Possible	High	Major
	Jamming (ABP, OTAA)	Basic	Moderate	Possible	High	Critical
	ACK Spoofing (ABP, OTAA)	Extensive	Low	Unlikely	Medium	Major
	Destroy, Remove, or Steal ED	Low	Medium	Likely	Moderate	Major
	Network Flooding (ABP, OTAA)	Low	Low	Unlikely	Significant	Minor
	Destroy, Remove, or Steal ED	Low	Medium	Likely	Moderate	Major
	False Join Packets (OTAA)	Low	Low	Unlikely	Significant	Minor
	Selective Forwarding (ABP, OTAA)	Low	Low	Unlikely	Significant	Minor
	Beacon Synchronization (ABP, OTAA)	Moderate	Low	Unlikely	High	Major
Authentication	Man In The Middle (ABP, OTAA)	No rating	Low	Unlikely	High	Critical
	Security Parameter Extraction(ABP, OTAA)	Low	Medium	Possible	Significant	Major
	Device Cloning or Firmware Replacement (ABP, OTAA)	Low	High	Likely	Significant	Critical
Integrity	Bit Flipping (ABP)	No rating	Low	Unlikely	Moderate	Minor
Confidentiality	Eavesdropping (ABP)	Basic	High	Possible	Medium	Minor
	Network Traffic Analysis Attack (ABP, OTAA)	Basic	Low	Unlikely	Moderate	Minor

10. LoRaWAN security threat and risk assessment

This section presents the risk assessment of LoRaWAN threats. First, the security assessment methodology is described including the different risk assessment methods. Then, we present a qualitative risk assessment evaluation of the LoRaWAN security threats.

10.1. Risk assessment methodology

The security assessment methodology that is used in this survey includes the likelihood, impact, and risk assessment methods. A detailed and categorized security analysis is included by applying these methods to each threat [82]. The three security assessment methods are described below:

1. **Likelihood assessment:** it consists of three discrete levels, which are likely, possible, and unlikely, depending on the probability of each threat. However, for having a better evaluation, two other factors (motivation level and technical difficulty) are also considered. These two factors can be described as follows [82]:
   - **Motivation level:** it presents the level of motivation for an attacker to maintain an attack. It can be one of three levels, low, medium, and high.
   - **Technical difficulty:** it represents the bounds and the challenges that face an attacker when performing an attack. This can be one of the three categories, none, solvable, and strong. As an example, the replay attack, which is based on capturing a message and replaying it, can be classified as “solvable” due to its medium technical difficulty.
   - **Likelihood of an attack:** it represents the risk factors which are the motivation level, and the technical difficulty of an attack.
2. **Impact assessment:** it is classified depending on the threat impact, which can be: minimal, moderate, and significant. However, for having a better evaluation, two other factors (scale level, and detectability and recoverability) are also used for likelihood assessment. These two factors can be described as follows:
   - **Scale level:** it presents the affected area scale due to a performed attack. This can be classified as ED, LoRaWAN, and LoRaWAN-EN. The ED level represents the attack affecting only one node. The LoRaWAN level indicates that the network is being employed by the attack. The LoRaWAN-EN level refers to multiple networks in case the attack invokes more than one network.
   - **Detectability and recoverability:** it describes the detectability of an attack and its impacts on the system and the recoverability possibility after the attack. It can be classified as low, medium, and high.
   - **Impact of an attack:** it is represented in Table 10 based on the two impact assessments, the scale level, and the detectability and recoverability).
3. **Risk assessment:** the risk assessment is related directly to the likelihood and the impact of an attack. It can be categorized as:
   - **Critical risk:** it indicates that the attack is likely happening and its impact is significant.
   - **Major risk:** it indicates that the attack is likely happening and its impact is moderate, or the attack is possible and its impact is significant, or the attack is possible and its impact is moderate.
   - **Minor risk:** it indicates that the attack is unlikely to happen and its impact is minimal.

10.2. Security risk analysis of LoRaWAN v1.1:

The risk assessment evaluation of the LoRaWAN vulnerabilities and threats is summarized in Table 10. Moreover, in the below, the different LoRaWAN security attacks are categorized based on their security risk levels [82]:

1. **LoRaWAN v1.1 exhibits minor security risk for the following attacks:**
   - Bit-flipping
   - Destroy, Remove, or Steal ED
   - False Join Packets
   - Network Flooding
   - Network Traffic Analysis
   - Selective Forwarding
   - RF Jamming
   - Sinkhole
   - Eavesdropping
2. **LoRaWAN v1.1 exhibits major security risk for the following attacks**:
   - Beacon Synchronization DoS (major risk for availability and minor risk for the rest)
   - Join Accept Replay (major risk for availability and minor risk for the rest)
   - Down-link Routing (major risk for availability and minor risk for the rest)
   - ACK Spoofing (major risk for availability and minor risk for the rest)
   - Security Parameter Extraction (minor risk for integrity and availability and major risk for the rest)
3. **LoRaWAN v1.1 exhibits critical security risk for the following attacks**:
   - Replay (critical risk for availability and minor risk for the rest)
   - Jamming (critical risk for availability and minor risk for the rest)
   - Man in The Middle (critical risk for authentication and access control availability and minor risk for the rest)
   - Device Cloning or Firmware Replacement (critical risk for authentication and access control, major risk for confidentiality and integrity and minor risk for availability)

