Privacy-Preserving Pseudonyms for LoRaWAN

Samuel Pélissier, Jan Aalmoes, Abhishek Kumar Mishra, Mathieu Cunche, Vincent Roca, Didier Donsez

Privacy-Preserving Pseudonyms for LoRaWAN

Samuel Pélissier
INSA-Lyon, Inria, CITI Lab.
Lyon, France
samuel.pelissier@insa-lyon.fr

Jan Aalmoes
INSA-Lyon, Inria, CITI Lab.
Lyon, France
jan.aalmoes@insa-lyon.fr

Abhishek Kumar Mishra
INSA-Lyon, Inria, CITI Lab.
Lyon, France
abhishek.mishra@inria.fr

Mathieu Cunche
INSA-Lyon, Inria, CITI Lab.
Lyon, France
mathieu.cunche@insa-lyon.fr

Vincent Roca
University Grenoble Alpes, Inria
Grenoble, France
vincent.roca@inria.fr

Didier Donsez
University Grenoble Alpes, LIG
Grenoble, France
didier.donsez@univ-grenoble-alpes.fr

ABSTRACT

LoRaWAN, a widely deployed LPWAN protocol, raises privacy concerns due to metadata exposure, particularly concerning the exploitation of stable device identifiers. For the first time in literature, we propose two privacy-preserving pseudonym schemes tailored for LoRaWAN: *resolvable* pseudonyms and *sequential* pseudonyms. We extensively evaluate their performance and applicability through theoretical analysis and simulations based on a large-scale real-world dataset of 71 million messages. We conclude that *sequential* pseudonyms are the best solution.

CCS CONCEPTS

• **Security and privacy** → *Mobile and wireless security*; *Security protocols*; **Privacy protections**; • **Networks** → **Link-layer protocols**; • **Computer systems organization** → **Sensor networks**.

KEYWORDS

Privacy, Pseudonyms, LoRaWAN, IoT, Link-layer

ACM Reference Format:
Samuel Pélissier, Jan Aalmoes, Abhishek Kumar Mishra, Mathieu Cunche, Vincent Roca, and Didier Donsez. 2024. Privacy-Preserving Pseudonyms for LoRaWAN. In *Proceedings of the 17th ACM Conference on Security and Privacy in Wireless and Mobile Networks (WiSec '24), May 27–30, 2024, Seoul, Republic of Korea.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3643833.3656120

1 INTRODUCTION

Low-Power Wide-Area Networks (LPWANs) have emerged as a critical communication infrastructure to support the connectivity of countless devices, sensors, and applications across vast areas. However, the proliferation of IoT devices and the ever-expanding network coverage bring significant privacy and security challenges [17]. Among the LPWAN technologies, LoRaWAN (*Lo*ng *Ra*nge Wide Area Network) stands out as a leading solution, offering long-range, low-power, and cost-effective wireless connectivity.

WiSec '24, May 27–30, 2024, Seoul, Republic of Korea

ACM ISBN 979-8-4007-0582-3/24/05…$15.00
https://doi.org/10.1145/3643833.3656120

One of the privacy concerns in LPWANs like LoRaWAN comes from the use of *static* link-layer device identifiers, which are essential for network access and communication, and are exposed in clear-text. These identifiers have been vulnerable to various privacy attacks, such as tracking and profiling [6, 11, 15]. *Identifiers' privacy through pseudonyms has been studied for other protocols such as Bluetooth Low Energy (BLE) and Wi-Fi [4, 8, 13], but to the best of our knowledge has never been explored in LoRaWAN.*

We design privacy-preserving pseudonym schemes for the highly constrained LPWAN environments. We study their performance and limitations through theoretical analysis and simulations based on a large-scale real-world dataset.

2 LORAWAN BACKGROUND

In this section, we present the main relevant aspects of LoRaWAN: its frame format, device identification, and cryptographic materials. We only consider LoRaWAN 1.1, but general concepts apply to both 1.0.X and 1.1 versions.

Frame format: Messages are transmitted in frames (see in Figure 1) containing an encrypted payload surrounded by the Message Integrity Code (MIC), and a header composed of an address field (`DevAddr`), a counter (`FCnt`), and various options.

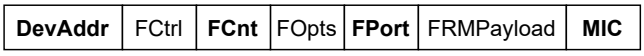

Figure 1: Structure of a LoRaWAN frame.

