НАУЧНО-ТЕХНИЧЕСКИЙ ВЕСТНИК ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ
сентябрь–октябрь 2025 Том 25 № 5 http://ntv.ifmo.ru/
SCIENTIFIC AND TECHNICAL JOURNAL OF INFORMATION TECHNOLOGIES, MECHANICS AND OPTICS
September–October 2025 Vol. 25 No 5 http://ntv.ifmo.ru/en/
ISSN 2226-1494 (print) ISSN 2500-0373 (online)

doi: 10.17586/2226-1494-2025-25-5-923-932

Experimental results of using AES-128 in LoRaWAN

Abdelouahab Nouar[1], Mounir Tahar Abbes[2]✉, Selma Boumerdassi[3], Mostefa Chaib[4]

[1,4] Hassiba Ben Bouali University (UHBC), LMA Laboratory, Chlef, 02010, Algeria
[2] Hassiba Ben Bouali University (UHBC), Chlef, 02010, Algeria
[3] Conservatoire National des Arts et Metiers (CNAM), Paris, 75141, France

[1] a.nouar@univ-chlef.dz, https://orcid.org/0009-0001-3355-1912
[2] m.taharabbes@univ-chlef.dz✉, https://orcid.org/0000-0001-5132-2366
[3] selma.boumerdassi@inria.fr, https://orcid.org/0000-0003-2603-2433
[4] m.chaib@univ-chlef.dz, https://orcid.org/0000-0001-9137-9527

Abstract

In the Internet of Things (IoT), Low Power Wide Area Networks (LPWAN) technologies have been obtaining considerable attention. Long-Range Wide-Area Networks (LoRaWAN) was created by the Long Range (LoRa) Alliance as an open standard operating over the unlicensed band. Its advantages include a large coverage area, low power consumption, and inexpensive transceiver chips. The standard of LoRaWAN encryption uses a 128-bit symmetric algorithm called Advanced Encryption Standard (AES). This standard secures communication and entities which are beneficial for resource-constrained devices on the IoT for efficient communication and security. The security problems with LoRa networks and devices remain an important challenge considering the technology large deployment for numerous applications. Even though LoRaWAN network architecture and security have been enhanced by the LoRa Alliance, the most recent version still has some weaknesses such as its susceptibility to attacks. Many studies and researchers have indicated that LoRaWAN versions 1.0 and 1.1 have security risks and vulnerabilities. This research proposes a method to construct and integrate cryptographic algorithms (AES-128) within widely utilized wireless Network Server Simulators NS-3. This module aims to increase the security of data in LoRa networks by protecting critical information from unauthorized access. Consequently, implementing the AES-128 encryption algorithm within the NS-3 simulator will benefit the scientific community greatly. This will enable an examination of the impact of various security measures on network performance metrics, including latency, overhead, energy consumption, throughput, and packet size.

Keywords

LoRaWAN, cryptography, LoRa, AES-128, security, NS-3, IoT

For citation: Nouar A., Tahar Abbes M., Boumerdassi S., Chaib M. Experimental results of using AES-128 in LoRaWAN. *Scientific and Technical Journal of Information Technologies, Mechanics and Optics*, 2025, vol. 25, no. 5, pp. 923–932. doi: 10.17586/2226-1494-2025-25-5-923-932

УДК 004.056.55

Экспериментальные результаты использования AES-128 в LoRaWAN

Абделуахаб Нуар[1], Мунир Тахар Аббес[2]✉, Сельма Бумердасси[4], Мостефа Хаиб[4]

[1,4] Университет Асиба Бенбуали Лаборатория ЛМА, Шлеф, 02010, Алжир
[2] Университет Асиба Бенбуали, Шлеф, 02010, Алжир
[3] Национальная консерватория искусств и ремесел, Париж, 75141, Франция

[1] a.nouar@univ-chlef.dz, https://orcid.org/0009-0001-3355-1912
[2] m.taharabbes@univ-chlef.dz✉, https://orcid.org/0000-0001-5132-2366
[3] selma.boumerdassi@inria.fr, https://orcid.org/0000-0003-2603-2433
[4] m.chaib@univ-chlef.dz, https://orcid.org/0000-0001-9137-9527

Аннотация

Технология Low Power Wide Area Networks (LPWAN) привлекает значительное внимание в Интернете вещей (IoT). Long-Range Wide-Area Networks (LoRaWAN) создан компанией Long Range (LoRa) как открытый нелицензионный стандарт. Его преимущества включают большую зону покрытия, низкое энергопотребление и недорогие чипы приемопередатчиков. Стандарт шифрования LoRaWAN использует 128-битный симметричный алгоритм Advanced Encryption Standard (AES). Этот стандарт защищает связь и объекты, что выгодно для устройств с ограниченными ресурсами в IoT для эффективной связи и безопасности. Проблемы безопасности сетей и устройств LoRa остаются важной задачей, учитывая широкое распространение этой технологии в многочисленных приложениях. Несмотря на то, что создатели LoRa улучшили архитектуру и безопасность сети LoRaWAN, последняя версия все еще имеет некоторые недостатки, такие как уязвимость к атакам. Многочисленные исследования показали, что версии LoRaWAN 1.0 и 1.1 содержат угрозы безопасности и уязвимости. В работе предлагается метод построения и интеграции криптографических алгоритмов (AES-128) в широко используемых симуляторах беспроводных сетей NS-3. Целью данного средства является повышение безопасности данных в сетях LoRaWAN путем защиты критически важной информации от несанкционированного доступа. Внедрение алгоритма шифрования AES-128 в симулятор NS-3 позволит изучить влияние различных мер безопасности на показатели производительности сети, включая задержку, накладные расходы, энергопотребление, пропускную способность и размер пакета.

Ключевые слова
LoRaWAN, криптография, LoRa, AES-128, security, NS-3, IoT

Ссылка для цитирования: Нуар А., Тахар Аббес М., Бумердасси С., Хаиб М. Экспериментальные результаты использования AES-128 в LoRaWAN // Научно-технический вестник информационных технологий, механики и оптики. 2025. Т. 25, № 5. С. 923–932 (на англ. яз.). doi: 10.17586/2226-1494-2025-25-5-923-932

Introduction

Long-Range Wide-Area Networks (LoRaWAN) is a Low-Power Wide Area Network (LPWAN) protocol that uses low-power algorithm to send data over long distances. LoRaWAN utilizes the unlicensed wireless spectrum, meaning anyone can use it without government permission. Multiple End Devices (EDs) communicate with a central gateway using a star topology [1]. The gateway then transmits the data packets to a Network Server (NS) that manages all devices and processes the data.

LoRaWAN is widely used in smart city applications [1], with many cities around the world, using the technology for various use cases, such as intelligent lighting, waste management, and air quality monitoring. Overall, these statistics highlight the growing popularity and adoption of LoRaWAN and its suitability for a wide range of Internet of Things (IoT) applications [2].