10.3. Existing LoRaWAN security assessment

As a result of evaluating the different security attacks, threats and vulnerabilities of LoRaWAN standards and protocols (i.e v1.0.x and v1.1), the most recent security solutions to mitigate these issues are presented in the following.

Yang [96] presented and discussed several vulnerabilities of LoRaWAN v1.0 that still exist in LoRaWAN v1.1, such as the "bit-flipping attack", along with their impact on LoRaWAN servers' security. In addition, Conan et al [97] presented and discussed several proof-of-concept solutions against LoRaWAN (i.e packet forging), sigfox (i.e replay) and NB-IoT attacks and vulnerabilities, and presented how packet forging can be mitigated by using a message authentication algorithm such as AES-CMAC mode with 128 bits secret key (*NwkSKey*).

Furthermore, Yang et al. [6] also presented a proof-of-concept and analysed the main attacks that target the LoRaWAN v1.0.2's confidentiality, integrity and availability, including replay, eavesdropping, packet modification, ACK spoofing and battery exhaustion, while suggesting various improvements. Moreover, LoRaWAN1.1ı́s security aspects were extensively studied and detailed before presenting the suggested ramification strategies by Butun et al. in [30]. Also, a comprehensive security risk analysis of LoRaWAN v1.1 protocol was provided in terms of security risks, in addition to a threat catalogue to discuss the likelihood and impact of each threat [32]. However, some overviews over the security issues surrounding IoT's main networks (i.e Wi-Sun, LoRaWAN 1.0.x/1.1, NB-IoT, Sigfox, DASH7), were less fruitful in providing a technically suitable solution in overcoming persistent jamming attacks against LoRaWAN, such as the case of [98].

Unfortunately, Ingham et al [99] discussed how LoRaWAN is still susceptible to various security flaws (i.e key management and counter management) and attacks (i.e bit flipping attacks, eavesdropping, and encryption flaws) but with no proper suggestion to overcome them, except for the predictive signal jamming attack analysis in LoRaWAN. A similar study was conducted previously by Miller in [27] that analyzes all possible vulnerabilities (i.e key management, communications, and network connection) and related countermeasures for LoRaWAN v1.0's. Kim and Song [100] presented dual key-based activation scheme to overcome different security issues that surround the *AppKey*'s root key in LoRaWAN v1.0. Moreover, the authors suggested the use of a second root key to separate the application/network session keys from the same root key and implemented its use in LoRaWAN v1.1. The simulation results indicated that this scheme is feasible in terms of delay and battery consumption. Voigt et al [101] presented several methods that enhance the LoRaWAN v1.0 network's capacity and availability under inter-network interference situation (i.e jamming). The experimental results show that the Data Extraction Rate (DER) is improved by 33% when using directional antennas, and by 133% when using additional gateways.

Gao et al [102] addressed to the issue of replay attacks and presented a Secure-Packet-Transmission (SPT) scheme as a suitable countermeasure to overcome replay attacks whenever the attacker obtains the root key in LoRaWAN v1.1. The simulation results indicate that the SPT is a lightweight, efficient and feasible solution that protects LoRaWAN v1.1 against malicious behaviours and attacks. Tomasin et al. [69] described LoRaWAN v1.0's security problems regarding the *DevNonce*,

which is used during the Over-The-Air-Activation (OTAA) procedure to prevent replay-attacks during the join-request messages from the end-device by the NS.

Gunathilake et al [103] covered the necessity of a next generation lightweight cryptography (LWC) as a novel approach headed for smart security applications in the IoT's low-power constrained data-processing devices especially those relying on LoRaWAN protocols. However, further theoretical, application-oriented and feasible empirical researches need to be conducted to ensure the IoT's right security assurance and privacy protection. Sanchez et al [73] evaluated LoRaWAN's key management security vulnerabilities and presented different theoretical alternative schemes whilst providing a comparative conceptual analysis in terms of their overhead, such as the case of Ephemeral Diffie-Hellman Over COSE (EDHOC) that provides flexibility in updating session keys at a low computational cost, and Static Context Header Compression (SCHC) algorithm to enable IPv6 communication between the LoRaWAN's end-nodes and external networks using the SCHC scheme. Reynders et al [104] presented a new MAC layer, RSLoRa, to improve reliability and scalability of LoRaWAN. This solution is based on a two-step lightweight scheduling to enable simultaneous transmissions and packets collision reduction. Simulation results show that this solution improves the LoRaWAN's performance in terms of packet error ratio, throughput, and fairness with a reasonable energy efficiency. Finally, Tsai et al. addressed to the AES encryption issue that affects the battery-powered IoT devices in terms of power consumption and resource exhaustion [105]. As a result, a low power consumed AES encryption architecture named as Low-Power AES Data Encryption Architecture (LPADA) that enables mutual authentication and resists against replay/eavesdropping attacks, along a key updating procedure that secure the session-key renewal were presented. LPADA aims to reduce the static/dynamic power consumption of the AES encryption. The experimental results show that the dynamic power is reduced by 62.0%, whilst the leakage power is lowered by 88.5%.

