IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received January 3, 2021, accepted January 13, 2021, date of publication February 1, 2021, date of current version February 9, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3055771

Data Authentication and Anonymization in IoT Scenarios and Future 5G Networks Using Chaotic Digital Watermarking

BORJA BORDEL[1], RAMÓN ALCARRIA[2], TOMAS ROBLES[1], AND MARCOS SÁNCHEZ IGLESIAS[3]

[1]Departamento de Sistemas Informáticos, Universidad Politécnica de Madrid, 28031 Madrid, Spain
[2]Departamento de Ingeniería Topográfica y Cartografía, Universidad Politécnica de Madrid, 28031 Madrid, Spain
[3]Escuela Superior de Informática, 13071 Ciudad Real, Spain

Corresponding author: Ramón Alcarria (ramon.alcarria@upm.es)

This work was supported by the Ministry of Science, Innovation and Universities through the COGNOS Project under Grant PID2019-105484RB-I00.

ABSTRACT Future Internet-of-Things (IoT) scenarios and applications are envisioned to be supported by emerging 5G networks. In this context, complex routing schemes for pervasive infrastructures are highly simplified, as every hardware element may stablish its own communication link with a 5G base station. However, this situation also introduces new risks, especially in the security field where innovative cyber-physical attacks and distributed denial of service attacks are becoming more popular and dangerous each day. Thus, data authentication, protection and anonymization in those new applications and schemes is a key challenge to be addressed. Besides, most devices in future IoT systems will be resource constrained, so traditional solutions based on private keys stored in devices' memory and computationally heavy cryptographic algorithms will turn unsecure, inefficient or, directly, impossible to run. Therefore, in this paper we propose a new mechanism to protect, authenticate and anonymize data in IoT systems supported by future 5G networks. The proposed solution employs both digital watermarking techniques and lightweight cryptographic technologies. To generate keys in a secure and simple manner, physical unclonable functions are employed. Besides, to reduce as much as possible the computational cost of algorithms, chaotic dynamics will be considered. In order to evaluate the performance of the proposed solution an experimental validation based on simulation techniques is also carried out.

INDEX TERMS 5G networks, Internet-of-Things, digital watermarking, physical unclonable functions, data authentication, chaotic encryption.

I. INTRODUCTION

Future Internet of Things (IoT) systems [1], as well as other future engineered solutions as Cyber-Physical Systems [2], are envisioned to be communicated through 5G networks [3] in a short time. This new union is creating a large catalogue of synergies and advantages. From the possibility of communicating IoT devices through the enhanced mobile broadband links provided by 5G networks; to the simplification of complex routing mechanisms in IoT deployments, as 5G networks are supporting massive machine type communications (so every IoT device could stablish its own communication link with the base station) [4].

However, this new approach also introduces new and dangerous risks [1]: the dependency on proprietary networks (problematic for critical or governmental applications), the creation of bottlenecks in the 5G base stations, and, of course, vulnerabilities related to data usurpation and espionage, cyber-physical attacks, distributed denial of service and reverse engineering.

Basically, with these new 5G communication architectures it is simpler to identify what components, sensors and/or devices are deployed in any IoT system. Protocol headers and metadata (among other information) transmitted through wireless media are easy to intercept. With this information, knowledge about the system configuration and its users may be extracted. And this knowledge may be employed to create powerful cyber-physical attacks to critical infrastructures [5] or distributed denial of service (DDoS) attacks [6].

The associate editor coordinating the review of this manuscript and approving it for publication was Ilsun You.

Or, if desired, to be sold as valuable personal information (biological signals, video streams, etc.).

Actually, if most elements in a IoT hardware deployment are identified (a simple task because all devices expose their communication interfaces to a well-known element, the base station), all this pervasive infrastructure may be used as vector to create a harmful effect in a remote device or area (even in the physical world, not only in the cyberspace) [7]. This approach, known as cyber-physical attack, may be based on the injection of false information (to simulate, for example, a person is suffering a cardiac problem in a eHealth solution); or on the massive infection of devices to employ their resources in a malicious manner. A particular case of this second methodology is the distributed denial of service attack, where devices are employed to block a legitimate service.

On the other hand, sometimes, identifying which kind of devices are deployed in an IoT system is interesting for criminals, so they can capture valuable personal information (for example, video streams in surveillance applications, biological signals in eHealth solutions, etc.), by distinguishing important data signals from irrelevant context sensor information.

As a solution, IoT devices in future 5G scenarios should include strong privacy protection, data anonymization and data authentication mechanisms to ensure the information is protected from illegal accesses, the origin of every data package is identified and legitime and, at the same time, no detail about the system configuration or composition is provided. Cryptographic technologies may address this problem, but IoT devices face important problems in that field. First, most IoT nodes are resource constrained, so they cannot support complex and computationally heavy algorithms, such as modern symmetric key encryption schemes [8]. Besides, IoT systems are many times deployed in geographically remote areas with no physical protection, so solutions based on private keys stored in devices' memory turn unsecure (as an intruder could easily extract that memory from the node and employ fraudulently the private key) [8]. Other innovative mechanisms such as Physical Unclonable Functions (PUF) have been successfully employed in these scenarios [9], but they cannot solve all the previously described challenges. In conclusion, new data authentication and anonymization mechanisms for IoT systems and future 5G networks are needed.

Therefore, in this paper, it is described an innovative new data authentication and anonymization solution, protecting both: the personal information collected and sent by IoT systems, and the configuration information about the IoT deployment itself. To remove unsecure private keys from devices' memory, PUFs are considered together with random numbers generators in order to generate dynamic keys. These keys are introduced into chaotic dynamics (which are numerically evaluated) to create lightweight encryption schemes and digital watermarking algorithms, providing privacy and data authentication. Finally, to anonymize data and prevent any attacker to learn about the system configuration, IoT nodes implement a pseudo-random routing protocol which applies these chaotic techniques in nested manner.

The rest of the paper is organized as follows. Section II describes the state of the art on data protection, authentication and anonymization techniques in IoT systems and future 5G networks. Section III presents the proposed new mechanism, including the key generation, the chaotic encryption and watermarking algorithms, and the pseudorandom routing protocol. Section IV describes an experimental validation, based on simulation techniques, carried out to evaluate the performance of the proposed solution. Section V presents the obtained experimental results and Section VI concludes the paper.

II. STATE OF THE ART

Data authentication and anonymization has been identified as a pending challenge in the area for many years [10]. However, there is an important decompensation, and a much higher number of works about data anonymization may be found.

Actually, proposals for data authentication in the Internet of Things are sparse, and commonly developed for specific applications. Simplest solutions, typically for non-networked industrial deployments, are supported by unique identifiers (such as MAC addresses) which are added to data to indicate the origin of the information [11]. However, these identifiers may still be cloned or spoofed, so these solutions do not reach a good performance in terms of security. Other alternatives provide deeper analysis of data signals using signal processing techniques to deduct whether they have been manipulated. Many eHealth applications follow this approach [12], [5], but even these algorithms can be easily hacked, for example, by injecting real but fraudulent signals. Most modern proposals employ, as almost every recent cryptographic technique, asymmetric encryption, and signature [13], [42]. Nevertheless, no information is provided about how these complex and precise mathematical algorithms may be implemented using low cost and resource constrained computing nodes. Finally, other works describe techniques based on digital watermarking [14], as in the present paper. However, these schemes associate a unique watermark to every single node and/or cluster, so information about the IoT system may be still extracted, although data are successfully authenticated.

Contrary to these works, in the proposed solution, watermark change dynamically, and a pseudorandom routing protocol is employed to hide information about the system configuration.