In LoRaWAN, cryptography plays a critical role in securing data transmission. One of the cryptographic methods used in LoRaWAN is Advanced Encryption Standard (AES) 128 bits. The latest version of LoRaWAN v.1.1 has provided a security framework that includes data privacy protection, data integrity control, device authentication, and key management [3]. The LoRaWAN protocol uses AES-128 algorithm in the core encryption mechanism to guarantee confidentiality, integrity, and authentication through two layers:

- Network Layer Encryption: This layer uses a Network Session Key (NwkSKey) to assure end-to-end device connection with the network server. It protects the integrity of the message to ensure that it comes from a legitimate device.
- Application Layer: This layer assures the confidentiality of data from the ED to the Application Server (AS), by introducing an Application Session Key called (AppSKey). It provides encryption to protect the payload, ensuring that only the AS can decrypt the actual message content.

In light of the latest development in resource-constrained IoT devices, various versions of LoRaWAN have been published to improve its performance in terms of security, scalability, and real-time long-range communication. The following summarizes the evolution of the LoRaWAN specifications, including the year of release and major changes:

- LoRaWAN version 1.0 (January 2015)[1]: first approval of LoRaWAN 1.0;
- LoRaWAN version 1.0.1 (February 2016) [4]: This update added a new frequency plan, modified some MAC-layer instructions;
- LoRaWAN version 1.0.2 (July 2016)[2]: Many problems were resolved, and others MAC commands were created in this version;
- LoRaWAN version 1.1 (October 2017)[3]: With the addition of a new server named Join Server, this significant revision introduced a new architecture. It also brought many improvements to the security mechanism, including support for roaming handover and the use of two root keys rather than one to derive the session security keys. Finally, it added numerous countermeasures to mitigate some of the vulnerabilities that had been reported in earlier versions;
- LoRaWAN version 1.0.3 (July 2018)[4]: In this revision a little number of MAC commands for class A devices are added;
- LoRaWAN version 1.0.4 (October 2020) [5]: This small update for v1.0.3 clarified various issues on Adaptive Data Rate (ADR) behavior, FCnt usage and behaviors, joining channel selection process, and retransmission backoff. This release includes two significant security-related changes that observed: first, DevNonce

[1] L. Specification, “LoRaWAN specification v1. 0”, San Francisco, CA, USA, 2015. Available at: https://lora-alliance.org/resource_hub/lorawan-specification-v1-0 (accessed: 17.04.2024).

[2] L. Alliance, “LoRaWAN specification v1. 0.2”, Date of retrieval, 2016. Available at: https://resources.lora-alliance.org/technical-specifications-v1-0-2/ (accessed: 11.11.2024).

[3] LoRaWAN specification v1.1. Available at: https://resources.lora-alliance.org/technical-specifications/ (accessed: 29.11.2023).

[4] L. Specification, “LoRaWAN specification v1.0.3”, San Francisco, CA, USA, 2018. Available at: https://lora-alliance.org/resource_hub/lorawan-specification-v1-0-3 (accessed: 12.03.2024).

generation is now incremental rather than random, and second, JoinEUI and AppNonce have been substituted for AppEUI and AppNonce.

This research focuses on the implementation of AES-128-bit cryptography under the NS-3 simulator; by doing so, it will improve the realism and security aspects of simulations, especially when the work will be on IoT or wireless network research. The main idea is to simulate secure communication within networks and to study the effects of encryption on network performance.

The rest of the paper is organized as follows: first, similar works are presented, then the activation methods and key derivation are discussed. Some basic ideas and an AES overview are presented in the following section. Next section illustrates the implementation of the AES-128 algorithm using NS-3. The results and discussion of the impact of cryptography on LoRaWAN performance and the concluding remarks are presented in the final sections.

Similar Works

AES-128 is considered to be the block cipher of choice for many applications in the future. However, that does not mean that the communication protocol is secure. Butun et al. [6] conclude that there are multiple attack vectors to LoRaWAN and that the security is dependent on the implementation. The authors state that there are a few critical mechanisms in the implementation that need to be considered.

There has been ample work on LoRaWAN. The literature shows that LoRaWAN version 1.0 has some security vulnerabilities. Many of these vulnerabilities have since been fixed in version 1.1 and have improved the security of LoRaWAN.

In [7], the authors proposed the use of PHYSEC-based key management which is based on physical layer security in LoRaWAN. The authors research showed that it can be a good solution to current key management solutions while having low energy consumption costs when compared to other key management methods.

The work presented in [8] elaborated an experimental performance analysis of the Over-the-Air-Activation (OTAA) procedure using a real LoRaWAN deployment in the field, with the objective of analyzing the delay in activation and energy consumption on a large-scale LoRaWAN. The authors came to the conclusion that high network traffic is a big problem in OTAA activation. Long activation delays occurred (50 % of the devices took more than 2 hours to activate). There were also a high number of packet retransmissions. Three main factors affect the performance of the OTAA procedure: collisions; retransmissions; and the communication request work cycle.

Another proposed secure LoRaWAN backend [9] Server Session Key Generation (S2KG), which uses it to generate network session keys.

As an example, the vulnerability of missing beacon authentication in Class B mechanism and the ADR spoofing attack are controlled using ChirpOTLE [10] by updating the LoRaWAN protocol.

Another research [11] presents a solution based on hybridization between GNU Radio and software-defined radio; this architecture is without LoRaWAN transceivers.

The authors in [12] propose Low-Power AES Data Encryption Architecture (LPADA) for LoRaWAN in physical layer based on different hardware construction. The core of this solution is composed from a low-energy lookup table to complete AES substitution and to optimize the energy consumption in several rounds.

Another interesting contribution [13] illustrates the high-level use with various AES key sizes alongside differing payload dimensions. The findings indicate that the costs associated with delay and energy consumption are moderate, and employing longer key sizes is a viable approach to enhance security.

Naoui et al. [14] assessed the security of the LoRaWAN 1.0 protocol. The authors concluded that the LoRaWAN protocol is susceptible to two potential assaults. The first one is the parameter DevNonce, this is a 16-bit counter that is increased by one with each join request, starting at 0 when the ED is first switched on. The attacker can use replay attacks when the DevNonce is not encrypted. Also, AppNonce is generated when the server receives join-request message from the EDs. After that the AppNonce is passed to both the ED and AS for authentication. In the next message, an attacker can send the ED the relevant join acceptance message which it initiated. The authors designed a trusted third-party computer which is utilized to dispatch the session key for NSs and ASs. The trusted third-party computer creates a timeline, and the NS stores the timeline when it receives a join-request message so as to prevent a replay attack.

Jakub et al. [15] aimed to integrate the fog computing concept into LoRaWAN. The basic tenet of this paradigm is to increase efficiency for massive volumes of data by putting data processing and storage closer to the EDs. In this regard, the authors presented three fog computing-based IoT network architecture. To determine the best architecture, each of the suggested architectures was simulated and compared in terms of service time. By reducing latency, bandwidth, and efficiency, fog computing offers several advantages to IoT sectors. But security concerns must not be overlooked.