11. Recommendations

This section includes the recommendations that should be considered when designing a high level of security and performance for LoRaWAN systems. The LoRaWAN technology suffers from several security issues that affect the data/network confidentiality, integrity, availability, and authenticity. In this context, non-cryptographic solutions can be employed to detect (e.g. IDS), prevent (e.g. IPS), analyze (e.g. honeypot), and block (e.g. firewall) security attacks. Thus, a hybrid Intrusion Detection/Prevention scheme can be implemented at the ED, GW, NS, and AS. Signature and specification techniques can be introduced at the EDs, while an anomaly detection scheme can be applied at the GW. In addition, an artificial-intelligence-based anomaly detection scheme can be applied at the NS. The IDS structure for a LoRaWAN system should be hybrid and distributed according to the device characteristics. Thus, using a network IDS at the GW and a host IDS at the ED plays an essential role in securing the LoRaWAN network. Also, for additional security, a honeypot system may be introduced with dynamic configuration. Moreover, a mini firewall can be integrated with an IDS/IPS in addition to the honeypot system towards preventing and filtering unwanted or malicious network traffic from reaching its destination. Furthermore, as mentioned in Section 7.4, there are many attacks that affect the availability, which can be prevented by taking security measures as follows:

- **Sinkhole**: this issue can be resolved by implementing an IDPS.
- **Jamming**: the message should go through another activation procedure in order to generate new session keys. Also, switching to another frequency may be another solution.
- **Down-link Routing**: this attack can be thwarted by authenticating the GW.
- **Join-Accept Replay**: the use of a pseudo-random nonce will authenticate the join-request message and thus preventing this type of attacks.
- **Beacon Synchronization**: in this case, using the *MIC* instead of the *CRC* can prevent the dismissal of the reception windows.
- **ACK Spoofing**: the use of *MIC* on both the AS and the NS will resolve the issue of bypassing the confirmation of the up-link message.

Other attacks affecting the system confidentiality may be resolved as follows:

- **Eavesdropping**: the attacker uses a known plain text for recovering another plain text which will directly affect the confidentiality of the message. Thus, using the nonce instead of the counter will solve this issue.
- **Network Traffic Analysis**: the data is collected about the sent and received packets by compromising the GW. Thus, by using variable and different identifications for each session, traffic analysis attacks will be more difficult.

Other types of attacks affect the message authenticity and can be prevented as follows:

- **Man-In-The-Middle**: this type of attacks can be prevented by mixing *MIC* with $h(AppKey)$ in a nonlinear and non-invertible manner.

Attacks affecting the integrity can be resolved as follows:

- **Bit Flipping**: this attack compromises the transport layer between the NS and the AS, which makes possible the modification of a message. Thus, using *MIC* at the AS and replacing the *CRC* by the *MIC* permit the message integrity checking to prevent this attack.

Table 11
Comparison between different key management solutions.

Attack Type	Description	Target	Suggested solutions	Reason
Signal jamming	**Service disruption/denial**	**A**	**FH, NM**	**Less network overhead/bottleneck**
Packet modification	**Packet alteration/injection**	**P, I, A**	**TTL, NGLWC, HMAC**	**Enhanced real-time protection**
Bit flipping	**Cryptanalysis/DoS**	**P, I, A**	**NGLWC, DS, HMAC**	**Enhanced real-time security**
Replay	**Data intercepted, delayed/repeated**	**A**	**Timestamps, KAP, CHAP**	**Enhanced security/availability**

Finally, this paper recommends the following security measures to overcome persistent attacks against LoRaWAN v1.1 in Table 11, where A stands for availability, I stands for integrity, P stands for privacy, C stands for confidentiality, TTL stands for Time-to-Live, NGLWC stands for Next Generation Lightweight Cryptography, KAP stands for Kerberos Authentication Protocol, CHAP stands for Challenge-Handshake Authentication Protocol, FH stands Frequency Hopping, DS stands for Digital Signature, HMAC stands for Keyed Hash Message Authentication Code and NM stands for Noise Masquerading.