On the other hand, many different schemes for data anonymization in IoT systems have been reported. In fact, privacy, in its multiple forms, is probable one of the most studied research lines nowadays in the IoT field [15]. One of the most common solutions to anonymize data is to aggregate them into some cluster signals, so no information about the hardware nodes is revealed [16]. However, with this approach, the system also loses capillarity. Some proposals even go further and propose to store data in a Hadoop (or similar) database to extract statistical indicators which are later sent to servers [17]. The problem of this approach, any case, is the same. Another popular and powerful technique for data anonymization is differential privacy [18]. In this mechanism, data suffer an artificial distortion which hides all valuable information and only a legitime final user can remove the noise and recover the data. This approach has been reported integrated into industrial systems (where the devices location must be hidden) [19], and into artificial intelligence schemes [20]. However, these techniques are typically computationally costly, and many IoT devices cannot support learning algorithms and signal processing techniques. In this area, a very recent approach is privacy-by-design [21], so privacy is not supported by new and additional modules, but by the system configuration itself. Different architectures based on the minimization of data acquisition [22], number of data sources [23], data storage [24], knowledge discovery [25], data granularity [26]; or on the creation of distributed signal processing [27] and distributed data storage [28] algorithms have been reported. Although all these solutions have a good performance, at the end, they cause a reduction in the system capabilities, which may reduce the applicability and utility of IoT technologies in some scenarios.

Finally, some hidden data routing techniques may be found [29]. In this approach, routing protocols are designed in such a manner that no information about the system configuration is provided or transferred. Most common approach employs TOR (The Onion Router) network protocols in closed IoT deployments [30]. Nevertheless, this technique requires both, complex asymmetric encryption algorithms, and computationally heavy protocols (such as TCP -Transmission Control Protocol-). Besides, no solution about how data managed in that way can be sent outside the IoT deployment (to a cloud server, for example).

On the other hand, as a main problem, all these techniques are typically focused only in one information type: user information or system information. But the proposed technology addresses both problems. Besides, contrary to reported works, in this paper the proposed mechanism provides privacy and authentication at the same time.

Finally, some works on data anonymization and authentication in future 5G networks may be found. The situation is the opposite to the previously described for IoT solutions. Works on data anonymization are very sparse. Only some initial works on how to provide privacy in massive machine type communications in 5G networks [34] have been reported. On the other hand, although they are not common, works on data authentication in 5G networks are more numerous. Some techniques to authenticate messages in future 5G handovers, based of digital signatures, have been reported [31]. Other mechanisms supported by cyclic redundant codes have been also described [32]; and, even, digital watermarking technologies to support copyright to content have been designed [33]. Nevertheless, in all these technologies the mobile network must participate actively, and that does not match well with end-to-end policies in IoT deployments. Other techniques where the network provides technological support but does not participate in the security technology are, then, needed. The proposed solution in this paper aims to fill this gap.

III. A NEW AUTHENTICATION AND ANONYMIZATION SOLUTION

In this section, a new authentication and anonymization technology is described, including all phases: from the pseudorandom routing protocol (Section III.B), to the proposed encryption and digital watermarking mechanism (Section III.C) and the key generation process (Section III.D).

A. APPLICATION SCENARIO AND OVERVIEW

Future 5G networks are envisioned to follow a two-level architecture, where microcells and macrocells get combined to provide IoT devices of enhanced mobile broadband communications, ultra-low-latency communications and massive machine-type communications. Up to M different microcells μ_i (1) support the 5G physical links with the devices within an IoT subsystem Φ, where up to N different devices communicate with the micro base stations (2). The subsystem, then, works in a tree-like topology (see Figure 1).

$$\mathcal{M} = \{\mu_i \quad i = 1, \ldots, M\} \tag{1}$$

$$\Phi = \{\phi_i \quad i = 1, \ldots, N\} \tag{2}$$

Although devices could move between different subsystems or macrocells (for example if devices are transferred by administrators between different applications of geographical scenarios), in this work we are assuming devices are fixed for a time long enough to enable the convergence and successful operation of the proposed protocols (see Section III.B). The scenario also considers a remote server (in the cloud) managing and receiving all data from the IoT subsystem (see Figure 1).

The proposed new technology must, then, hide all personal or configuration information about the IoT subsystem and its users to any component outside the subsystem and different from the remote cloud server. Besides, the technique must enable the remote server to identify and guarantee the legitimate origin of information.

Figure 2 presents the block-diagram for the proposed novel authentication and anonymization technology. Each IoT node presents four different interfaces: one (named as I_{base}) connects the node with a micro base station, the second one (named as I_{input}) connects the node with other nodes from which it receives messages to be routed, the third one (named as I_{data}) connects the authentication and anonymization module with sensors and other devices generating data signals in the IoT node, and, finally, the fourth one (named as I_{output}) is employed by the node to send messages to other physically-connected nodes.

Every message or data received by interfaces I_{input} or I_{data} follows, initially, the same process. The input information is encrypted using a chaotic cryptographic mechanism.

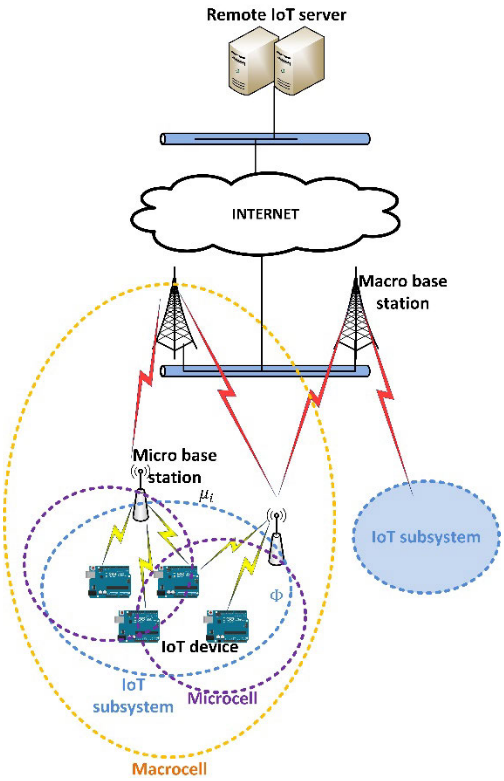


FIGURE 1. Application scenario of IoT and future 5G networks.

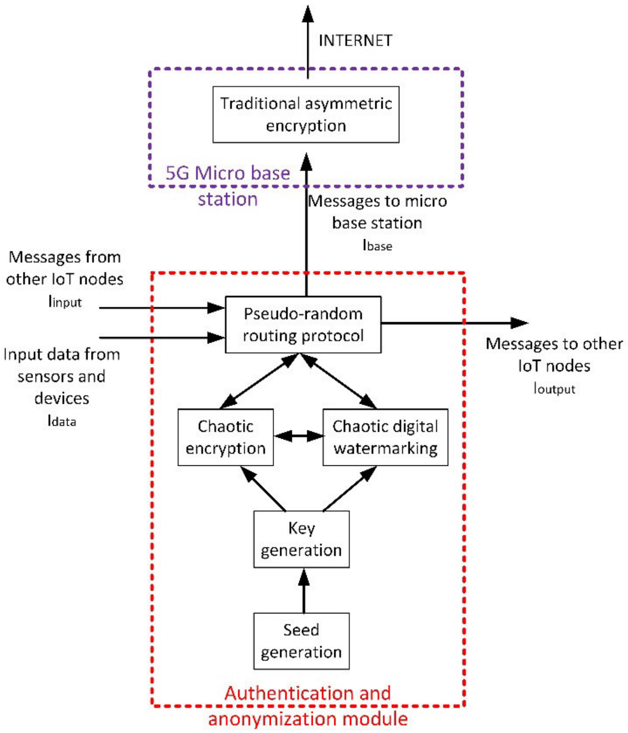

INTERNET
Traditional asymmetric encryption
5G Micro base station
Messages to micro base station
l_{base}
Messages from other IoT nodes
l_{input}
Input data from sensors and devices
l_{data}
Pseudo-random routing protocol
Messages to other IoT nodes
l_{output}
Chaotic encryption
Chaotic digital watermarking
Key generation
Seed generation
Authentication and anonymization module

FIGURE 2. Diagram for the 'proposed authentication and anonymization mechanism.