According to Qadir et al. [16], EDs that are located on the network edge is a major target for cyber-attackers. In light of safe key management, they therefore provide a remedy known as the Key Generation and Distribution (KGD) method which lessens cyber attacks. There are three steps involved in the KGD algorithm. Initially, it uses a cryptographically safe deterministic random bit generator approach to produce the secret keys. The Elliptic-Curve Diffie-Hellman technique is then used to exchange the produced keys between the ED and Join Server. The Elliptic Curve Digital Signature Algorithm, a key authentication procedure, is subsequently taken into consideration to confirm if the keys were transferred to the authorized parties. The results demonstrate that their suggested KGD has authentication, integrity, and transmission secrecy.

The authors in [17] propose a novel security protocol that reduces the total time required for key creation and renewal. The technique initiates with random pairing locations and utilizes Lagrange interpolation, effectively decreasing the message count while generating a group key. The chain of hashes concept facilitates the renewal of a group certificate through a single message, thereby negating the necessity for additional message exchanges. The evaluation results indicate that this strategy significantly decreases both the volume of messages and the configuration time relative to prior methods. This enhancement increases the efficiency of secure communication and fortifies overall security by reducing potential vulnerabilities.

The authors in [18] introduce FLoRa, a technique for key generation at the physical layer. The initial key is generated using an adaptive multibit quantization method which enhances the initiation process influences to the rate of bit generation. This minimizes key reconciliation duration and enhances the recovery rate. The method utilizes a robust algorithm to assess channel conditions for the optimization of the key generation process.

To the best of our knowledge, there is a lack of existing research regarding the implementation of the AES-128 standard under network simulators. Furthermore, not much research has been done on how these security paradigms affect ED energy usage.

Activation Methods and Key Derivation

LoRaWAN supports two distinct methods of activating devices: OTAA and Activation by Personalization [19]. Both of these methods are interchangeable. Independently, the activation mechanism for LoRaWAN is explained in the two following Fig. 1, and Fig. 2.

Step 1: The ED consistently initiates the joining procedure in all instances. A join-request message is sent to the network by the final device intending to join. This message includes critical information regarding the device identity and capabilities. This mechanism preserves network integrity by ensuring that each join request is novel and not a repetition of previous attempts (referred to as a replay attack [20]).

The *AppKey* used to calculate the Message Integrity Code (MIC) by using all the fields in the join-request message.

The join-request message is then updated with the computed MIC and is not encrypted, nor is the AppKey transmitted, as illustrated in Fig. 2.

Step 2: The message requesting to join the network is processed and generated by the server (NwkSKey and AppSKey).

Step 3: As part of the normal down-link mode, the NS gives the encrypted join-accept data to the ED, and the NS does not accept the Join-request message, the ED will not receive any response from the server.

Step 4: The role of NS is to maintain the NwkSKey and also distribute AppSKey to the AS.

Step 5: The join-accept information is deciphered by means of the ED via the AES encryption method. Each of the two keys are produced by the ED, the AppSKey and the NwkSKey, using the AppKey and AppNonce [21].

AES Overview

AES employs a symmetric block cipher scheme and offers key lengths of 128, 192, and 256 bits for encryption and decryption [22]; these key lengths determine the number of rounds in the encryption process to meet different environmental needs, as is illustrated in Table 1.

Implementation of AES-128 Algorithm under NS-3

Despite the enormous research and the various works carried out by researchers and labs, in all the literature, according to our knowledge, we do not find the deployment

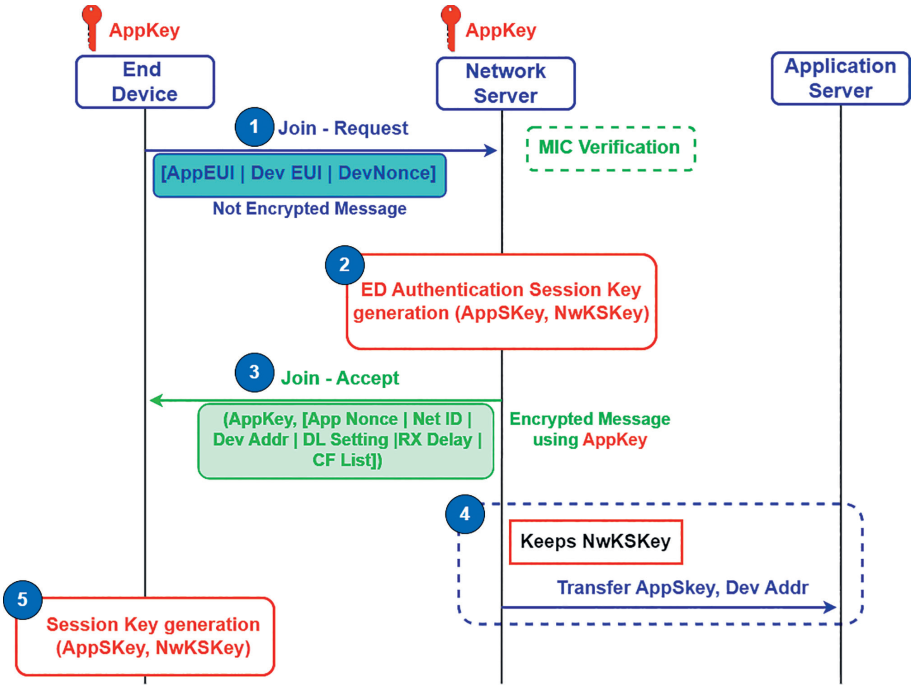

AppKey
End Device
AppKey
Network Server
Application Server
1
Join - Request
MIC Verification
[AppEUI | Dev EUI | DevNonce]
Not Encrypted Message
2
ED Authentication Session Key generation (AppSKey, NwKSKey)
3
Join - Accept
(AppKey, [App Nonce | Net ID | Dev Addr | DL Setting |RX Delay | CF List])
Encrypted Message using AppKey
4
Keeps NwKSKey
Transfer AppSkey, Dev Addr
5
Session Key generation (AppSKey, NwKSKey)

Fig. 1. OTAA message flow in LoRaWAN Network

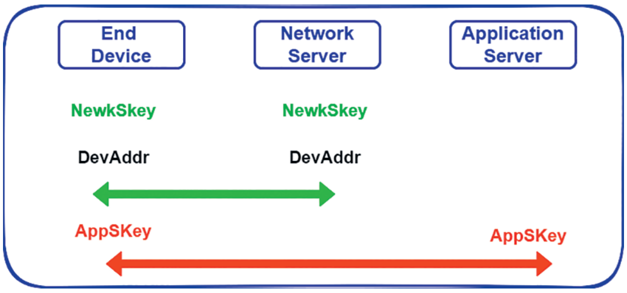


Fig. 2. Pre-sharing DevAddr and session keys for Activation by Personalization

of cryptography using the NS-3 simulator. So, the objective is to implement the AES algorithm under the NS-3 simulator; for this, it will work as follows:

— Modify the application layer code to add encryption;

— Encrypt data before calling the *Send()* function, and decrypt it after receiving the packet.

As illustrated in Fig. 3, at the physical layer, the packet will be split and take just the payload in plaintext, then encrypt only the payload according to the specification [9], using the AES-128 encryption algorithm based on the library accessible via the following link[1], once the ciphertext is successfully received by the receiver (by the NS), it will be decrypted with the same encryption key, and finally the decrypted uplink message will be displayed in plaintext.