12. Conclusion

LoRaWAN security suffers from different vulnerabilities, which could be targeted by an attacker to launch authentication, availability, integrity and confidentiality attacks. Unless such issues are addressed, they will render LoRaWAN systems unsafe and unsuitable for IoT applications. In this paper, we analyzed the LoRaWAN security issues and attacks, and we proposed countermeasures towards preventing a set of these attacks.

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.

References

[1] J. Kim, J. Song, A secure device-to-device link establishment scheme for LoRaWAN, IEEE Sens. J. 18 (5) (2018) 2153–2160, doi:10.1109/JSEN.2017.2789121.
[2] G. Avoine, L. Ferreira, Rescuing LoRaWAN 1.0, in: International Conference on Financial Cryptography and Data Security, Springer, 2018, pp. 253–271.
[3] L. Ntseane, B. Isong, Analysis of LoRa/LoRaWAN challenges, in: 2019 International Multidisciplinary Information Technology and Engineering Conference (IMITEC), IEEE, 2019, pp. 1–7.
[4] L. Alliance, LoRaWAN r1. 0 open standard released for the IoT, 2015.
[5] N. Sornin, M. Luis, T. Eirich, T. Kramp, O. Hersent, LoRaWAN specification, LoRa alliance (2015).
[6] 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.
[7] L. Alliance, LoRaWAN™ 1.0. 2 regional parameters, no. Feb (2017) 1–55.
[8] D. Singh, O.G. Aliu, M. Kretschmer, Lora wanevaluation for ITt communications, in: 2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI), IEEE, 2018, pp. 163–171.
[9] L.A.T. Committee, et al., LoRaWAN 1.1 regional parameters, Standard V1 1(2018).
[10] L. Alliance, Lorawan™ 1.1 specification, LoRa Alliance 11 (2017). 2018–04
[11] L. Alliance, Lorawan 1.0. 3 specification, lora-alliance. org 1 (2018). 2018–07
[12] A. Yegin, T. Kramp, P. Dufour, R. Gupta, R. Soss, O. Hersent, D. Hunt, N. Sornin, LoRaWAN protocol: specifications, security, and capabilities, in: LPWAN Technologies for IoT and M2M Applications, Elsevier, 2020, pp. 37–63.
[13] E. Aras, N. Small, G.S. Ramachandran, S. Delbruel, W. Joosen, D. Hughes, Selective jamming of LoRaWAN using commodity hardware, arXiv:1712.02141. (2017).
[14] A. Augustin, J. Yi, T. Clausen, W.M. Townsley, A study of lora: long range & low power networks for the internet of things, Sensors 16 (9) (2016) 1466.
[15] L. Vangelista, A. Zanella, M. Zorzi, Long-range IoT technologies: the dawn of LoRa™, in: V. Atanasovski, A. Leon-Garcia (Eds.), Future Access Enablers for Ubiquitous and Intelligent Infrastructures, Springer International Publishing, Cham, 2015, pp. 51–58.
[16] M.W. Chase, D.E. Soltis, R.G. Olmstead, D. Morgan, D.H. Les, B.D. Mishler, M.R. Duvall, R.A. Price, H.G. Hills, Y.-L. Qiu, et al., Phylogenetics of seed plants: an analysis of nucleotide sequences from the plastid gene RBCL, Ann. Missouri Bot. Garden (1993) 528–580.
[17] S.A. A'ssri, F.H. Zaman, S. Mubdi, The efficient parking bay allocation and management system using LoRaWAN, in: Control and System Graduate Research Colloquium (ICSGRC), 2017 IEEE 8th, IEEE, 2017, pp. 127–131.
[18] S. Bhattacharya, S. Sridevi, R. Pitchiah, Indoor air quality monitoring using wireless sensor network, in: Sensing Technology (ICST), 2012 Sixth International Conference on, IEEE, 2012, pp. 422–427.
[19] S. Bjelcevic, J. Jemson, N. Karusala, D. Purcell, Lambs: light and motion based safety.
[20] K. Winkley, M. Veeman, A temperature-adjusted developmental timer for precise embryonic staging, Biol. Open (2018) bio–032110.
[21] B. Oniga, A. Munteanu, V. Dadarlat, Open-source multi-protocol gateway for internet of things, in: 2018 17th RoEduNet Conference: Networking in Education and Research (RoEduNet), IEEE, 2018, pp. 1–6.
[22] J. Petajajarvi, K. Mikhaylov, A. Roivainen, T. Hanninen, M. Pettissalo, On the coverage of LpWANs: range evaluation and channel attenuation model for LoRa technology, in: ITS Telecommunications (ITST), 2015 14th International Conference on, IEEE, 2015, pp. 55–59.
[23] 04181320.pdf, 2017, (http://advcloudfiles.advantech.com/ecatalog/2017/04181320.pdf). (Accessed on 12/03/2018).
[24] S. Park, S. Yun, H. Kim, R. Kwon, J. Ganser, S. Anthony, Forestry monitoring system using LoRa and drone, in: Proceedings of the 8th International Conference on Web Intelligence, Mining and Semantics, ACM, 2018, p. 48.
[25] W. Ayoub, M. Mroue, F. Nouvel, A.E. Samhat, J.-c. Prévotet, Towards ip over LpWANs technologies: LoRaWAN, dash7, nb-IoT, in: Digital Information, Networking, and Wireless Communications (DINWC), 2018 Sixth International Conference on, IEEE, 2018, pp. 43–47.
[26] M. Saari, A.M. bin Baharudin, P. Sillberg, S. Hyrynsalmi, W. Yan, LoRa – a survey of recent research trends, in: 2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), 2018, pp. 0872–0877, doi:10.23919/MIPRO.2018.8400161.
[27] R. Miller, LoRa Security: Building a Secure LoRa Solution, MWR Labs Whitepaper, 2016.