Messages can be transmitted from the device to the servers (`uplink` messages) or in the opposite direction (`downlink` messages). The frame counter field (`FCnt`) is sent in clear-text and used to order frames and detect losses in both the uplink (`FCntUp`) and downlink (`FCntDown`) directions. Those two counters are initialized at 0 and represented on 32 bits, but only the 16 least-significant bits are included in the frame. Although `FCntUp` is a unique value, `FCntDown` corresponds to two counters increasing asynchronously: `NFCntDown` is incremented for each `downlink` message on port 0 (or when the field is missing), while `AFCntDown` is incremented for each `downlink` message with a port different than 0.

Device identification: `End-Devices` are connected to servers through one or multiple gateways and are addressed by a randomly generated 32-bit identifier named `DevAddr`. Produced during the

join process by the Network Server, it is exposed in clear in all the following frames. It includes two parts: `AddrPrefix` for network identification and `NwkAddr` for `End-Device` identification within the network. The bit allocation for `NwkAddr` varies (from 7 to 25 bits) based on the network type (from 7 to 0).

A single `DevAddr` can sometimes be simultaneously used by multiple `End-Devices`. This can be caused by configuration errors, or optimization of address allocation.[1] In this case, `End-Devices` are distinguished via the MIC: the receiver computes the MIC with potential keys until finding a key that generates a correct MIC, thus identifying the device.[2]

Cryptographic secrets: In LoRaWAN 1.1, two keys are assumed shared between the `End-Device` and the infrastructure: the NwkKey and AppKey. Multiple session keys are derived from them during the join process [5, fig. 49]. For example, the AppKey is used to derive only one key: the AppSKey, responsible of the payload confidentiality via AES-128 CCM encryption [5, sec. 6.2.5]. Network-specific information such as MAC commands are encrypted using the NwkSEncKey. The MIC is computed with AES-128 CMAC [5, sec. 6.2.5] using two keys, guaranteeing integrity and authenticity. Finally, the other header fields (`DevAddr`, `FCnt`...) are not encrypted and are thus available in clear to any eavesdropper.

3 NEEDING LORAWAN-TUNED PSEUDONYMS

LoRaWAN operates under several constraints, which distinguishes it from other protocols such as Wi-Fi or BLE. To quantify the privacy threats level, we investigate the stability of existing identifiers.

Compared to Wi-Fi and BLE, the packet loss rate is high (reported up to 40%) [12], and the communication is mainly based on `uplink` messages, with optional acknowledgment mechanisms. Energy is a limited resource, and everything is made to optimize its usage as devices often operate on battery (a few hundreds mAh) for long periods of time. For instance, the high transmission cost encourages developers to reduce the payload to a minimum [3]. Memory is also sparse: Semtech, the leading manufacturer of LoRaWAN chipsets, states that RAM can be as low as 8kB.[3] While devices have the capability to leave and rejoin the network, it is discouraged due to the additional energy consumption (see Section 5.1).

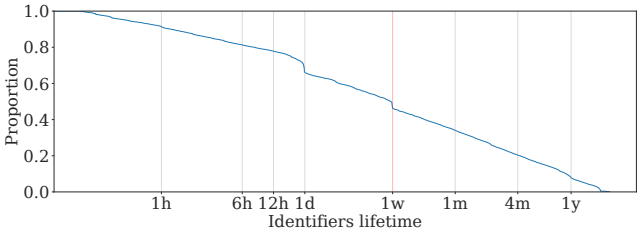

Figure 2: Complementary CDF of the identifiers' lifetime.

Figure 2 shows the complementary cumulative distribution function of the lifetimes of identifiers (`DevAddr`) in a real-world dataset of 71 million LoRaWAN messages, captured from June 2020 to August 2023[4]. We show that 47% of `End-Devices` can be tracked for more than one week via their `DevAddr`, with an average lifespan of approximately 64 days. The ability to consistently monitor `End-Devices` communications over such a substantial duration, thereby collecting increasingly detailed information about their behavior, underlines the need for LoRaWAN-tailored solutions.

4 THREAT MODEL AND PROPERTIES

As seen in Section 3, attackers can leverage the `DevAddr` as a stable identifier to build comprehensive device profiles, effectively tracking users' activities and location [6, 11]. Our threat model includes a passive eavesdropper equipped with sniffers capturing LoRaWAN frames, yet lacking knowledge of the shared keys between devices and servers.

Currently, linking is trivially accomplished due to the `DevAddr`; our goal is to thwart such an approach. A suitable privacy-preserving pseudonym scheme should have the following properties:

$\mathcal{P}_1$**: Unlinkability:** Two messages originating from the same device should not be linked using the link-layer identifier (the `DevAddr`) and other variable frame fields available in clear (e.g. `FCnt`).

$\mathcal{P}_2$**: Minimal communication overhead:** Because of the strict duty cycle and energy constraints of `End-Devices`, the communication overhead has to be kept minimal [3].