The MAC layer is responsible for transmitting the packets, and the helpers are charged with initializing the configuration parameters of each scenario, including the kind of encryption (AES-128, AES-192, and AES-256 [23]). The encryption keys are configured in the *PeriodicSenderHelper*.

Table 1. Key sizes in AES [22]

Parameter	AES-128	AES-192	AES-256
Rounds, numbers	10	12	14
Key sizes, bit	128	192	256
Data block lengths, bit	128	128	128

This operation calls the *setAesKey()* function and instantiates an object of the *PeriodicSender class* which is responsible for encryption. Through the *encrypt()* function, the encryption is done before the packet will be sent.

The *PeriodicSender* class even implements the *Send Packet()* function to call an object of the *LorawanMac class* which implements another *Send()* function whose role is to send messages as shown in Fig. 4.

The *encrypt()* function is between timespec begin and timespec end to calculate CPU time; the time needed by the CPU to execute the encryption as shown in Fig. 5.

On the other side, in the NS side, the same steps are done for the decryption once the message is received. The *NetworkServerHelper* class initializes the shared parameters (Symmetric Encryption) by the *SetDecrypt()* function and calls an object of the NetworkServer model class to do the necessary, as shown in the Fig. 6:

— Receive the message by the *Receive()* function;

— Remove the headers;

— Decrypt the message via the *Decrypt()* function.

The *decrypt()* function aims to decrypt the message received using the same pre-shared encryption key, as illustrated in Fig. 7.

Results and Discussion

To measure the energy consumption, Packet Delivery Rate (PDR) and Time on Air (ToA) induced by the cryptographic primitives used in the LoRaWAN stack according to various packet sizes based on the parameters

[1] Tiny AES in C. Available at: https://github.com/kokke/tiny-AES-c (accessed: 08.06.2024).

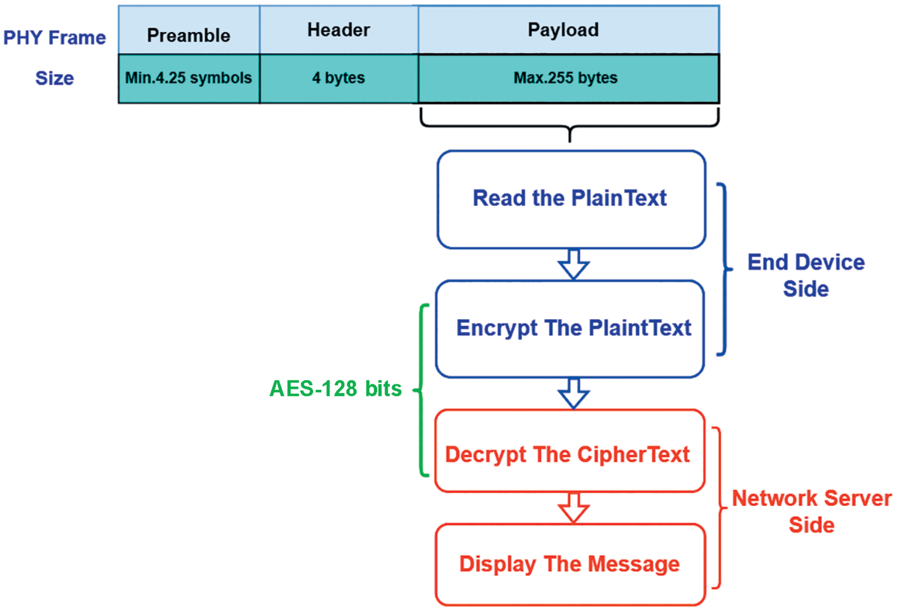


Fig. 3. AES-128 Flow Implementation

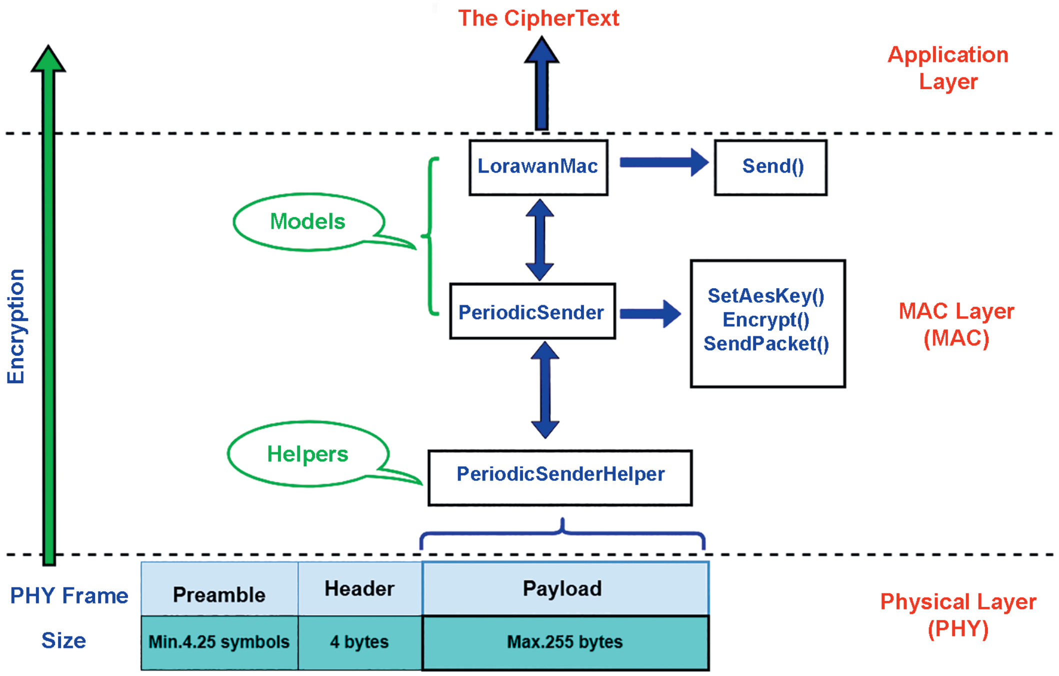
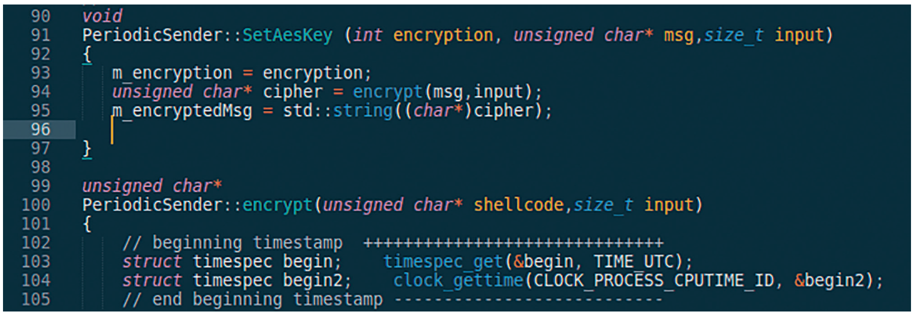