This mechanism employs a non-linear dynamic to generate in a lightweight manner a pseudo-random signal which is employed to modulate and protect the original raw data.

After this encryption phase, the encrypted message is processed by a pseudorandom routing module. In this module, a probabilistic classifier decides if the encrypted message must be sent to micro base station to be transmitted to the remote server, or it must be transmitted to a secondary IoT node (randomly selected among the nodes physically reachable). Data generated and encrypted by the IoT node itself are always sent to a secondary IoT node. If the encrypted message must be transmitted to a secondary IoT node, the packet is encapsulated with a header indicating the IoT node identifier and the number of hops that the message has already performed inside the IoT subsystem. As a result, the original message will circulate in a random manner within the IoT subsystem, nesting different headers and encryption schemes which hide both the origin or the information, the private data and the system configuration.

Before being transmitted (to the micro base station or to a secondary node), the message (encapsulated or not) is marked using a chaotic digital watermarking technology, based in a similar philosophy to the encryption scheme. In that way, it is possible to detect information modifications, malicious information injections and other similar frauds and cyber-physical attacks.

In the last hop, when the message is sent outside the IoT subsystem, the micro base station learns about the device that sends the message to return the response using the same path. Nested headers added in the pseudorandom routing protocol enable this return path. Besides, from this point, traditional computationally heavy mechanisms can be freely employed (base station does not have constrained resources), so typically asymmetric encryption will be employed to protect messages during the traveling through the Internet. Any case, with this new technology, no component in the 5G network or intruder in the wireless media can now learn about the structure or users in the IoT subsystem.

Finally, chaotic encryption and digital watermarking mechanisms, to be secure, must be initialized with a secret key with validated cryptographic properties. This key cannot be generated using pre-fixed stored data in the device's memory (as this scheme is totally unsecure against cyber-physical attacks). Then, to generate that key we are using modulation functions and lightweight pseudo-random number generators (PRNG), which are fed with a seed obtained through Physical Unclonable Functions (PUF). As a result, a good cryptographic key is obtained, but this key is dynamic and cannot be cloned by any attacker as it is supported by PUF.

In the next subsection we are analyzing each module with details.

B. ROUTING PROTOCOL FOR DATA ANONYMIZATION

In typical IoT subsystems, routing mechanisms reveal a lot information about the system configuration. Public addresses, metadata in protocol headers and, even, the data format in the payload may provide valuable information to attackers about the subsystem configuration. Besides, typical wireless interfaces among nodes in an IoT subsystem, and between nodes and the micro base station, must be protected against intruders in the physical medium (performing, basically two different attacks: data injection and data sniffing).

To hide this system information in the routing protocol and protect and authenticate the user information, a new routing technique is needed, which (at least) must fulfill the following requirements:

- REQ#1: The protocol must be anonymous. No node in the subsystem or in the 5G network must know the identity of the other nodes in the IoT deployment. Except from those to which the node is physically connected.
- REQ#2: The routing protocol must be lightweight to meet IoT devices characteristics
- REQ#3: The routing protocol must be datagram oriented. No session should be stablished.
- REQ#4: The routing must be unpredictable, dynamic and prevent an intruder to learn about the entire system, even if it keeps sniffing for an indefinite time.

To solve this problem, we propose a protocol based on nested, encrypted and watermarked packets, following a random and totally unpredictable path (hereinafter, pseudorandom routing protocol). We define a partition of the set of IoT devices Φ (3), see figure 3.

$$\Phi = \Phi_{\text{tier1}} \cup \Phi_{\text{tier2}} \cup \Phi_{\text{tier3}} \tag{3}$$

In this paper, we are considering three different sets and tiers, but any arbitrary number of layers could be created. However, all possible structures and subdivisions may be reduced to the elemental scenario where only three layers are defined: (i) devices acting as primary data sources, (ii) devices acting as datagram relays, and (iii) devices acting as gateway with the mobile network. A more detailed description for each layer, including all their properties and restrictions is provided below.

The first subset Φ_{tier1} includes all critical or tier#1 devices. These devices never reveal their identity to the micro base station and must be low congested as their resources are sparse. In general, a maximum congestion value ρ_1 must be tolerated. This congestion value is calculated as in the standard traffic theory (4), from the number of devices in the tier λ_1 and the medium time required for a packet to be transmitted outside the tier η_1. Besides, if we consider the nominal (or effective, if desired) bitrate of the wireless technology supporting the communication among IoT nodes, r bit/s, and the average length of data packets L, it is simple to estimate the maximum number of hops Z_1 a packet may do inside the tier, before being transmitted to the next tier (5).

$$\rho_1 = \frac{\eta_1}{\lambda_1} \tag{4}$$

$$\begin{aligned} \lambda_1 &= card\,\{\Phi_{\text{tier1}}\} \\ Z_1 &= \lambda_1 \cdot \rho_1 \cdot \frac{r}{L} \end{aligned} \tag{5}$$

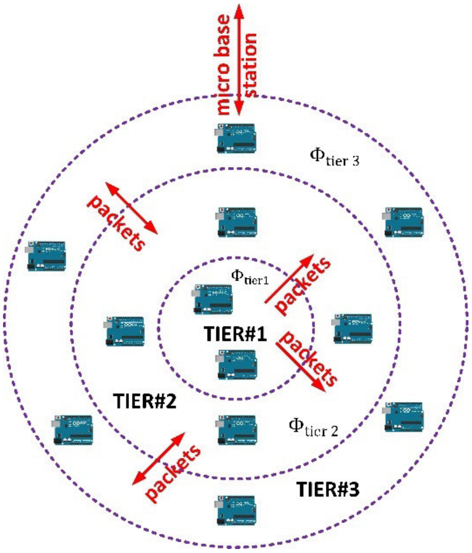


FIGURE 3. Internal structure in an IoT subsystem: tiers.

The second subset Φ_{tier2} includes all sub-critical or tier#2 devices. These devices are adjacent to tier#1 devices, they receive their packets, but they can only inject packets inside the tier#1 with destination within that tier. They have relatively abundant resources, with several redundancies, and can support a higher maximum congestion value ρ_2, which may be employed to obtain the maximum number of hops within the tier Z_2 as previously done (6). However, they also protect their identity from micro base stations.

$$Z_2 = \lambda_2 \cdot \rho_2 \cdot \frac{r}{L} \tag{6}$$

Only adjacent tiers may communicate, so devices within tier#1 and tier#3 cannot exchange data packets directly.

Finally, the third subset Φ_{tier3} includes all expendable devices. These devices are adjacent to tier#2 devices with which they can freely communicate (bidirectional). These devices, besides, perform actions that are not essential for the IoT subsystem, so if they get blocked or infected, the main system's functionality is not affected. Thus, devices in tier#3 can expose their identity and communicate to micro base stations. However, in order to keep running the connection of the IoT subsystem with the remote server, these devices tolerate a level of congestion ρ_3 lower than supported by devices in tier#2. Then, the maximum number of hops Z_3 is also expected to be lower (7).

$$Z_3 = \lambda_3 \cdot \rho_3 \cdot \frac{r}{L} \tag{7}$$

Additionally, the i-th node in the IoT subsystem Φ also defines a new set Φ^i_{local}, containing all the IoT nodes with which it has physical connection.