[28] J. Haxhibeqiri, E. De Poorter, I. Moerman, J. Hoebeke, A survey of LoRaWAN for Iot: from technology to application, Sensors 18 (11) (2018), doi:10.3390/s18113995.
[29] A. Lavric, V. Popa, Internet of things and LoRa low-power wide-area networks challenges, in: 2017 9th International Conference on Electronics, Computers and Artificial Intelligence (ECAI), 2017, pp. 1–4, doi:10.1109/ECAI.2017.8166405.
[30] I. Butun, N. Pereira, M. Gidlund, Analysis of LoRaWAN v1. 1 security, in: Proceedings of the 4th ACM MobiHoc Workshop on Experiences with the Design and Implementation of Smart Objects, ACM, 2018, p. 5.
[31] Lora iot sensor node solutions, advantech select, (http://select.advantech.com/lora/en-us/).
[32] I. Butun, N. Pereira, M. Gidlund, Demystifying security of LoRaWAN v1. 1(2018).
[33] Y. Jeon, H. Ju, S. Yoon, Design of an LpWAN communication module based on secure element for smart parking application, in: 2018 IEEE International Conference on Consumer Electronics (ICCE), 2018, pp. 1–2, doi:10.1109/ICCE.2018.8326112.
[34] M. Chan, D. Estève, C. Escriba, E. Campo, A review of smart homes-present state and future challenges, Comput. Methods Programs Biomed. 91 (1) (2008) 55–81.
[35] A. Caragliu, C. Del Bo, P. Nijkamp, Smart cities in Europe, J. Urban Technol. 18 (2) (2011) 65–82.
[36] J.-P.A. Yaacoub, M. Noura, H.N. Noura, O. Salman, E. Yaacoub, R. Couturier, A. Chehab, Securing internet of medical things systems: limitations, issues and recommendations, Future Gener. Comput. Syst. 105 (2020) 581–606.
[37] L. Catarinucci, D. De Donno, L. Mainetti, L. Palano, L. Patrono, M.L. Stefanizzi, L. Tarricone, An IoT-aware architecture for smart healthcare systems, IEEE Internet Things J. 2 (6) (2015) 515–526.
[38] L. Lipper, P. Thornton, B.M. Campbell, T. Baedeker, A. Braimoh, M. Bwalya, P. Caron, A. Cattaneo, D. Garrity, K. Henry, et al., Climate-smart agriculture for food security, Nat. Clim. Change 4 (12) (2014) 1068.
[39] T. Hägerstraand, What about people in regional science? Pap. Reg. Sci. 24 (1) (1970) 7–24.
[40] J.-P. Yaacoub, O. Salman, Security analysis of drones systems: attacks, limitations, and recommendations, Internet Things (2020) 100218.
[41] V. Sharma, I. You, G. Pau, M. Collotta, J.D. Lim, J.N. Kim, LoRaWAN-based energy-efficient surveillance by drones for intelligent transportation systems, Energies 11 (3) (2018) 573.
[42] J.-M. Martinez-Caro, M.-D. Cano, IoT system integrating unmanned aerial vehicles and LoRa technology: a performance evaluation study, Wirel. Commun. Mob. Comput. 2019 (2019).
[43] T. Gupta, F. Arena, I. You, Efficient resource allocation for Backhaul-aware unmanned air vehicles-to-everything (u2x), Sensors 20 (10) (2020) 2994.
[44] A. Rahmadhani, R. Isswandhana, A. Giovani, R.A. Syah, et al., LoRaWAN as secondary telemetry communication system for drone delivery, in: 2018 IEEE International Conference on Internet of Things and Intelligence System (IOTAIS), IEEE, 2018, pp. 116–122.
[45] V. Delafontaine, F. Schiano, G. Cocco, A. Rusu, D. Floreano, Drone-aided localization in LoRa iot networks, arXiv:2004.03852. (2020).
[46] J.-P.A. Yaacoub, O. Salman, H.N. Noura, N. Kaaniche, A. Chehab, M. Malli, Cyber-physical systems security: Limitations, issues and future trends, Microprocessors Microsyst. (2020) 103201.
[47] G. Loubet, A. Takacs, D. Dragomirescu, Implementation of a battery-free wireless sensor for cyber-physical systems dedicated to structural health monitoring applications, IEEE Access 7 (2019) 24679–24690.
[48] M. Luvisotto, F. Tramarin, L. Vangelista, S. Vitturi, On the use of LoRaWAN for indoor industrial IoTapplications, Wirel. Commun. Mob. Comput. 2018 (2018).
[49] H.H.R. Sherazi, M.A. Imran, G. Boggia, L.A. Grieco, Energy harvesting in LoRaWAN: a cost analysis for the industry 4.0, IEEE Commun. Lett. 22 (11) (2018) 2358–2361.
[50] K.E. Nolan, W. Guibene, M.Y. Kelly, An evaluation of low power wide area network technologies for the internet of things, in: Wireless Communications and Mobile Computing Conference (IWCMC), 2016 International, IEEE, 2016, pp. 439–444.
[51] B. Singh, B. Kaur, Comparative study of internet of things infrastructure and security, Global Wireless Submit 2016, 2016.