Fig. 4. Encryption process



```
void
PeriodicSender::SetAesKey (int encryption, unsigned char* msg,size_t input)
{
    m_encryption = encryption;
    unsigned char* cipher = encrypt(msg,input);
    m_encryptedMsg = std::string((char*)cipher);
    
}

unsigned char*
PeriodicSender::encrypt(unsigned char* shellcode,size_t input)
{
     // beginning timestamp  ++++++++++++++++++++++++++++++
     struct timespec begin;    timespec_get(&begin, TIME_UTC);
     struct timespec begin2;    clock_gettime(CLOCK_PROCESS_CPUTIME_ID, &begin2);
     // end beginning timestamp --------------------------
```

Fig. 5. encrypt() function

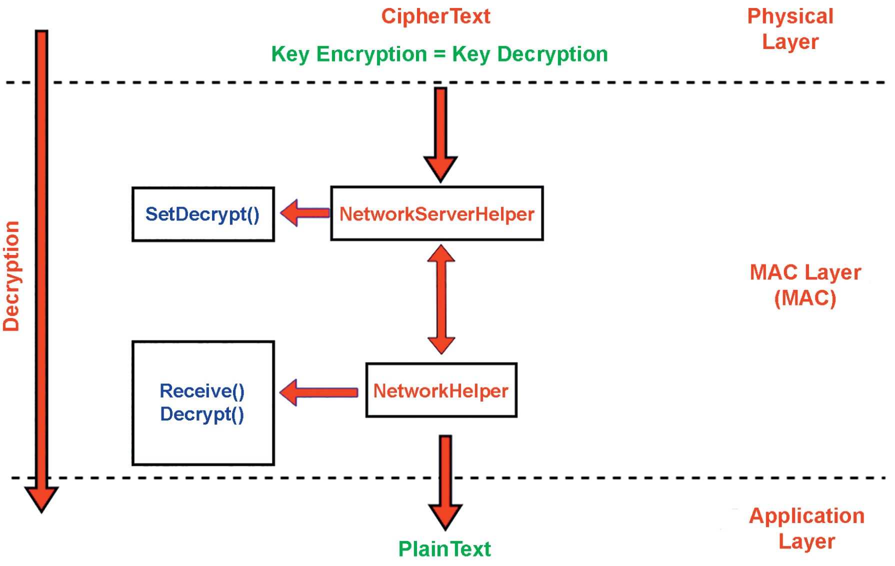
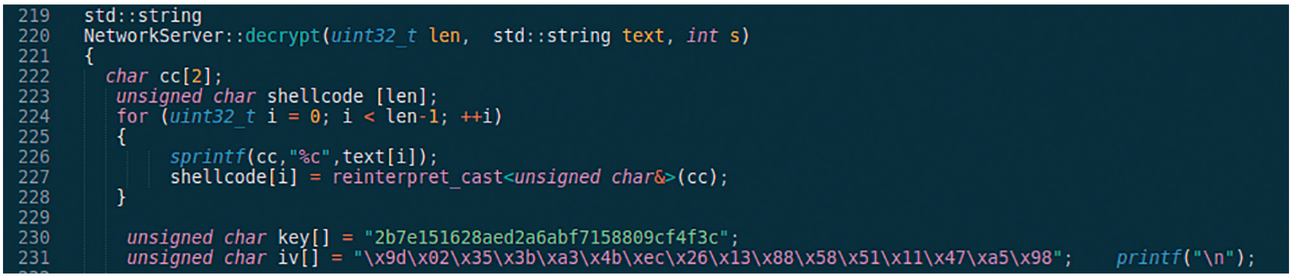


Fig. 6. Decryption process



```
std::string
NetworkServer::decrypt(uint32_t len,  std::string text, int s)
{
  char cc[2];
   unsigned char shellcode [len];
   for (uint32_t i = 0; i < len-1; ++i)
   {
        sprintf(cc,"%c",text[i]);
        shellcode[i] = reinterpret_cast<unsigned char&>(cc);
   }

    unsigned char key[] = "2b7e151628aed2a6abf7158809cf4f3c";
    unsigned char iv[] = "\x9d\x02\x35\x3b\xa3\x4b\xec\x26\x13\x88\x58\x51\x11\x47\xa5\x98";    printf("\n");
```

Fig. 7. decrypt() function

Table. 2. Simulation Parameters

Parameter	Value	Unit
N of Nodes	50	—
Radius	20	m
Period	5	s
Packet Size	12, 24, 32, 64, 128, 192, 216	byte
GateWay (GW)	1	—
Simulation Time	3,600	s
Energy Initial	200	mAh
Battery PD2032	2,664	J
	3.7	V
Simulator	NS-3 (Version 3.35)	—
Operating System	Ubuntu 24.4 64 bit	—

used in Table 2, successfully gathering data from multiple experiments. This information allowed us to analyze the performance metrics comprehensively, providing insights into the efficiency of the cryptographic methods in relation to varying packet sizes and their impact on overall network reliability.

Consider a scenario that sends a packet of 12-byte every 5 s with the maximum transmission power (+14 dBm). For a simulation time of 3,600 s, the PD2032 battery model has a capacity of 4,000 mAh and 2,664 J, with the EDs randomly distributed in a radius of 20 m around a single (01) GW, as the same parameters used in experience [24]. In order to demonstrate the influence of AES-128 cryptography, the difference with and without AES-128 in value is shown in Table 3.

Fig. 8 shows the ToA for different payload lengths (in bytes), using a radius equal to 20 m and a single GW. In LoRaWAN, ToA defines the elapsed time for a LoRaWAN packet between the ED and GW. ToA for different configurations for each packet can be calculated using a formula provided in LoRaWAN specifications [9]. As expected, payload length plays an important role for ToA.

The ToA increases with increasing packet size with and without AES-128 encryption; there is a slight difference between the two histograms.

Since ToA is directly related to the amount of energy, a node needs to spend to transmit the data packet, it is important to determine the battery life of a node. Simulate different scenarios by increasing the packet size ∈ {12, 32, 64, 128, 192, 216}. Fig. 9 examines the energy remaining of nodes in a network during 1 hour of simulation, focusing on packet sizes, with energy measured in joules. A significant decrease in the remaining energy occurs with increasing the packet size from 64 to 216 byte, highlighting the impact of encryption AES-128, packet size, and communication frequency on energy usage.

In Fig. 10, the PDR is calculated based on the packet size where 7 different packet size values are plotted with and without AES-128. The Table 4 shows the difference in value.