$\mathcal{P}_3$**: Legacy support:** To facilitate its co-existence with current specifications and progressive adoption, the scheme should require only limited modifications of the protocol.[5]

$\mathcal{P}_4$**: Low computation/memory overhead:** The computation and memory overhead has to be minimal, in particular for constrained `End-Devices` [3].

$\mathcal{P}_5$**: Reliability:** Receivers should be able to decipher the identity of `End-Devices` without introducing additional message losses.

5 PSEUDONYM SCHEMES FOR LORAWAN

In this section, we consider the design of privacy-preserving link-layer schemes for LoRaWAN. We first delve into existing legacy schemes, before introducing new adapted strategies.

5.1 Limits of legacy schemes

Two schemes can be implemented by leveraging the current LoRaWAN standard: 1) using the existing rejoin process, which allows renewal of the `DevAddr` by the Network Server, or 2) sharing a `DevAddr` with multiple `End-Devices`.

LoRaWAN communication is costly: a complete transmission is around a few milli-joules while encryption only requires pico-joules for the same payload length ($\sim 10^9$ times less) [2, 18]. The first solution implies a communication overhead of at least one ReJoin request and one Join Accept message. Furthermore, a passive observer can easily leverage this rejoin traffic to re-identify devices and thus defeat the purpose of privacy-preserving pseudonyms [14].

[1] https://docs.helium.com/iot/run-an-lns/buy-an-oui/#multiplexing-with-devaddrs.
[2] An widely used open source implementation can be found here: https://github.com/TheThingsNetwork/lorawan-stack/blob/v3.20/pkg/networkserver/grpc_gsns.go#L91
[3] https://lora-developers.semtech.com/documentation/tech-papers-and-guides/mcu-memory-management/
[4] Gateways are deployed around the city of Grenoble, France, and managed by the University of Grenoble Alpes. We exclude experimental networks to only study realistic third-parties. After discussion with our IRB, this dataset cannot be disclosed.
[5] In particular, the general structure of the frame should be preserved. The scheme should only use primitives available in the specifications (e.g. AES).

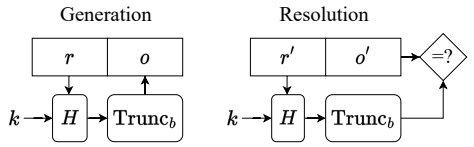

Figure 3: Resolvable Pseudonym scheme in LoRaWAN.

The second scheme does not renew pseudonyms, so the join process has to be used as well. It would only offer a privacy protection of type k-anonymity [16], which is limited because a) creating heterogeneous groups of `End-Devices` allows to easily differentiate devices from each others, and b) homogeneous groups introduce the risk of generating clusters containing only the same class of devices. In addition, the second scheme also induces an overhead on the Network Server by making it compute the MIC to find the correct device. A `DevAddr` shared by D devices requires $1 + \frac{D-1}{2}$ MIC computations on average, instead of one if no `DevAddr` is shared.

We do not further consider the two legacy schemes, as they violate properties $\mathcal{P}_1$ (unlinkability), $\mathcal{P}_2$ (minimal communication overhead), and $\mathcal{P}_4$ (low computation/memory overhead).

5.2 Resolvable pseudonyms

In this section, we first look at utilizing BLE Resolvable random private addresses (RPAs) [9] in LoRaWAN as *resolvable* pseudonyms. We identify drawbacks in directly applying them for LoRaWAN because of the the limited available space in the `NwkAddr` (see Section 2). Using at worst 7 bits for an RPA induces a significant number of identifier collisions. Finally, we propose an adapted scheme that addresses this issue.

Resolvable random private address (RPA): RPAs for BLE are pseudonyms used by peripheral devices to avoid linkability between their subsequent packets. They are built using an Identity Resolution Key (IRK) I_k and a 22-bit random value r by deriving a 24-bit hash h via the hash function H: $h = H(I_k, r)$. The two values are concatenated as $\phi = r||h$ [9, sec. 1.3.2.2]. When observing the RPA, a central device can "resolve" the address utilizing the corresponding IRK to identify the peripheral device, while for eavesdroppers the address appears as random. It can be updated by changing the random value and the corresponding hash, with current implementations typically rotating RPAs every 15 mins [4].

Designing resolvable pseudonyms for LoRaWAN: We illustrate a similar solution fitting LoRaWAN's constraints in Figure 3, both for pseudonym generation and resolution.