[52] V. Nikolić, N. Begenešić, The IoT bases: LoRa networks, Zbornik Meunarodne konferencije o obnovljivim izvorima električne energije–MKOIEE 5 (1) (2017) 235–238.
[53] M. Chen, Y. Miao, X. Jian, X. Wang, I. Humar, Cognitive-LpWAN: towards intelligent wireless services in hybrid low power wide area networks, IEEE Trans. Green Commun.Netw. 3 (2) (2018) 409–417.
[54] P. Neumann, J. Montavont, T. Noël, Indoor deployment of low-power wide area networks (LpWAN): a LoRaWAN case study, in: 2016 IEEE 12th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), IEEE, 2016, pp. 1–8.
[55] Y. Li, X. Cheng, Y. Cao, D. Wang, L. Yang, Smart choice for the smart grid: narrowband internet of things (Nb-IoT), IEEE Internet Things J. 5 (3) (2017) 1505–1515.
[56] R.S. Sinha, Y. Wei, S.-H. Hwang, A survey on LpWa technology: LoRa and Nb-IoT, ICT Express 3 (1) (2017) 14–21.
[57] K. Mekki, E. Bajic, F. Chaxel, F. Meyer, Overview of cellular lpwan technologies for IoTdeployment: Sigfox, LoRaWAN, and nb-IoT, in: 2018 IEEE International Conference on Pervasive Computing and Communications Workshops (percom workshops), IEEE, 2018, pp. 197–202.
[58] A.F. Molisch, K. Balakrishnan, C.-C. Chong, S. Emami, A. Fort, J. Karedal, J. Kunisch, H. Schantz, U. Schuster, K. Siwiak, IEEE 802.15. 4a channel model–final report, IEEE P802 15 (04) (2004) 0662.
[59] H. Harada, K. Mizutani, J. Fujiwara, K. Mochizuki, K. Obata, R. Okumura, IEEE 802.15. 4g based wi-sun communication systems, IEICE Trans. Commun. 100 (7) (2017) 1032–1043.
[60] K. Mochizuki, K. Obata, K. Mizutani, H. Harada, Development and field experiment of wide area wi-sun system based on IEEE 802.15. 4g, in: 2016 IEEE 3rd World Forum on Internet of Things (WF-IoT), IEEE, 2016, pp. 76–81.
[61] A. Lavric, V. Popa, Internet of things and LoRa low-power wide-area networks: a survey, in: Signals, Circuits and Systems (ISSCS), 2017 International Symposium on, IEEE, 2017, pp. 1–5.
[62] J. de Carvalho Silva, J.J. Rodrigues, A.M. Alberti, P. Solic, A.L. Aquino, LoRaWAN-a low power wan protocol for internet of things: A review and opportunities, in: Computer and Energy Science (SpliTech), 2017 2nd International Multidisciplinary Conference on, IEEE, 2017, pp. 1–6.
[63] P.I.R. Grammatikis, P.G. Sarigiannidis, I.D. Moscholios, Securing the internet of things: challenges, threats and solutions, Internet Things (2018).
[64] O. Salman, I. Elhajj, A. Chehab, A. Kayssi, IoT survey: an SDN and FOG computing perspective, Comput. Netw. 143 (2018) 221–246.
[65] C. Salinesi, R. Mazo, O. Djebbi, D. Diaz, A. Lora-Michiels, Constraints: the core of product line engineering, in: Research Challenges in Information Science (RCIS), 2011 Fifth International Conference on, IEEE, 2011, pp. 1–10.
[66] K. Feichtinger, Y. Nakano, K. Fukushima, S. Kiyomoto, Enhancing the security of over-the-air-activation of LoRaWAN using a hybrid cryptosystem, Int. J. Comput. Sci.Netw. Secur. 18 (2) (2018) 1–9.
[67] J. Han, J. Wang, An enhanced key management scheme for LoRaWAN, Cryptography 2 (4) (2018), doi:10.3390/cryptography2040034.
[68] B. Oniga, V. Dadarlat, E.D. Poorter, A. Munteanu, Analysis, design and implementation of secure LoRaWANsensor networks, in: 2017 13th IEEE International Conference on Intelligent Computer Communication and Processing (ICCP), 2017, pp. 421–428, doi:10.1109/ICCP.2017.8117042.
[69] S. Tomasin, S. Zulian, L. Vangelista, Security analysis of LoRaWAN join procedure for internet of things networks, in: Wireless Communications and Networking Conference Workshops (WCNCW), 2017 IEEE, IEEE, 2017, pp. 1–6.
[70] J. Zhang, A. Marshall, L. Hanzo, Channel-envelope differencing eliminates secret key correlation: LoRa-based key generation in low power wide area networks, IEEE Trans. Veh. Technol. 67 (12) (2018) 12462–12466, doi:10.1109/TVT.2018.2877201.
[71] W. Xu, S. Jha, W. Hu, Exploring the feasibility of physical layer key generation for LoRaWAN, in: 2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/ 12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE), 2018, pp. 231–236, doi:10.1109/TrustCom/BigDataSE.2018.00044.
[72] J. Han, J. Wang, An enhanced key management scheme for LoRaWAN, Cryptography 2 (4) (2018) 34.