Considering this scenario, we assume an IoT node receives a new private information $\Im$ directed to the remote server. This information may be a packet from another device, or new data generated by sensors or computing elements making up the node. No data stream or flow is considered, as the protocol must be datagram oriented (REQ#3). The proposed pseudo-random routing protocol, then, employs a chaotic encryption function $\mathbb{E}(\cdot)$, see Section III.C, to protect the received information. An encrypted information $\Im_{en}$ is then obtained (8).

$$\Im_{en} = \mathbb{E}(\Im) \tag{8}$$

This encrypted information is, then, classified using a probabilistic module. This module determines if the information must be transmitted towards the following tier (if the IoT node belongs to Φ_{tier3} the following tier is the micro base station), or if the packet must perform another hop inside the same tier. In order to guarantee an intruder cannot use reverse engineering techniques to learn about the system through the routing paths, this decision is random (REQ#4). The pseudo-random routing module includes a random function, which is evaluated to make the decision. This function is discrete for all devices (only two or three values are possible). This function, for devices in tier#1 and tier#2 is a Bernoulli variable, as only two possible values can be taken. On the one hand, devices in tier#1 can only communicate with devices in the same tier and devices in tier#2. On the other hand, although devices in tier#2 may communicate with devices in all tiers, they cannot inject packets direct to the remote server in tier#1, so finally only tier#2 and tier#3 are reachable. Finally, for devices in tier#3, this function may take three different values, as they can communicate to micro base stations, devices in tier#3 and devices in tier#2.

Moreover, the proposed model must guarantee two important factors: (i) every data packet must be finally transmitted to the remote server and (ii) packets circulating through the different tiers do not cause congestion. In order to guarantee these requirements, proposed random functions are not fixed, but depend on the maximum number of hops for each tier, and the number of hops already performed by the packet to be forwarded.

Thus, random function for devices in tier#1, X_1, evolves according to an exponential law (9), so only a small number of hops are performed in this tier (reducing as much as possible the resource consumption). On the other hand, random function for devices in tier#2, X_2, follows a rational function (10), which guarantees packets circulate randomly around the network enough to hide all valuable information. Finally, random function for devices in tier#2, X_3, follows a power law (11), whose growth is between a rational and an exponential function.

$$\begin{aligned}
&X_1 \sim Be\left(\Phi_{next}; n|n \leq Z_1\right)\\
&= \begin{cases} exp\left(\dfrac{Z_1 - n}{n}\right) & if\ \Phi_{next} = \Phi_{tier1}\\ 1 - exp\left(\dfrac{Z_1 - n}{n}\right) & if\ \Phi_{next} = \Phi_{tier2}\end{cases}\\
&X_1 \sim Be\left(\Phi_{next}; n|n > Z_1\right)\\
&= \begin{cases} 0 & if\ \Phi_{next} = \Phi_{tier1}\\ 1 & if\ \Phi_{next} = \Phi_{tier2}\end{cases}
\end{aligned} \tag{9}$$

$$\begin{aligned}
&X_2 \sim Be\left(\Phi_{next}; n|n \leq Z_2\right)\\
&= \begin{cases} \dfrac{Z_2 - n}{Z_2} & if\ \Phi_{next} = \Phi_{tier2}\\ \dfrac{n}{Z_2} & if\ \Phi_{next} = \Phi_{tier3}\end{cases}\\
&X_2 \sim Be\left(\Phi_{next}; n|n > Z_2\right)\\
&= \begin{cases} 0 & if\ \Phi_{next} = \Phi_{tier2}\\ 1 & if\ \Phi_{next} = \Phi_{tier3}\end{cases}
\end{aligned} \tag{10}$$

$$\begin{aligned}
&X_3 \sim X\left(\Phi_{next}; n|n \leq Z_3\right)\\
&= \begin{cases} n \cdot \displaystyle\prod_{j=0}^{K} \left(\dfrac{Z_3 - n}{Z_3}\right)^{2j} & if\ \Phi_{next} = \Phi_{tier3}\\ 1 - n \cdot \displaystyle\prod_{j=0}^{K} \left(\dfrac{Z_3 - n}{Z_3}\right)^{2j} - \dfrac{1}{K+1}\displaystyle\sum_{j=0}^{K}\left(\dfrac{n}{Z_3}\right)^{2j+1} & \\ & if\ \Phi_{next} = \Phi_{tier2}\\ \dfrac{1}{K+1}\displaystyle\sum_{j=0}^{K}\left(\dfrac{n}{Z_3}\right)^{2j+1} & if\ \Phi_{next} = \mathcal{M}\end{cases}\\
&X_3 \sim X\left(\Phi_{next}; n|n > Z_3\right)\\
&= \begin{cases} 0 & if\ \Phi_{next} = \Phi_{tier3}\\ 0 & if\ \Phi_{next} = \Phi_{tier2}\\ 1 & if\ \Phi_{next} = \mathcal{M}\end{cases}
\end{aligned} \tag{11}$$

Being n the number of hops already performed by the packet ($n = 0$ if the information is generated by the IoT node), and Φ_{next} is the subset from which the following node in the routing path is selected.

It is important to note that all random functions guarantee, at least, one random hop, as all random function for $n = 0$ forces (probability equal to the unit) to transmit packets within the tier or to lower tiers (12). Besides, all random functions guarantee that no congestion is never caused in the different tiers, as the maximum number of hops is always preserved (13).

$$\begin{aligned}
X_1 \sim Be\left(\Phi_{next}; 0\right) &= \begin{cases} 1 & if\ \Phi_{next} = \Phi_{tier1}\\ 0 & if\ \Phi_{next} = \Phi_{tier2}\end{cases}\\
X_2 \sim Be\left(\Phi_{next}; 0\right) &= \begin{cases} 1 & if\ \Phi_{next} = \Phi_{tier2}\\ 0 & if\ \Phi_{next} = \Phi_{tier3}\end{cases}\\
X_3 \sim X\left(\Phi_{next}; 0\right) &= \begin{cases} 0 & if\ \Phi_{next} = \Phi_{tier3}\\ 1 & if\ \Phi_{next} = \Phi_{tier2}\\ 0 & if\ \Phi_{next} = \mathcal{M}\end{cases}
\end{aligned} \tag{12}$$

$$\begin{aligned}
X_1 \sim Be\left(\Phi_{next}; n \to Z_1^{-}\right) &= Be\left(\Phi_{next}; n|n > Z_1\right)\\
X_2 \sim Be\left(\Phi_{next}; n \to Z_2^{-}\right) &= Be\left(\Phi_{next}; n|n > Z_2\right)\\
X_3 \sim Be\left(\Phi_{next}; n \to Z_3^{-}\right) &= Be\left(\Phi_{next}; n|n > Z_3\right)
\end{aligned} \tag{13}$$

Then, the encrypted message is encapsulated, resulting in a new encapsulated packet $\Im_{cap}$. The included header depends on the device toward the packet is going to be forwarded (see Figure 4). If the next device belongs to the IoT subsystem, the header includes the following three fields:

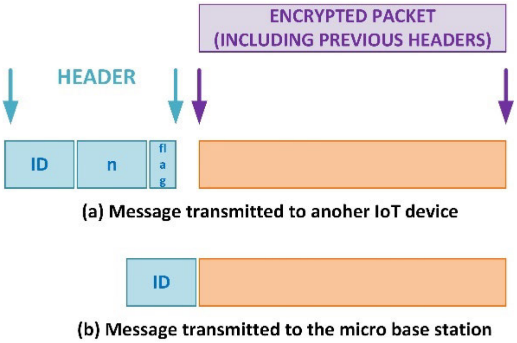

ENCRYPTED PACKET
(INCLUDING PREVIOUS HEADERS)
HEADER
ID
n
fl
a
g
(a) Message transmitted to anoher IoT device
ID
(b) Message transmitted to the micro base station

FIGURE 4. Pseudorandom routing protocol: message format.