For `End-Devices` to generate a resolvable pseudonym ϕ based on RPAs, we need a shared resolution key k, a random value r, and a hash function H. The resolution key can be derived during the join process and is shared between the Network Server and the `End-Device`. While random values can be generated easily, we choose AES-128 CMAC for hashing, as it is already used by each device in the LoRaWAN standard for the MIC computation. The resolvable pseudonym for the `NwkAddr` field consists of r (3 to 12 bits, based on the network type) and a truncated hash value $o = \text{Trunc}_b(\text{H}(r, k))$, where Trunc_b is the function that keeps only the first b bits, and b is between 4 to 13 bits.

Resolving a pseudonym $\phi = (r', o')$ by the Network Server requires the same shared resolution key k (see Figure 3). If both $H(r', k)$ and o' are equal, the identity of the sender is recovered; else, the message may belong to another device and the rest of the shared keys must be tested. This raises two major challenges.

First, 24-bit hashes in RPAs are highly likely to be resolved by a unique key. In contrast, the length of hashes in LoRaWAN can be as low as 4 bits, leading to collisions, i.e. a single pseudonym corresponding to multiple `End-Devices`. To avoid identity conflicts during the resolution, we propose to utilize the MIC (see Section 2). Comparing the received MIC with the locally re-calculated value allows us to reliably identify the actual sender in case of collisions.

Second, a LoRaWAN network can support thousands of active devices, as opposed to BLE. When the Network Server receives an `uplink` message, it has to try *all* the keys of N active devices until it successfully resolves the received pseudonym (testing $1 + \frac{N-1}{2}$ tries on average). Though in `downlink` messages, the `End-Device` just resolves the address with its own key. This open challenge for the `uplink` message resolution alternatively leads us to explore sequential pseudonyms in Section 5.3.

Finally, to protect the `FCnt` from being used for pseudonym linkage purposes [14], we propose to encrypt this field like other network-specific information, using the NwkSEncKey.

5.3 Sequential pseudonyms

In this section, we first investigate an existing solution proposed for Wi-Fi that uses encrypted link-layer addresses [8] as *sequential* pseudonyms. They are subject to similar space constraint issues as with RPAs, leading us to propose an adapted scheme for LoRaWAN.

Encrypted link-layer addresses in Wi-Fi: A variation of the 802.11 link layer protocol, *Shroud*, replaces the unique MAC address by a sequence of pseudonyms generated from a pre-shared key k. The 128-bit pseudonym ϕ_i appearing in the header is derived from the key k and the frame counter i: $\phi_i = \text{AES}_k(i)$. For each paired sender, the receiver maintains a list of pseudonyms for the upcoming counter values. Upon receiving a frame, the receiver performs a lookup to determine the sender's identity.

Designing sequential pseudonyms for LoRaWAN: Figure 4 illustrates sequential pseudonyms tailored for LoRaWAN.

Based on *Shroud*, sequential pseudonyms require a counter i, a shared key k, and an encryption function E. LoRaWAN possesses a counter `FCnt` and a shared network key between the `End-Device` and the Network Server. We utilize AES-CTR for encryption, which already exists in LoRaWAN specification, and truncate the encrypted output to match the size requirements of the `NwkAddr`.

To identify the sender of a pseudonym ϕ', the receiver uses a sequence of m counters and pre-generated pseudonyms. Receiver exploits this information to match ϕ' with the actually utilized counter value and the sender identity. As Network Server can potentially receive messages from N active devices, it needs to maintain N lists of m pseudonyms, while a device only has to maintain one.

Contrary to the *Shroud* implementation using a complete 128-bit AES block as pseudonym, LoRaWAN only has 7 to 25 bits available. Hence, there is a high probability of collisions, i.e. multiple devices generating the same sequential pseudonym. We propose leveraging

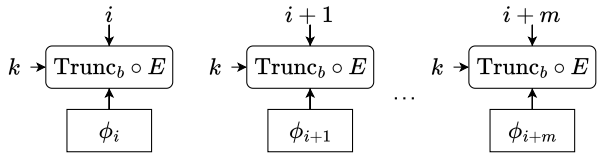

Figure 4: Sequential pseudonyms in LoRaWAN

the MIC to detect the origin when two or more devices use the same sequential pseudonym, as is the case with resolvable pseudonyms.

Similar to the resolvable scheme, the FCnt field cannot be kept in clear in the payload [14]. It is made redundant in sequential pseudonyms as they intrinsically act as counters themselves. We propose to extend the NwkAddr field by the 2 vacated bytes of the FCnt, hence reducing the number of potential collisions. The sequential scheme can thus use pseudonyms of size ranging from 23 (i.e. 16 + 7) to 41 (i.e. 16 + 25) bits depending on the network type.