As expected, with the packet size increasing, the PDR also decreases for both histograms, either with or

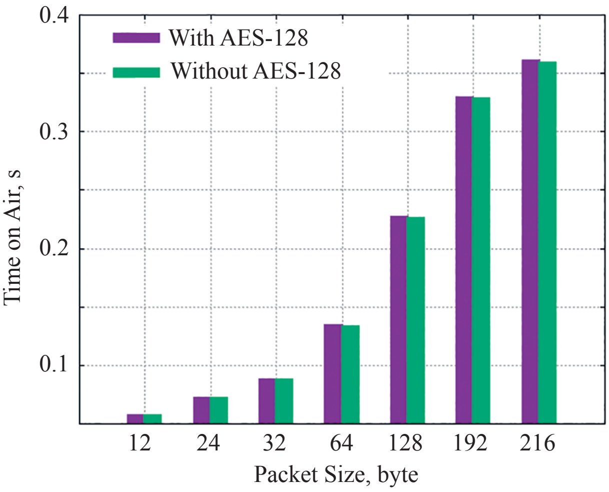


Fig. 8. Time on Air vs. Packet Size

Table 3. Time difference in ToA with and without AES-128, bits

Parameter	12 byte	24 byte	32 byte	64 byte	128 byte	192 byte	216 byte
ToA Without.AES-128	0.0564760	0.0718360	0.0871960	0.133376	0.225536	0.327936	0.358656
ToA With.AES-128	0.0567319	0.0721809	0.0876003	0.133918	0.226560	0.329472	0.360356
Difference, s	0.0002559	0.0003449	0.0004043	0.000542	0.001024	0.001536	0.001700

Table 4. PDR with and without AES-128, bits

Parameter	12 byte	24 byte	32 byte	64 byte	128 byte	192 byte	216 byte
Packet sent with AES-128	30,862	25,300	20,769	13,600	8,050	5,550	5,050
Packet received with AES-128	25,697	19,027	14,724	8,154	3,166	1,665	1,395
Packet sent without AES-128	30,873	25,180	20,840	13,650	8,010	5,750	5,150
Packet received without AES-128	25,697	19,027	14,792	8,192	3,182	1,782	1,442
PDR with AES-128, %	83.26	75.50	70.89	59.95	39.32	30.00	27.62
PDR without AES-128, %	83.23	75.56	70.97	60.01	39.72	30.99	28.00

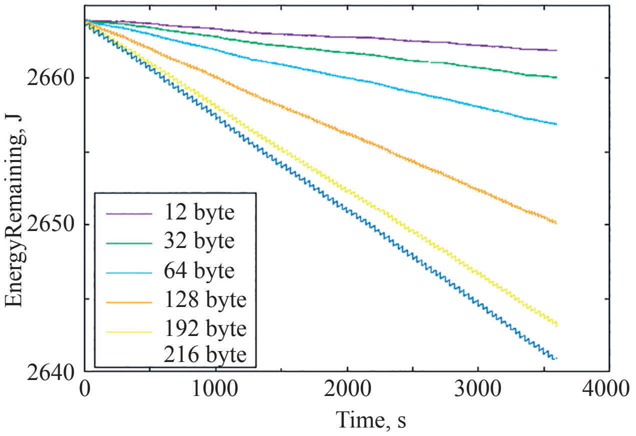


Fig. 9. Energy remaining vs. Time

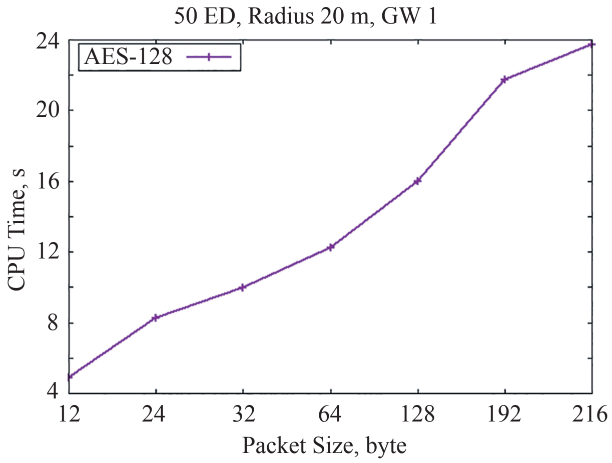


Fig. 11. CPU Time vs. Packet size with AES-128

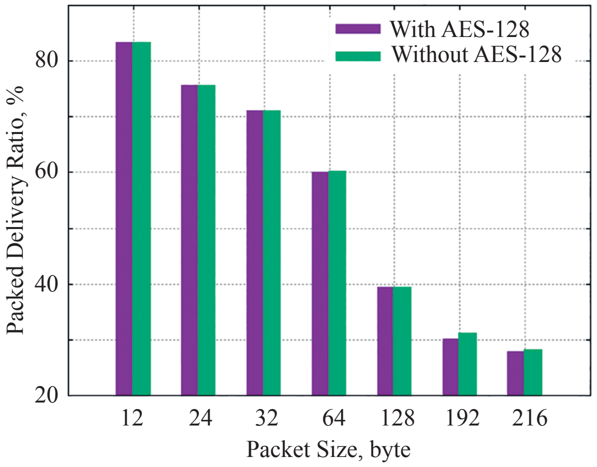


Fig. 10. Packet Delivery Ratio vs. Packet Size

without AES-128 encryption. Notice that for the smallest size of 12 byte with the lowest airtime, almost 83 % PDR is achieved. While for the bigger size of 216 byte, the PDR value drops to only less than 28 % due to the longer packet transmission time, packets are more vulnerable to collisions. In addition, it should be noted that the transmission of packets encrypted with AES-128 impacts the transmission time which is longer than without using AES-128 cryptography and which increases the propensity to collisions and therefore directly impacts the PDR. This confirms the results obtained in Fig. 8 when the packet size increases.

Fig. 11 shows the execution time in seconds compared with the packet size using the AES-128 encryption. There is a linear increase up to the maximum size which took 24 s for a packet of 216 bytes.

Conclusion

In this research, we have introduced and extensively evaluated the implementation of the AES-128 encryption standard for Long-Range Wide-Area Networks (LoRaWAN) using the NS-3 simulator. When simulating such systems, it is important to include encryption to make the simulation results more realistic. This contribution lies in the implementation of AES-128. The results show that when transmitting ciphertext, AES-128 suffers an average delay of about 0.7867 ms. A significant decrease in the remaining energy occurs when the packet size increases from 64 to 216 byte. Even if security has some negative effects on network performance, the trade-off is necessary.

Currently, while AES-128 is the standard in LoRaWAN devices, despite its relatively small key size (128 bits), AES-128 is considered secure and is widely used across various industries. It offers sufficient protection for most LoRaWAN use cases. This work offers the way for the integration of other security modes such as AES-192 and AES-256, expanding the application scope of LoRaWAN.