- Device identity or identifier (ID): The unique identifier of the node within the IoT subsystem must be included.
- Number of hops (n): It indicated the number of hops already performed by the packet. It is set to the unit if the information is genuinely generated by the IoT device. If the packet was received from another IoT device, this value is obtained increasing in one unit the previous value (which may be read from the previous header, encrypted in the first step).
- Control flag: This flag is set to the unit if data come from the remote server and are directed to an IoT device in the subsystem. On the contrary, the flag is set to zero. In forwarded packets, the flag is directly taken from the previous header.

If the following device is the micro base station, the header will only contain the identification of the encrypting node.

It is important to note that only physically connected devices are communicating in our solution, so forwarding takes place at link level, where addresses, protocols, and header according to the wireless technology being employed will be used.

Now, before sending this encapsulated packet $\Im_{cap}$, all the packet is protected using a chaotic digital watermarking algorithm $\mathbb{W}(\cdot)$, see Section III.C. This new protection allows the authentication of the origin of the packet, while public headers are still accessible (14).

$$\Im_{mark} = \mathbb{W}\left(\Im_{cap}\right) \tag{14}$$

Finally, the market packet $\Im_{mark}$ is sent to the next device. Any device in the subset (intersection) $\Phi_{next} \cap \Phi^{i}_{\text{local}}$ could be selected as next device. The device will be selected using a uniform random variable (15), so every possible device has the same probability to be the next hop. Then, no device in the network can learn about the entire path; and only physically connected devices may know the previous and the following hop in the transmission path (REQ#1).

$$P\left(\phi_i\right) = \frac{1}{card\left\{\Phi_{next} \cap \Phi^{i}_{\text{local}}\right\}} \quad \forall \phi_i \in \Phi_{next} \cap \Phi^{i}_{\text{local}} \tag{15}$$

Figure 5 shows a flowchart describing the entire routing algorithm.

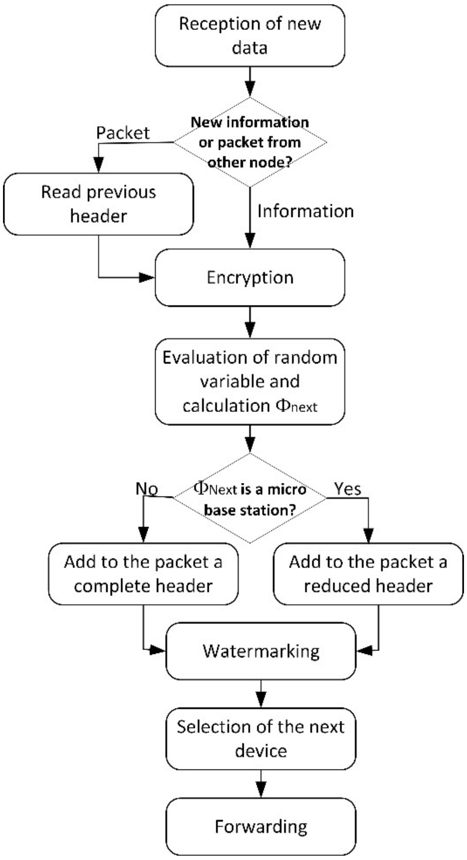

Reception of new data
New information or packet from other node?
Packet
Information
Read previous header
Encryption
Evaluation of random variable and calculation Φnext
ΦNext is a micro base station?
No
Yes
Add to the packet a complete header
Add to the packet a reduced header
Watermarking
Selection of the next device
Forwarding

FIGURE 5. Flowchart for the proposed pseudorandom routing protocol.

As a result, the remote server will receive a packet containing a random sequence (with a random length) of nested encrypted and marked headers and information. The remote server, as any standard IoT application, must know the identity of every device in the IoT subsystem (as well as its configuration). Then, only if the declared identity in the header, the digital watermark and the encryption key are coherent for very nested layer in the packet, the information is considered authentic and valid. On the contrary, an alarm or decision may be done, according to the security policy in the application (we are analyzing this point in this work). Methods employed to validate the watermark and decrypt the packets are explained in Section III.C.

After processing, the remote server may send a response to the IoT subsystem. To do that, the server calculates a random path, and creates (by its own) the entire nested watermarked and encrypted packet corresponding to the reverse path. Typically, this path is similar or equal to the reported in the original message (so we guarantee physical connections among nodes are still alive), but it is not mandatory. In this case, the server sets the control flag to the unit, to inform the IoT devices in the subsystem about the destination. In this case, IoT devices (after noticing the control flag is set to the unit) do not proceed as explained before. In this case, the node first validates its own watermark. If it is valid, removes the header indicating

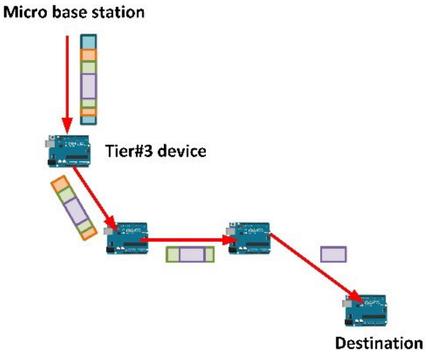


FIGURE 6. Scheme for a returning path.

its identity and decrypts the payload; then reads the next hop and forwards the packet. If the watermark of the encryption is not valid, the packet is automatically discarded. This process is repeated through the entire path, until the destination IoT device receives the message (see Figure 6).

C. DATA ENCRYPTION AND DIGITAL WATERMARKING

The proposed pseudo-random routing protocol is supported by two basic cryptographic algorithms: a chaotic encryption mechanism and a chaotic digital watermarking technique.

Chaotic techniques are highly dependent on the selected dynamic. Dynamics may be discrete (based on an iteration function) or continuous (composed of a system of differential equations). Typically, discrete chaos is much simpler and lighter to implement and compute, although it is less complex and less unpredictable (what reduces its cryptographic applications). This is the typical approach we can find in IoT proposal [36]. To increase the chaos complexity in discrete dynamics, coupled systems have been proposed [37], including complex functions such as the square root. However, this approach is highly affected by the precision of the microprocessor, which is typically reduced in IoT nodes. On the other hand, continuous chaos present better cryptographic properties, with a higher Lyapunov exponent [35] (i.e. signals are more unpredictable), a larger dimension (never lower than two, because of the Poincaré-Bendixson theorem [38]) and a richer catalogue of behaviors. Nevertheless, most complex continuous dynamics are difficult to manage numerically, and they can diverge easily [39].

Therefore, in this proposal we are considering a continuous chaotic dynamic, instead of typical discrete functions, but with a smooth behavior, with no discontinuities, a good numerical behavior and non-linearities easy to compute (such as products or power functions). In particular, we employ the standard Lorenz dynamic (16).

$$\dot{\bar{S}} = \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix} = F\left(\bar{S}\right) = \begin{pmatrix} c_1(y-x) \\ c_2x - y - xz \\ xy - c_3z \end{pmatrix} \tag{16}$$

This dynamic represents, in a numerical and simplified manner, the unpredictable evolution of the atmosphere weather. Signals x, y and z are non-periodic unpredictable signals if parameters c_1, c_2 and c_3 are adequately selected. Figure 7 shows the bifurcation diagram for signal x and bifurcation parameter c_1. Besides, Table 1 shows the maximum value of the Lyapunov exponents for the Lorenz dynamic reported nowadays. As can be seen, the complexity and randomness of signals is very high, and they present a chaotic behavior for almost every possible value of the parameters.