6 EVALUATION

We evaluate both pseudonym schemes with regards to all necessary guarantees $\mathcal{P}_1$ to $\mathcal{P}_5$ mentioned in Section 4. When feasible, we complement theoretical analysis with simulations based on a real-world dataset of 71 million of uplink messages (cf. Section 3).

6.1 $\mathcal{P}_1$: Unlinkability

Both resolvable and sequential schemes generate their distinct pseudonyms per message using AES as a Cryptographically Secure PRNG (CSPRNG). The input of resolvable pseudonyms is a random value, and the one of sequential is an incremented counter.

Pseudonyms do not exist in isolation: other metadata could be used by an attacker to link messages. The trailing MIC and payload exhibit different values per frame, as they are generated using the *unique* clear-text FCnt value. Even for two equal payload inputs (e.g. temperature sensors reporting the same value), the encrypted output remains different [5, sec. 4.3.3]. The rest of the header (FCtrl and FPort) contains flags and options amounting for a total of 16 bits, but they only exhibit around 4 bits of entropy in our dataset.

Adding the cryptographic guarantees of AES to above-mentioned properties, any two generated pseudonyms are deemed unlinkable.

6.2 $\mathcal{P}_2$: Communication overhead & $\mathcal{P}_3$: Legacy

From a communication perspective, both resolvable and sequential pseudonyms are equivalent. We do not change the message size, and assuming no desynchronization occurs, there is no communication overhead (see Section 6.4.2).

We re-purpose the DevAddr and FCnt fields. Moreover, LoRaWAN supports multiple versions via the the Major Version field of the frame (e.g. v1.0.X and v1.1.Y), allowing legacy support.

6.3 $\mathcal{P}_4$: Computation and memory overhead

We analyze computation and memory overheads, for both End-Devices (ED) and the Network Server (NS) in worst case scenarios. We start with considering N active EDs hosted by a NS. For both schemes, we assume c collisions at the NS for an uplink message.

AES-128 encryption is fundamental to most primitives used in the proposed schemes. For comparison purposes, we approximate the computation overhead based on the required number of blocks of AES-128 encryption. For instance, the FCnt is encrypted in resolvable pseudonyms, requiring one AES block encryption by default for both transmission and reception. The memory overhead is computed in number of bits. We illustrate both overheads in Table 1.

Scheme	Tx (# AES)	Rx (# AES)	Memory (bits)
Resolvable (NS)	1	$4c + 2N + 1$	-
Sequential (NS)	1	$4c + 1$	$N(3mb - \ell)$
Resolvable (ED)	4	3	-
Sequential (ED)	1	1	$3mb - \ell$

Table 1: Computation and memory overhead.

Computation overhead: For transmission (Tx), 1 AES encryption is required for sequential pseudonym by both NS and ED. On the other hand, resolvable pseudonyms require 4 encryption operations for the ED: 1 to generate the random value, 2 to generate the hash (see Figure 3) via CMAC, and 1 to encrypt the FCnt.

For reception (Rx) and for c collisions, $4c$ AES encryptions are required by both the sequential and resolvable schemes: 2 CMAC operations (4 AES encryption) are needed to compute the MIC by the NS [5, sec. 4.4]. In addition, sequential pseudonyms need to generate 1 new pseudonym to maintain the pre-generated list length. Resolvable pseudonyms also require $2N + 1$ additional operations: $2N$ to compute the hash via CMAC based on the shared keys of all N active devices, and 1 to decrypt the FCnt.

One AES encryption is needed by the ED on reception when using sequential pseudonyms to keep the pre-generated list up-to-date. Resolvable pseudonyms require 3 operations: 2 for hash computation via CMAC, and 1 for FCnt decryption.

Memory overhead: We require $3mb - \ell$ bits to store the pseudonyms for the sequential scheme. NS needs to maintain such an information for all N active devices. Each ED uses 3 lists of m pre-generated pseudonyms, one for each FCnt (see Section 2). This introduces a linear memory overhead based on the number of pre-generated pseudonyms m. A pseudonym requires b bits, and we make sure to subtract the existing length (ℓ) of the NwkAddr from the overhead. We confirm in Section 6.4.2 that m remains low (< 30) and identifiers are at most 41 bits. In essence, sequential pseudonyms have significantly lower computation overhead than resolvable ones while having a higher, yet low, memory overhead.

6.4 $\mathcal{P}_5$: Reliability

We study two aspects of pseudonyms' robustness. First, as discussed in Section 5, both resolvable and sequential schemes may lead to pseudonym collisions, inducing additional computations. Second, there might be losses in uplink messages leading to potential End-Device and Network Server desynchronization.

6.4.1 Pseudonym collisions. We study the occurrence of pseudonym collisions through a theoretical analysis and simulations.