[73] R. Sanchez-Iborra, J. Sánchez-Gómez, S. Pérez, P.J. Fernández, J. Santa, J.L. Hernández-Ramos, A.F. Skarmeta, Enhancing LoRaWAN security through a lightweight and authenticated key management approach, Sensors 18 (6) (2018) 1833.
[74] T.C. Dönmez, E. Nigussie, Key management through delegation for LoRaWAN based healthcare monitoring systems, in: 2019 13th International Symposium on Medical Information and Communication Technology (ISMICT), IEEE, 2019, pp. 1–6.
[75] J. Xing, L. Hou, K. Zhang, K. Zheng, An improved secure key management scheme for LoRa system, in: 2019 IEEE 19th International Conference on Communication Technology (ICCT), IEEE, 2019, pp. 296–301.
[76] X. Chen, J. Wang, L. Wang, A fast session key generation scheme for LoRaWAN, in: 2019 Australian & New Zealand Control Conference (ANZCC), IEEE, 2019, pp. 63–66.
[77] T.C. Dönmez, E. Nigussie, Security of LoRaWAN v1. 1 in backward compatibility scenarios, Procedia Comput. Sci. 134 (2018) 51–58.
[78] D. Bui, D. Puschini, S. Bacles-Min, E. Beigné, X. Tran, Aes datapath optimization strategies for low-power low-energy multisecurity-level internet-of-things applications, IEEE Trans. Very Large Scale Integr. (VLSI) Syst. 25 (12) (2017) 3281–3290, doi:10.1109/TVLSI.2017.2716386.
[79] R. Schaupp, Security in LoRaWANapplications – LpWAN LoRaWAN IoT simplified, 2018, (https://smartmakers.io/en/security-in-lorawan-applications/).
[80] R. Miora, Key management with a trusted third party using LoRaWAN protocol (2018).
[81] S. Naoui, M.E. Elhdhili, L.A. Saidane, Enhancing the security of the iot LoRaWANarchitecture, in: Performance Evaluation and Modeling in Wired and Wireless Networks (PEMWN), International Conference on, IEEE, 2016, pp. 1–7.
[82] I. Butun, N. Pereira, M. Gidlund, Security risk analysis of LoRaWAN and future directions, Future Internet 11 (1) (2019) 3.
[83] W. Sung, H. Ahn, J. Kim, S. Choi, Protecting end-device from replay attack on LoRaWAN, in: 2018 20th International Conference on Advanced Communication Technology (ICACT), 2018, pp. 167–171, doi:10.23919/ICACT.2018.8323684.
[84] S. Na, D. Hwang, W. Shin, K.-H. Kim, Scenario and countermeasure for replay attack using join request messages in LoRaWAN, in: Information Networking (ICOIN), 2017 International Conference on, IEEE, 2017, pp. 718–720.
[85] J. Kim, J. Song, A simple and efficient replay attack prevention scheme for LoRaWAN, in: Proceedings of the 2017 the 7th International Conference on Communication and Network Security, in: ICCNS 2017, ACM, New York, NY, USA, 2017, pp. 32–36, doi:10.1145/3163058.3163064.
[86] E.v. Es, H. Vranken, A. Hommersom, Denial-of-service attacks on LoRaWAN(2018).
[87] A. Martínez, U. Zurutuza, R. Uribeetxeberria, M. Fernández, J. Lizarraga, A. Serna, I. Vélez, Beacon frame spoofing attack detection in ieee 802.11 networks, in: 2008 Third International Conference on Availability, Reliability and Security, IEEE, 2008, pp. 520–525.
[88] E. Aras, G.S. Ramachandran, P. Lawrence, D. Hughes, Exploring the security vulnerabilities of LoRa, in: 2017 3rd IEEE International Conference on Cybernetics (CYBCONF), 2017, pp. 1–6, doi:10.1109/CYBConf.2017.7985777.