References

1. Mostefa C., Mounir T.A., Abdelmadjid A.M., Nouar A. Ft-CSMA: A fine-tuned CSMA protocol for LoRa-based networks. *Journal of Communications*, 2024, vol. 19, no. 2, pp. 65–77. https://doi.org/10.12720/jcm.19.2.65-77
2. Umbreen S., Shehzad D., Shafi N., Khan B., Habib U. An energy-efficient mobility-based cluster head selection for lifetime enhancement of wireless sensor networks. *IEEE Access*, 2020, vol. 8, pp. 207779–207793. https://doi.org/10.1109/access.2020.3038031
3. Mostefa C., Abdelouahab N., Mounir T.A., Boumerdassi S., Femmam S., Amel Z.A. Formal validation of ADR protocol in LoRaWAN network using Event-b. *Proc. of the 7th International Conference on Computer, Software and Modeling (ICCSM)*, 2023, pp. 11–15. https://doi.org/10.1109/ICCSM60247.2023.00011
4. Sornin N., Luis M., Eirich T., Kramp T., Hersent O. *LoRaWAN Specification*. V. 1. LoRa Alliance Inc., 2015, 82 p.
5. LoRaWAN® L2 1.0.4 Specification (TS001-1.0.4). *Lora Alliance Technical Committee*, 2020, 90 p.
6. Butun I., Pereira N., Gidlund M. Analysis of LoRaWAN v1.1 security: research paper. *Proc. of the 4th ACM MobiHoc Workshop on Experiences with the Design and Implementation of Smart Objects*, 2018, pp. 1–6. https://doi.org/10.1145/3213299.3213304
7. Andreas W., de la Fuente A.G., Christoph L., Michael K. Physical layer security based key management for LoRaWAN. *arXiv*, 2021, arXiv:2101.02975. https://doi.org/10.48550/arXiv.2101.02975
8. El Fehri C., Baccour N., Berthou P., Kammoun I. Experimental analysis of the over-the-air activation procedure in LoRaWAN. *Proc. of the 17th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, 2021, pp. 30–35. https://doi.org/10.1109/wimob52687.2021.9606301
9. Tsai K.-L., Leu F.-Y., Hung L.-L., Ko C.-Y. Secure session key generation method for LoRaWAN servers. *IEEE Access*, 2020, vol. 8, pp. 54631–54640. https://doi.org/10.1109/ACCESS.2020.2978100
10. Hessel F., Almon L., Alvarez F. ChirpOTLE: a framework for practical LoRaWAN security evaluation. *Proc. of the 13th ACM Conference on Security and Privacy in Wireless and Mobile Networks*, 2020, pp. 306–316. https://doi.org/10.1145/3395351.3399423
11. Pospisil O., Fujdiak R., Mikhaylov K., Ruotsalainen H., Misurec J. Testbed for LoRaWAN security: design and validation through man-in-the-middle attacks study. *Applied Sciences*, 2021, vol. 11, no. 16, pp. 7642. https://doi.org/10.3390/app11167642
12. Tsai K.-L., Leu F.-Y., You I., Chang S.-W., Hu S.-J., Park H. Low-power AES data encryption architecture for a LoRaWAN. *IEEE Access*, 2019, vol. 7, pp. 146348–146357. https://doi.org/10.1109/access.2019.2941972
13. Thaenkaew P., Quoitin B., Meddahi A. Evaluating the cost of beyond AES-128 LoRaWAN security. *Proc. of the International Symposium on Networks, Computers and Communications (ISNCC)*, 2022, pp. 1–6. https://doi.org/10.1109/isncc55209.2022.9851811
14. Naoui S., Elhdhili M.E., Saidane L.A. Trusted third party based key management for enhancing LoRaWAN security. *Proc. of the IEEE/ACS 14th International Conference on Computer Systems and Applications (AICCSA)*, 2017, pp. 1306–1313. https://doi.org/10.1109/AICCSA.2017.73
15. Jalowiczor J., Rozhon J., Voznak M. Study of the efficiency of fog computing in an optimized LoRaWAN cloud architecture. *Sensors*, 2021, vol. 21, no. 9, pp. 3159. https://doi.org/10.3390/s21093159
16. Qadir J., Butun I., Gastaldo P., Aiello O., Caviglia D.D. Mitigating cyber attacks in LoRaWAN via lightweight secure key management scheme. *IEEE Access*, 2023, vol. 11, pp. 68301–68315. https://doi.org/10.1109/ACCESS.2023.3291420
17. Hanna Y., Cebe M., Leon J., Akkaya K. Efficient group key management for resilient operation of LoRaWAN-based smart grid applications. *IEEE Transactions on Control Systems Technology*, 2024, vol. 32, no. 5, pp. 1706–1717. https://doi.org/10.1109/tcst.2024.3378988
18. Han B., Li Y., Wang X., Li H., Huang J. FLoRa: Sequential fuzzy extractor based physical layer key generation for LPWAN. *Future Generation Computer Systems*, 2023, vol. 140, pp. 253–265. https://doi.org/10.1016/j.future.2022.10.018
19. Islam M., Jamil H.M.M., Pranto S.A., Das R.K., Amin A., Khan A. Future industrial applications: exploring LPWAN-driven IoT protocols. *Sensors*, 2024, vol. 24, no. 8, pp. 2509. https://doi.org/10.3390/s24082509

Литература

1. Mostefa C., Mounir T.A., Abdelmadjid A.M., Nouar A. Ft-CSMA: A fine-tuned CSMA protocol for LoRa-based networks // Journal of Communications. 2024. V. 19. N 2. P. 65–77. https://doi.org/10.12720/jcm.19.2.65-77
2. Umbreen S., Shehzad D., Shafi N., Khan B., Habib U. An energy-efficient mobility-based cluster head selection for lifetime enhancement of wireless sensor networks // IEEE Access. 2020. V. 8. P. 207779–207793. https://doi.org/10.1109/access.2020.3038031
3. Mostefa C., Abdelouahab N., Mounir T.A., Boumerdassi S., Femmam S., Amel Z.A. Formal validation of ADR protocol in LoRaWAN network using Event-b // Proc. of the 7th International Conference on Computer, Software and Modeling (ICCSM). 2023. P. 11–15. https://doi.org/10.1109/ICCSM60247.2023.00011
4. Sornin N., Luis M., Eirich T., Kramp T., Hersent O. LoRaWAN Specification. V. 1. LoRa Alliance, Inc. 2015. 82 p.
5. LoRaWAN® L2 1.0.4 Specification (TS001-1.0.4) // Lora Alliance Technical Committee, 2020. 90 p.
6. Butun I., Pereira N., Gidlund M. Analysis of LoRaWAN v1.1 security: research paper // Proc. of the 4th ACM MobiHoc Workshop on Experiences with the Design and Implementation of Smart Objects. 2018. P. 1–6. https://doi.org/10.1145/3213299.3213304
7. Andreas W., de la Fuente A.G., Christoph L., Michael K. Physical layer security based key management for LoRaWAN // arXiv. 2021. arXiv:2101.02975. https://doi.org/10.48550/arXiv.2101.02975
8. El Fehri C., Baccour N., Berthou P., Kammoun I. Experimental analysis of the over-the-air activation procedure in LoRaWAN // Proc. of the 17th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob). 2021. P. 30–35. https://doi.org/10.1109/wimob52687.2021.9606301
9. Tsai K.-L., Leu F.-Y., Hung L.-L., Ko C.-Y. Secure session key generation method for LoRaWAN servers // IEEE Access. 2020. V. 8. P. 54631–54640. https://doi.org/10.1109/ACCESS.2020.2978100
10. Hessel F., Almon L., Alvarez F. ChirpOTLE: a framework for practical LoRaWAN security evaluation // Proc. of the 13th ACM Conference on Security and Privacy in Wireless and Mobile Networks. 2020. P. 306–316. https://doi.org/10.1145/3395351.3399423
11. Pospisil O., Fujdiak R., Mikhaylov K., Ruotsalainen H., Misurec J. Testbed for LoRaWAN security: design and validation through man-in-the-middle attacks study // Applied Sciences. 2021. V. 11. N 16. P. 7642. https://doi.org/10.3390/app11167642
12. Tsai K.-L., Leu F.-Y., You I., Chang S.-W., Hu S.-J., Park H. Low-power AES data encryption architecture for a LoRaWAN // IEEE Access. 2019. V. 7. P. 146348–146357. https://doi.org/10.1109/access.2019.2941972
13. Thaenkaew P., Quoitin B., Meddahi A. Evaluating the cost of beyond AES-128 LoRaWAN security // Proc. of the International Symposium on Networks, Computers and Communications (ISNCC). 2022. P. 1–6. https://doi.org/10.1109/isncc55209.2022.9851811
14. Naoui S., Elhdhili M.E., Saidane L.A. Trusted third party based key management for enhancing LoRaWAN security // Proc. of the IEEE/ACS 14th International Conference on Computer Systems and Applications (AICCSA). 2017. P. 1306–1313. https://doi.org/10.1109/AICCSA.2017.73
15. Jalowiczor J., Rozhon J., Voznak M. Study of the efficiency of fog computing in an optimized LoRaWAN cloud architecture // Sensors. 2021. V. 21. N 9. P. 3159. https://doi.org/10.3390/s21093159
16. Qadir J., Butun I., Gastaldo P., Aiello O., Caviglia D.D. Mitigating cyber attacks in LoRaWAN via lightweight secure key management scheme // IEEE Access. 2023. V. 11. P. 68301–68315. https://doi.org/10.1109/ACCESS.2023.3291420
17. Hanna Y., Cebe M., Leon J., Akkaya K. Efficient group key management for resilient operation of LoRaWAN-based smart grid applications // IEEE Transactions on Control Systems Technology. 2024. V. 32. N 5. P. 1706–1717. https://doi.org/10.1109/tcst.2024.3378988
18. Han B., Li Y., Wang X., Li H., Huang J. FLoRa: Sequential fuzzy extractor based physical layer key generation for LPWAN // Future Generation Computer Systems. 2023. V. 140. P. 253–265. https://doi.org/10.1016/j.future.2022.10.018
19. Islam M., Jamil H.M.M., Pranto S.A., Das R.K., Amin A., Khan A. Future industrial applications: exploring LPWAN-driven IoT protocols // Sensors. 2024. V. 24. N 8. P. 2509. https://doi.org/10.3390/s24082509