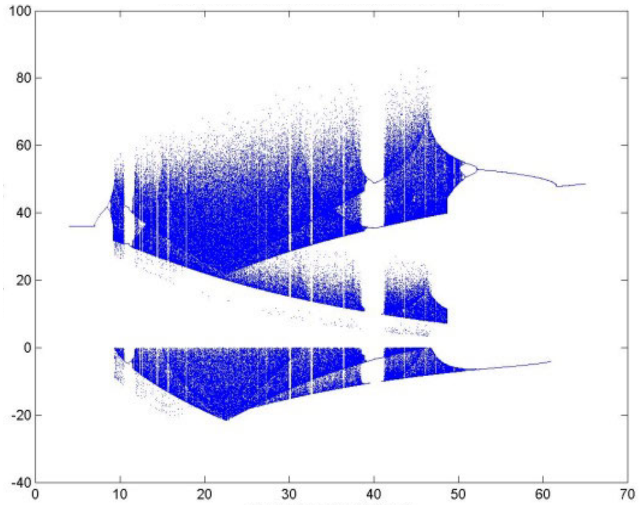


FIGURE 7. Bifurcation diagram of the Lorenz dynamic.

TABLE 1. Lyapunov exponents for the Lorenz dynamic.

Configuration (c_1, c_2, c_3)	Lyapunov exponents
$\left(10, 28, \frac{8}{3}\right)$	$\{0.906\ ,0\ , -14.572\ \}$
$\left(10, 175, \frac{8}{3}\right)$	$\{2.2402\ , 0, -15.9006\}$

However, continuous dynamics cannot be directly integrated to obtain chaotic signal using only digital microprocessors. Therefore, in this work, we are integrating the dynamic to obtain the chaotic signals through four order Runge-Kutta numerical method (17).

$$\begin{aligned} \overline{S_{t+1}} &= \begin{pmatrix} x_{t+1} \\ y_{t+1} \\ z_{t+1} \end{pmatrix} = \overline{S_t} + \frac{1}{6}h\left(\overline{k_1} + 2\overline{k_2} + 2\overline{k_3} + \overline{k_4}\right) \\ \overline{k_1} &= F\left(\overline{S_t}\right) \\ \overline{k_2} &= F\left(\overline{S_t} + \frac{1}{2}h\overline{k_1}\right) \\ \overline{k_3} &= F\left(\overline{S_t} + \frac{1}{2}h\overline{k_2}\right) \\ \overline{k_4} &= F\left(\overline{S_t} + h\overline{k_3}\right) \end{aligned} \tag{17}$$

Before operating with the chaotic signals, it is necessary to select and calculate two important data: the value of the bifurcation parameters (c_1, c_2, c_3) and the initial conditions (x_0, y_0, z_0) to trigger the Runge-Kutta method. These are the secret keys of the proposed encryption and watermarking mechanisms. These key are generated through a specific

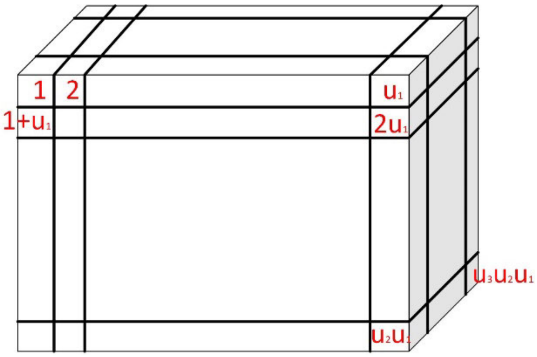


FIGURE 8. Segmentation and labeling in a multidimensional zigzag method.

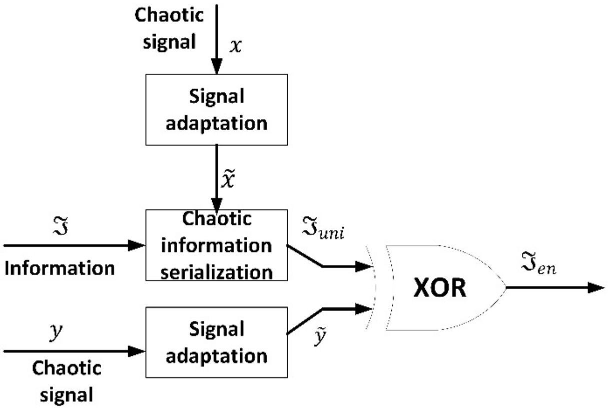


FIGURE 9. Proposed encryption scheme.

method (see Section III.D), which is represented as a set of two vector modulation functions, which take as input the number of packets exchanged by the encrypting node with the remote server b_{server}, and the number of hops already performed by the packet, n (18).

$$\begin{aligned}(c_1, c_2, c_3) &= \mathfrak{C}\left(b_{server}, n\right)\\(x_0, y_0, z_0) &= F_0\left(b_{server}, n\right)\end{aligned}\tag{18}$$

With these secret keys, in each iteration, the Runge-Kutta method generates a new chaotic sample of each signal (three in total).

Now, in general, data packets are unidimensional structures, but raw data $\Im$ generated by nodes may be multidimensional (for example, enriched video streams may include more than three dimensions). Then, the first step in the encryption mechanism is to transform multidimensional information in a one-dimensional stream. This process, in our proposal follows, also, an unpredictable (chaotic) sequence, so the entropy of the encrypted signal goes up and the final encryption is stronger.

First, multidimensional information $\Im$ is divided into $\mathfrak{U}$ different segments (19); being u_i the number of segments in which the i-th dimension is divided into, and D the number of dimensions.

$$\mathfrak{U} = u_1 \times \ldots \times u_i \times \ldots \times u_D \tag{19}$$

Then, every segment is labeled with an integer number in the range $[0, \mathfrak{U} - 1]$, following a multidimensional zig-zag scheme. Figure 8 represents this process. Now, segments are juxtaposed according to values in signal $\tilde{x}$, an adaptation of chaotic signal x to the operating range (20). Each segment, besides, may be serialized using standard techniques to obtain a simple sample.

$$\tilde{x} = \left(x \cdot 10^{\mathfrak{U}}\right) mod\ (\mathfrak{U} - 1) \tag{20}$$

When a unidimensional vector $\Im_{uni}$ is obtained (or when a packet is received), the encryption process may be performed. For this process we have selected a XOR encryption scheme, as it is simple and very lightweight (REQ#2). Basically, the XOR encryption operates using a XOR gate the information vector with a pseudorandom signal, p (21).

$$\Im_{en} = \mathbb{E}\left(\Im\right) = \mathbb{E}\left(\Im_{uni}\right) = \Im_{uni} \otimes p \tag{21}$$

XOR encryption may be highly strong or very weak, depending on the use of this technology. If signal p is a flow of totally random numbers, then, all possible values have the same probability, and this characteristic is transferred to the encrypted signal.

Then, considering the Shannon's information theory, the mutual information between the original and the encrypted information, $I(\Im_{uni}; \Im_{en})$ represents the residual information that remains in the encrypted packet about the original one (22). A simple calculation proves that this quantity is zero as $P\left(\Im_{uni} = \xi_j | \Im_{en} = \xi_i\right) = P\left(\Im_{en} = \xi_i\right)$

$$\begin{aligned}I(\Im_{uni}; \Im_{en}) = &\sum_{\forall \xi_j, \xi_i} P\left(\Im_{uni} = \xi_j, \Im_{en} = \xi_i\right)\\ &\cdot log\left(\frac{P\left(\Im_{uni} = \xi_j | \Im_{en} = \xi_i\right)}{P\left(\Im_{en} = \xi_i\right)}\right)\\ = &\ 0\end{aligned}\tag{22}$$

The encrypted signal, then, contains no information about the real information, and even if an intruder captures traffic for an unlimited time, no information about the private data can be deducted [40]. However, in practice, obtaining totally random signals is not possible and pseudo-random sequences are employed. In our case, we are employing an adapted chaotic signal $\tilde{y}$ (23). Being I_m the maximum value for samples in $\Im_{uni}$.

$$\tilde{y} = p = \left(x \cdot 10^{I_m}\right) mod\ (I_m - 1) \tag{23}$$

$$I_m = max\left\{\Im_{uni}\right\} \tag{24}$$

Figure 9 represents the final scheme for the proposed encryption solution.