[89] S.M. Danish, A. Nasir, H.K. Qureshi, A.B. Ashfaq, S. Mumtaz, J. Rodriguez, Network intrusion detection system for jamming attack in LoRaWAN join procedure, in: 2018 IEEE International Conference on Communications (ICC), 2018, pp. 1–6, doi:10.1109/ICC.2018.8422721.
[90] Z. Feng, C. Hua, Machine learning-based RF jamming detection in wireless networks, in: 2018 Third International Conference on Security of Smart Cities, Industrial Control System and Communications (SSIC), IEEE, 2018, pp. 1–6.
[91] K.-L. Tsai, Y.-L. Huang, F.-Y. Leu, I. You, Y.-L. Huang, C.-H. Tsai, AES-128 based secure low power communication for LoRaWAN iot environments, IEEE Access 6 (2018) 45325–45334.
[92] F. Kuipers, Security vulnerabilities in LoRaWAN.
[93] T. Mundt, A. Gladisch, S. Rietschel, J. Bauer, J. Goltz, S. Wiedenmann, General security considerations of LoRaWAN version 1.1 infrastructures, in: Proceedings of the 16th ACM International Symposium on Mobility Management and Wireless Access, in: MobiWac'18, ACM, New York, NY, USA, 2018, pp. 118–123, doi:10.1145/3265863.3265882.
[94] X. Ma, Security vulnerabilities in lorawan.pdf(https://cmpe259-spring18-01.courses.soe.ucsc.edu/system/files/attachments/Security%20Vulnerabilities%20in%20LoRaWAN.pdf).
[95] J. Lee, D. Hwang, J. Park, K.-H. Kim, Risk analysis and countermeasure for bit-flipping attack in LoRaWAN, in: Information Networking (ICOIN), 2017 International Conference on, IEEE, 2017, pp. 549–551.
[96] X. Yang, LoRaWAN: Vulnerability Analysis and Practical Exploitation, Delft University of Technology, 2017 Master of Science.
[97] F.L. Coman, K.M. Malarski, M.N. Petersen, S. Ruepp, Security issues in internet of things: vulnerability analysis of LoRaWAN, sigfox and nb-IoT, in: 2019 Global IoT Summit (GIoTS), IEEE, 2019, pp. 1–6.
[98] S. Chacko, M.D. Job, et al., Security mechanisms and vulnerabilities in LpWAN, in: IOP Conf. Ser. Mater. Sci. Eng, 396, 2018.
[99] M. Ingham, J. Marchang, D. Bhowmik, IoT security vulnerabilities and predictive signal jamming attack analysis in LoRaWAN, IET Inf. Secur. (2020).
[100] J. Kim, J. Song, A dual key-based activation scheme for secure LoRaWAN, Wirel. Commun. Mob. Comput. 2017 (2017).
[101] T. Voigt, M. Bor, U. Roedig, J. Alonso, Mitigating inter-network interference in LoRa networks, arXiv:1611.00688. (2016).
[102] S.-Y. Gao, X.-H. Li, M.-D. Ma, A malicious behavior awareness and defense countermeasure based on LoRaWAN protocol, Sensors 19 (23) (2019) 5122.
[103] N.A. Gunathilake, W.J. Buchanan, R. Asif, Next generation lightweight cryptography for smart IoT devices:: implementation, challenges and applications, in: 2019 IEEE 5th World Forum on Internet of Things (WF-IoT), IEEE, 2019, pp. 707–710.
[104] B. Reynders, Q. Wang, P. Tuset-Peiro, X. Vilajosana, S. Pollin, Improving reliability and scalability of LoRaWANs through lightweight scheduling, IEEE Internet Things J. 5 (3) (2018) 1830–1842.
[105] K.-L. Tsai, F.-Y. Leu, I. You, S.-W. Chang, S.-J. Hu, H. Park, Low-power AES data encryption architecture for a LoRaWAN, IEEE Access 7 (2019) 146348–146357.