Analytical evaluation: We aim to theoretically find the number of pseudonym collisions (Y) for both schemes. We begin with analyzing sequential pseudonyms.

Let ϕ be a device pseudonym and X the number of devices using this pseudonym at a given time. For N devices, sequential pseudonyms require to maintain N lists of m pseudonyms. p is the probability that ϕ belongs to a pseudonym list. Counting the number of devices generating ϕ in their pseudonym list is the same as counting the number of successes of a Bernoulli experiment of probability p. Formally, this counting follows a binomial probability law with the number of lists tested and p as parameters. Since ϕ necessarily belongs to at least one of the lists, we can write $X = 1 + Y$, where Y is the number of pseudonym collisions and X the number of matching lists. Y follows the binomial law of parameter $(N-1, p)$.

Let T be the number of available addresses; for instance, a 32-bit pseudonym has $T = 2^{32}$ available addresses. Then $p = 1-(1-\frac{1}{T})^m$. In conclusion, the probability law of X is the same as $1 + Y$ where Y follows a binomial law of parameter $\left(N-1, 1-(1-\frac{1}{T})^m\right)$.[6]

Considering resolvable pseudonyms, we note that they use half of the available bits to store a random value. Thus, for a 32-bit pseudonym, there are $T = 2^{\frac{32}{2}}$ available addresses. Using the expression deduced above for Y, we notice that decreasing T increases the probability of collisions. In conclusion, resolvable pseudonyms have a higher probability of collisions than sequential pseudonyms.

Simulation: We complete the above analytical evaluation by simulating the proposed schemes on a large-scale real-world dataset presented in Section 3. For the sequential scheme, we generate pseudonyms with and without the additional 16 bits of `FCnt` for comparison purposes.

We consider time windows of one month, during which all observed `DevAddr` are considered as an active `End-Device`. This is because we lack the knowledge of actually active devices in the network. We find the median number of devices is 4789 monthly. We utilize all months observed in the dataset and average the results.

Figure 5 reports the median number of collisions for one `uplink` message in function of the number of bits pseudonyms require. When a pseudonym matches n devices, we count $n-1$ collisions. The sequential scheme outperforms the resolvable one at most points, even without using the `FCnt` (less than 25 bits). We want to stress the numbers presented here correspond to *a single* `uplink` message received by the server. Resolvable pseudonyms generating a median of 1 collision for 25 bits could accumulate rapidly as the network grows to thousands of active devices.

6.4.2 Message losses and desynchronization. The packet loss rate (PLR) of LoRa networks is highly dependent on the deployment and the environment; average values as low as 1% and as high as 40% have been reported [12]. `End-Devices` and Network Server desynchronize when consecutive losses exceed the number of the pre-generated sequential pseudonyms. In this case, any subsequent messages will not be recognized by the receiving end, requiring a costly rejoin process upon detection. We next analyze the probability of desynchronization both analytically and through simulation.

[6]We provide a tool that plots the mass function of the law of X as well as probabilistic indicators in order to choose the appropriate parameters for specific applications. https://gitlab.inria.fr/jaalmoes/spistats

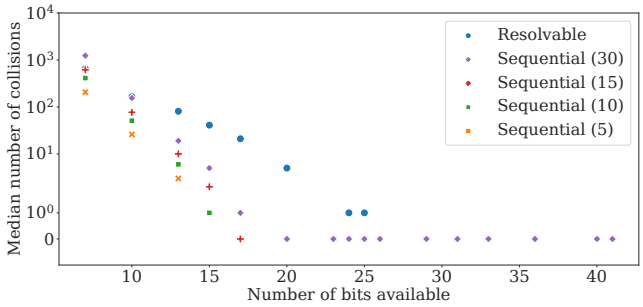

Figure 5: Median number of collisions for an `uplink` message.

Analytical evaluation: The average number of packets sent before m consecutive losses is $\frac{PLR^{-m}-1}{1-PLR}$ [7], where m is the pre-generated list length.

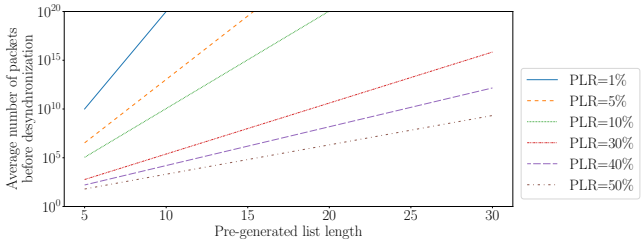

Figure 6: Avg. number of packets before desynchronization.