The strength of the resulting encryption using a chaotic signal instead of a random one has been already proved in previous works [41]. The description process, as well as the information reconstruction, is very simple, as both algorithms

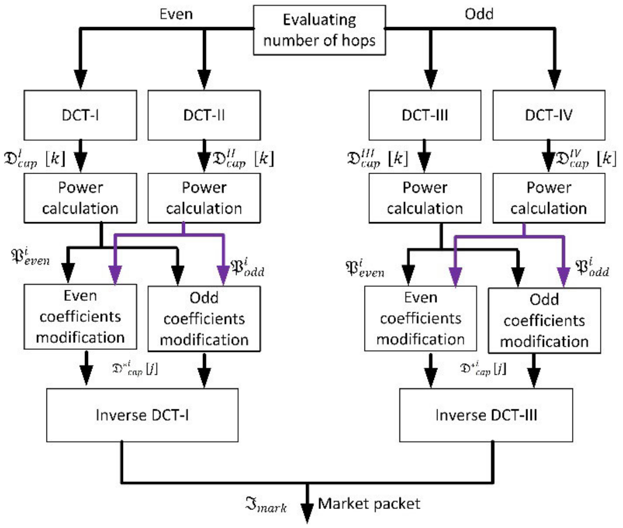

Even
Evaluating number of hops
Odd
DCT-I
DCT-II
DCT-III
DCT-IV
$\mathfrak{D}^{I}_{cap}[k]$
$\mathfrak{D}^{II}_{cap}[k]$
$\mathfrak{D}^{III}_{cap}[k]$
$\mathfrak{D}^{IV}_{cap}[k]$
Power calculation
Power calculation
Power calculation
Power calculation
$\mathfrak{P}^{i}_{even}$
$\mathfrak{P}^{i}_{odd}$
$\mathfrak{P}^{i}_{even}$
$\mathfrak{P}^{i}_{odd}$
Even coefficients modification
Odd coefficients modification
Even coefficients modification
Odd coefficients modification
$\mathfrak{D}^{*i}_{cap}[j]$
$\mathfrak{D}^{*i}_{cap}[j]$
Inverse DCT-I
Inverse DCT-III
$\mathfrak{I}_{mark}$
Market packet

FIGURE 10. Proposed watermarking scheme.

are directly reversible (25). This also guarantees the IoT nodes may perform the decryption process on the server responses.

$$\mathfrak{I} = \mathbb{E}^{-1}\left(\mathfrak{I}_{en}\right) = \mathfrak{I}_{en} \otimes \tilde{y} \tag{25}$$

At this point, two of the three calculated chaotic signals have been employed. Although it is possible to employ the same signal for different purposes, this reduces the entropy of the algorithm and reduces its security. Thus, the chaotic digital watermarking algorithm must be supported by chaotic signal z. Figure 10 describes the proposed watermarking method.

The watermark is designed to be visible, i.e. neither the original element, nor the watermark is required by the receptor to authenticate the origin, only the secret key is needed. Basically, the watermark is embedded in the DCT (Discrete Cosine Transform) domain.

Two different DCT transformations are calculated using the same encapsulated information $\mathfrak{I}_{cap}$, taken from a set of four different DCT transforms: DCT-I (26), DCT-II (27), DCT-III (28) and DCT-IV (29). DCT-II transformation is the transformation usually referred as DCT. Transforms DCT-I and DCT-II are employed with packets that have already performed an even number of hops. Transforms DCT-III and DCT-IV are employed with packets that have already performed an odd number of hops. Each one of these transforms generates a different sequence of coefficients or frequencies, $\mathfrak{D}^{i}_{cap}$.

$$\begin{aligned} \mathfrak{D}^{I}_{cap}[k] = {} & \frac{1}{2}\left(\mathfrak{I}_{cap}[0] + (-1)^{k}\cdot\mathfrak{I}_{cap}[\mathfrak{U}-1]\right) \\ & + \sum\nolimits_{j=1}^{\mathfrak{U}-2} \mathfrak{I}_{cap}[j]\cdot cos\left[\frac{\pi}{\mathfrak{U}-1}jk\right] \\ & k = 0,\ldots,\mathfrak{U}-1 \end{aligned} \tag{26}$$

$$\begin{aligned} \mathfrak{D}^{II}_{cap}[k] = {} & \sum\nolimits_{j=1}^{\mathfrak{U}-1} \mathfrak{I}_{cap}[j]\cdot cos\left[\frac{\pi}{\mathfrak{U}}\left(j+\frac{1}{2}\right)k\right] \\ & k = 0,\ldots,\mathfrak{U}-1 \end{aligned} \tag{27}$$

$$\begin{aligned} \mathfrak{D}^{III}_{cap}[k] = {} & \frac{1}{2}\mathfrak{I}_{cap}[0] + \sum\nolimits_{j=1}^{\mathfrak{U}-1} \mathfrak{I}_{cap}[j] \\ & \cdot cos\left[\frac{\pi}{\mathfrak{U}}\left(k+\frac{1}{2}\right)j\right] \quad k = 0,\ldots,\mathfrak{U}-1 \end{aligned} \tag{28}$$

$$\begin{aligned} \mathfrak{D}^{IV}_{cap}[k] = {} & \sum_{j=1}^{\mathfrak{U}-1} \mathfrak{I}_{cap}[j]\cdot cos\left[\frac{\pi}{\mathfrak{U}}\left(j+\frac{1}{2}\right)\left(k+\frac{1}{2}\right)\right] \\ & k = 0,\ldots,\mathfrak{U}-1 \end{aligned} \tag{29}$$

Now, power accumulated into even ($\mathfrak{P}^{i}_{even}$) and odd ($\mathfrak{P}^{i}_{odd}$) frequencies in the different DCT transforms is calculated (30).

$$\begin{aligned} \mathfrak{P}^{i}_{even} &= \sum_{j=0}^{\frac{\mathfrak{U}-1}{2}} \left|\mathfrak{D}^{i}_{cap}[2j]\right| \\ \mathfrak{P}^{i}_{odd} &= \sum_{j=0}^{\frac{\mathfrak{U}-2}{2}} \left|\mathfrak{D}^{i}_{cap}[2j+1]\right| \quad i \in \{I, II, III, IV\} \end{aligned} \tag{30}$$

Now, to embed the watermark, we enforce the DCT coefficients to meet new restrictions (31), where $\tilde{z}$ is a normalized chaotic signal (32) and β is a real parameter to control the strength of the mark. To ensure these new restrictions are met, DCT coefficients in DCT-I (or DCT-III, depending on the case) transform are modified in a homogeneous manner (33) obtaining the sequence $\mathfrak{D}^{*i}_{cap}$.

$$\begin{aligned} & \left|\mathfrak{P}^{i}_{even} - \mathfrak{P}^{i+1}_{even}\right| \\ & \quad = \beta\tilde{z}_{t} \end{aligned} \tag{31}$$

$$\begin{aligned} & \left|\mathfrak{P}^{i}_{odd} - \mathfrak{P}^{i+1}_{odd}\right| \\ & \quad = \beta\widetilde{z_{t+1}} \quad i \in \{I, III\} \\ & \tilde{z} = \frac{z}{max\{z\}} \end{aligned} \tag{32}$$

$$\begin{aligned} & \mathfrak{D}^{*i}_{cap}[j] \\ & \quad = \begin{cases} \mathfrak{D}^{i}_{cap}[j] + \dfrac{\mathfrak{P}^{i+1}_{even} + \beta\tilde{z}_{t} - \mathfrak{P}^{i}_{even}}{\mathfrak{U}} & if\ j\ even \\ \mathfrak{D}^{i}_{cap}[j] + \dfrac{\mathfrak{P}^{i+1}_{odd} + \beta\widetilde{z_{t+1}} - \mathfrak{P}^{i}_{odd}}{\mathfrak{U}} & if\ j\ odd \end{cases} \\ & \quad i \in \{I, III\} \quad j = 0,\ldots,\mathfrak{U}-1 \end{aligned} \tag{33}$$

Using the corresponding inverse DCT transformation on modified DCT coefficients $\mathfrak{D}^{*i}_{cap}$, the watermarked packet $\mathfrak{I}_{mark}$ is obtained.