20. Na S., Hwang D., Shin W., Kim K.-H. Scenario and countermeasure for replay attack using join request messages in LoRaWAN. *Proc. of the International Conference on Information Networking (ICOIN)*, 2017, pp. 718–720. https://doi.org/10.1109/ICOIN.2017.7899580
21. Kang J.-M., Lim D.-W. On the quasi-orthogonality of LoRa modulation. *IEEE Internet of Things Journal*, 2023, vol. 10, no. 14, pp. 12366–12378. https://doi.org/10.1109/jiot.2023.3245885
22. Tsai K.-L., Huang Y.-L., Leu F.-Y., You I., Huang Y.-L., Tsai C.-H. AES-128 based secure low power communication for LoRaWAN IoT environment. *IEEE Access*, 2018, vol. 6, pp. 45325–45334. https://doi.org/10.1109/access.2018.2852563
23. Abboud S., Abdoun N. Enhancing LoRaWAN security: an advanced AES-based cryptographic approach. *IEEE Access*, 2024, vol. 12, P. 2589–2606. https://doi.org/10.1109/ACCESS.2023.3348416
24. Nouar A., Abbes M.T., Boumerdassi S., Chaib M. Impact of mobility model on LoRaWAN performance. *Journal of Communications*, 2024, vol. 19, no. 1. pp. 7–18. https://doi.org/10.12720/jcm.19.1.7-18

20. Na S., Hwang D., Shin W., Kim K.-H. Scenario and countermeasure for replay attack using join request messages in LoRaWAN // Proc. of the International Conference on Information Networking (ICOIN). 2017. P. 718–720. https://doi.org/10.1109/ICOIN.2017.7899580
21. Kang J.-M., Lim D.-W. On the quasi-orthogonality of LoRa modulation // IEEE Internet of Things Journal. 2023. V. 10. N 14. P. 12366–12378. https://doi.org/10.1109/jiot.2023.3245885
22. Tsai K.-L., Huang Y.-L., Leu F.-Y., You I., Huang Y.-L., Tsai C.-H. AES-128 based secure low power communication for LoRaWAN IoT environment // IEEE Access. 2018. V. 6. P. 45325–45334. https://doi.org/10.1109/access.2018.2852563
23. Abboud S., Abdoun N. Enhancing LoRaWAN security: an advanced AES-based cryptographic approach // IEEE Access. 2024. V. 12. P. 2589–2606, https://doi.org/10.1109/ACCESS.2023.3348416
24. Nouar A., Abbes M.T., Boumerdassi S., Chaib M. Impact of mobility model on LoRaWAN performance // Journal of Communications. 2024. V. 19. N 1. P. 7–18. https://doi.org/10.12720/jcm.19.1.7-18

Authors

Abdelouahab Nouar — PhD Student, Hassiba Ben Bouali University (UHBC), LMA Laboratory, Chlef, 02010, Algeria, sc 58865584200, https://orcid.org/0009-0001-3355-1912, a.nouar@univ-chlef.dz

Mounir Tahar Abbes — Professor, Hassiba Ben Bouali University (UHBC), Chlef, 02010, Algeria, sc 57212811077, https://orcid.org/0000-0001-5132-2366, m.taharabbes@univ-chlef.d

Selma Boumerdassi — Professor, Conservatoire National des Arts et Metiers (CNAM), Paris, 75141, France, sc 6602291128, https://orcid.org/0000-0003-2603-2433, selma.boumerdassi@inria.fr

Mostefa Chaib — PhD, Researcher, Hassiba Ben Bouali University (UHBC), LMA Laboratory, Chlef, 02010, Algeria, sc 58835296600, https://orcid.org/0000-0001-9137-9527, m.chaib@univ-chlef.dz

Авторы

Нуар Абделуахаб — аспирант, Университет Асиба Бенбуали Лаборатория ЛМА, Шлеф, 02010, Алжир, sc 58865584200, https://orcid.org/0009-0001-3355-1912, a.nouar@univ-chlef.dz

Тахар Аббес Мунир — профессор, Университет Асиба Бенбуали, Шлеф, 02010, Алжир, sc 57212811077, https://orcid.org/0000-0001-5132-2366, m.taharabbes@univ-chlef.dz

Бумердасси Сельма — профессор, Национальная консерватория искусств и ремесел, Париж, 75141, Франция, sc 6602291128, https://orcid.org/0000-0003-2603-2433, selma.boumerdassi@inria.fr

Хаиб Мостефа — PhD, исследователь, Университет Асиба Бенбуали Лаборатория ЛМЕ, Шлеф, 02010, Алжир, sc 58835296600, https://orcid.org/0000-0001-9137-9527, m.chaib@univ-chlef.dz

Received 13.12.2024
Approved after reviewing 02.09.2025
Accepted 25.09.2025

Статья поступила в редакцию 13.12.2024
Одобрена после рецензирования 02.09.2025
Принята к печати 25.09.2025