Figure 6 shows that lists with 15 pseudonyms are enough to ensure that on average it takes $\sim 10^5$ packets for the `End-Device` to desynchronize, even with a 50% loss rate. At a rate of one packet per hour, it would take 11 years on average for an `End-Device` to disconnect, which usually exceeds the battery capacity.

Simulation: We compute the fraction of devices that are desynchronized at least once during the observation period, exploiting our real-world dataset. We only keep `End-Devices` with an estimated loss rate below 50% based on their `FCnt`. This allows us to exclude `End-Devices` with significantly erratic behavior, though a fraction of third-party devices may re-enter the coverage area.

# of pre-generated id. (m)	5	10	15	30
Fraction of devices	30.5%	9.6%	4.8%	0.0%

Table 2: Fraction of devices desynchronizing at least once.

As seen in Table 2, 30.5% of devices experience desynchronization with only 5 pre-generated pseudonyms, but this number falls to 0.0% with a longer list of 30 pseudonyms. Those results show that a sufficiently long list of pre-generated pseudonyms prevents desynchronization for most devices.

In conclusion, $\mathcal{P}_5$ only affects the sequential scheme, and can be rectified with increasing the number of pre-generated pseudonyms.

7 SECURITY CONSIDERATIONS

We investigate both resolvable and sequential pseudonyms from a security perspective, exploring potential side-channel attacks.

First, an attacker could sniff sent messages and replay them at a later time to trigger a specific behavior from the server [10, 21]. For instance, BLE central devices send a specific response if the pseudonym is part of their allow-list, effectively linking randomized MAC addresses to a particular device [21]. This kind of attack is permitted due to the absence of sequence number or time-based randomized address generation. For both resolvable and sequential pseudonyms, the `FCnt` is functionally still available, allowing the Network Server to discard any replayed messages. Timestamps can be included either in pseudonym generation or in the encrypted payload, to check the validity of identifiers (introducing overheads).

Second, various jamming techniques can be used to stop messages from reaching the Network Server [10]. It can also complete replays, where a jammed message is replayed at a later time. Though, such an attack is more advanced and requires to jam *all* the gateways receiving a message. Desynchronizing sequential pseudonyms through jamming is hard as it requires stopping a significant number of consecutive messages (higher than the number of pre-generated pseudonyms m). Inclusion of timestamps alongside of the frame payload is a potential countermeasure.

Third, message length has been used for linking and activity inference [1], and it could also reduce pseudonyms privacy. LoRaWAN messages are generally shorter that their maximum length, limited by the current data rate. Padding remaining bits of the payload mitigates such attacks (at the cost of overheads).

Fourth, an attacker tampering the message integrity can be easily detected by the already-existing MIC [10]. Lastly, networks with a low number of `End-Devices` can be subject to linking and inference attacks via timing of sent messages [11]. To lower the efficiency of such an attack, we opt for changing pseudonyms every message.

8 RELATED WORKS

A key concern of privacy threats in wireless technologies is the presence of a stable, unique link-layer identifier, allowing tracking of device and owner whereabouts by eavesdroppers [19]. Furthermore, thanks to this identifier, the network activity of a device can be monitored and profiled [1].

To tackle above privacy threats, rotating pseudonyms has been introduced for Wi-Fi (802.11) [13] and BLE [4]. However, these solutions generally demand more computation and/or network resources than LoRaWAN has to offer. Asymmetric cryptography-based solutions require comparatively high bandwidth [8]. Other works necessitate acknowledgments for each message [20], which is unrealistic for unreliable LoRaWAN networks [12].

Previously identified LoRaWAN vulnerabilities include jamming, replay attacks, man-in-the-middle, or denial of service [10]. Although privacy is a less researched area, a few previous works explored LoRaWAN specific issues. Like in other wireless protocols, stable identifiers provide an effective way of tracking users activity [11], allowing for device re-identification [14, 15]. To the best of our knowledge, privacy-preserving pseudonyms respecting LoRaWAN constraints have never been proposed.

9 CONCLUSION

This paper presents privacy-preserving pseudonyms tailored to LoRaWAN networks, for the first time in literature. Contrary to existing BLE or Wi-Fi pseudonyms, LoRaWAN demands schemes respecting its resource constraints. We propose two solutions that thwart linking identifiers from the same device: resolvable and sequential pseudonyms. We define necessary properties for LoRaWAN constraints and evaluate their compliance. We study computational and memory overheads, as well as adverse events such as identifier collisions and desynchronization. We utilize both analytical evaluation and simulations considering a real-world dataset of 71 million messages. We demonstrate that sequential pseudonyms outperform resolvable pseudonyms, offering minimal overhead and a reduced risk of collisions and desynchronization. In the future, we plan to explore pseudonyms for other LPWAN protocols.

ACKNOWLEDGMENTS

This work has been supported by the ANR-BMBF PIVOT project (ANR-20-CYAL-0002), H2020 SPARTA project and the INSA-Lyon SPIE ICS IoT Chair.

REFERENCES

[1] A Acar, H Fereidooni, T Abera, AK Sikder, M Miettinen, H Aksu, M Conti, AR Sadeghi, and S Uluagac. 2020. Peek-a-Boo: i See Your Smart Home Activities, Even Encrypted!. In *WiSec*.
[2] E Bäumker, A Miguel Garcia, and P Woias. 2019. Minimizing Power Consumption of LoRa ® and LoRaWAN for Low-Power Wireless Sensor Nodes. *Journal of Physics: Conference Series* 1407, 1 (Nov. 2019).
[3] L Casals, B Mir, R Vidal, and C Gomez. 2017. Modeling the Energy Performance of LoRaWAN. *Sensors* 17, 10 (2017), 2364.
[4] G Celosia and M Cunche. 2019. Saving Private Addresses: An Analysis of Privacy Issues in the Bluetooth-Low-Energy Advertising Mechanism. In *MobiQuitous'19*.
[5] LoRa Alliance Technical Committee. 2017. LoRaWAN® Specification v1.1. https://lora-alliance.org/resource_hub/lorawan-specification-v1-1/.
[6] L Dujić Rodić, T Perkovic, M Skiljo, and P Solic. 2022. Privacy Leakage of Lorawan Smart Parking Occupancy Sensors.
[7] P Ginsparg. 2005. How many coin flips on average does it take to get n consecutive heads? https://www.cs.cornell.edu/ ginsparg/physics/INFO295/mh.pdf.
[8] B Greenstein, D McCoy, J Pang, T Kohno, S Seshan, and D Wetherall. 2008. Improving Wireless Privacy with an Identifier-Free Link Layer Protocol. In *MobiSys*.
[9] Core Specification Working Group. 2023. Bluetooth® Core Specification 5.4.
[10] F Hessel, L Almon, and M Hollick. 2022. LoRaWAN Security: An Evolvable Survey on Vulnerabilities, Attacks and Their Systematic Mitigation. *ACM Transactions on Sensor Networks* (Nov. 2022).
[11] P Leu, I Puddu, A Ranganathan, and S Čapkun. 2018. I Send, Therefore I Leak: Information Leakage in Low-Power Wide Area Networks. In *WiSec'11*.
[12] Q Liu, Y Mu, J Zhao, J Feng, and B Wang. 2020. Characterizing Packet Loss in City-Scale LoRaWAN Deployment: Analysis and Implications. In *IFIP*. IEEE.
[13] J Martin, T Mayberry, C Donahue, L Foppe, L Brown, C Riggins, E C. Rye, and D Brown. 2017. A Study of MAC Address Randomization in Mobile Devices and When It Fails. *arXiv:1703.02874 [cs]* (March 2017). arXiv:1703.02874 [cs]
[14] S Pélissier, M Cunche, V Roca, and D Donsez. 2022. Device Re-Identification in LoRaWAN through Messages Linkage. In *WiSec*.
[15] P Spadaccino, D Garlisi, F Cuomo, G Pillon, and P Pisani. 2021. Discovery Privacy Threats via Device De-Anonymization in LoRaWAN. In *MedComNet'21*.
[16] L Sweeney. 2002. K-Anonymity: A Model for Protecting Privacy. *International journal of uncertainty, fuzziness and knowledge-based systems* (2002).
[17] N Torres, P Pinto, and SI Lopes. 2021. Security Vulnerabilities in LPWANs—An Attack Vector Analysis for the IoT Ecosystem. *Applied Sciences* 11, 7 (Jan. 2021).
[18] KL Tsai, FY Leu, I You, SW Chang, SJ Hu, and H Park. 2019. Low-Power AES Data Encryption Architecture for a LoRaWAN. *IEEE Access* 7 (2019).
[19] M Vanhoef, C Matte, M Cunche, L S. Cardoso, and F Piessens. 2016. Why MAC Address Randomization Is Not Enough: An Analysis of Wi-Fi Network Discovery Mechanisms *(ASIA CCS '16)*. Association for Computing Machinery.
[20] Q Zhang and K Zhang. 2022. Protecting Location Privacy in IoT Wireless Sensor Networks through Addresses Anonymity. *Security and Comm. Networks* (2022).
[21] Y Zhang and Z Lin. 2022. When Good Becomes Evil: Tracking Bluetooth Low Energy Devices via Allowlist-based Side Channel and Its Countermeasure. In *ACM CCS*.