In order to validate the watermark, it is only necessary to evaluate if the corresponding DCT coefficients meet the proposed restrictions, according to the secret signal $\tilde{z}$.

D. KEY GENERATION

At this point, in order to clarify the proposed technology behavior, we must explain how secret keys for the chaotic encryption and watermarking algorithms are obtained.

Both, initial conditions and bifurcation parameters are obtained through two vector modulation functions (18),

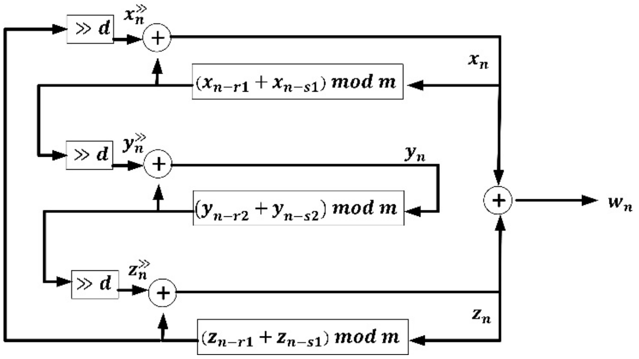

$\gg d$
$x_n^{\gg}$
x_n
$(x_{n-r1}+x_{n-s1})\ mod\ m$
$\gg d$
$y_n^{\gg}$
y_n
$(y_{n-r2}+y_{n-s2})\ mod\ m$
w_n
$\gg d$
$z_n^{\gg}$
z_n
$(z_{n-r1}+z_{n-s1})\ mod\ m$

FIGURE 11. Block diagram of the Trifork generator.

which (basically) consist of two independent lightweight pseudorandom number generators (PNRG). The objective of this approach is to maximize the entropy of the obtained keys.

Although many different PNRG have been reported, in this work we have selected Lagged Fibonacci Generators (LFGs), as they can generate number sequences with a high entropy and using only simple operation such as binary addition, subtraction, mod-m multiplication and/or bitwise exclusive-or (XOR). In particular, we employ the Trifork generator (34) as it has been proved to have a good performance when implemented in resource constrained devices [40]. In particular, Trifork generator has been proved to pass all randomness tests for PRNG proposed by the National Institute of Science and Technology (NIST). Figure 11 shows the implementation scheme of the Trifork generator. As said, two different Trifork generator w_t^{birf} and w_t^{init} are considered in the key generation module (one per each modulation function).

$$\begin{aligned}
x_t &= ((x_{t-r1}+x_{t-s1})\ mod\ m) \oplus z_t^{\gg} \\
y_t &= ((y_{t-r2}+y_{t-s2})\ mod\ m) \oplus x_t^{\gg} \\
z_t &= ((z_{t-r3}+z_{t-s3})\ mod\ m) \oplus y_t^{\gg} \\
x_t^{\gg} &= ((x_{t-r1}+x_{t-s1})\ mod\ m) \gg d \\
y_t^{\gg} &= ((y_{t-r2}+y_{t-s2})\ mod\ m) \gg d \\
z_t^{\gg} &= ((z_{t-r3}+z_{t-s3})\ mod\ m) \gg d \\
w_t &= x_t \oplus z_t
\end{aligned} \tag{34}$$

Parameters r_1, s_1, r_2, s_2, r_3,s_3, d and m may be freely selected, in order to create different modulation functions. The seed employed to initiate the Trifork generator includes a large sequence of R samples (35), which must be carefully selected to guarantee a good performance of the PRNG.

$$R = max\,\{r_1, s_1\} + max\,\{r_2, s_2\} + max\,\{r_3, s_3\} \tag{35}$$

Basically, modulation functions iterate three times the PRNG each time a new key must be obtained. The sequence of three samples that is obtained is the (vector) secret key (36).

$$\begin{aligned}
(c_1, c_2, c_3) &= \mathcal{C}\,(b_{server}, n) = \left(w_t^{birf}, w_{t+1}^{birf}, w_{t+2}^{birf}\right) \\
(x_0, y_0, z_0) &= F_0\,(b_{server}, n) = \left(w_t^{init}, w_{t+1}^{init}, w_{t+2}^{init}\right)
\end{aligned} \tag{36}$$

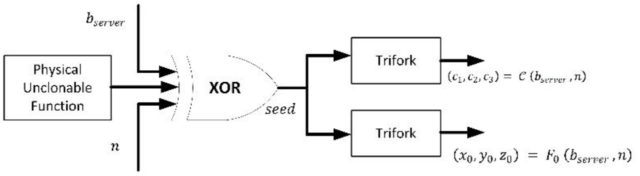

b_{server}
Physical Unclonable Function
XOR
seed
n
Trifork
$(c_1, c_2, c_3) = \mathcal{C}\,(b_{server}, n)$
Trifork
$(x_0, y_0, z_0) = F_0\,(b_{server}, n)$

FIGURE 12. Key generation module.

The seed for the Trifork generator is obtained through an electronic Physical Unclonable Function (PUF). PUF formalize the idea of random physical features employed to create unique number sequences or identifiers, which was firstly known as one-way functions, later as physical random functions and finally as PUF. Electronic PUF consists of measuring an analog signal generated by an electronic system. Electronic systems, especially solid-state components such as transistors, present characteristic behaviors which are impossible to replicate in two elements. The main advantage of these PUF in our scenario, is that electronic devices are always present in IoT devices, contrary to other techniques. In this approach PUF output (usually named as "response"), res_t is unique and unpredictable for each input (usually named as "challenge"), ch_t Many different electronic PUF may be defined, and all of them are adequate to be integrated in the proposed solution.

Hereinafter we are naming the PUF as $f_{puf}\,(\cdot)$ (37)

$$res_t = f_{puf}\,(ch_t) \tag{37}$$

The challenge function ch_t is periodically proposed by the remote server to the IoT nodes. If this challenge is captured by an attacker, no effect is produced in the subsystem, as only the IoT device provided with the specific electronic device producing the expected response may generate the adequate seed. The nature of PUF prevents anyone to clone the response or the function. Responses are pre-calibrated in the remote server by managers, so seeds may be directly synchronized.

In order to enrich the entropy and variability of the seed, the response signal res_t is mixed through XOR functions with b_{server} and n parameters (38).

$$\begin{aligned}
seed &= \big(x_0, \ldots, x_{max\{r_1,s_1\}}, y_0, \ldots, y_{max\{r_2,s_2\}}, z_0, \ldots, \\
&\qquad z_{max\{r_3,s_3\}}\big) \\
&= res_t \otimes b_{server} \otimes n
\end{aligned} \tag{38}$$

Figure 12 show the proposed scheme for the key generation module.

IV. EXPERIMENTAL VALIDATION: METHODS AND METHODOLOGY

In order to validate the proposed technology, an experimental validation was designed and carried out. The experimental validation included three different evaluation methodologies, so all relevant variables and hypotheses are studied and proved. The first methodology (Section IV.A) was focused on a formal verification technique based on rigorous models, in order to discover unexpected behaviors. The second methodology (Section IV.A) was focused on a formal security analysis of the proposed technology. The third methodology (Section IV.B) analyzed the performance of the proposed solution from an operational point of view (processing delay, success rate, etc.).

All experiments were based on a simulation scenario. The proposed scenario consists of a 5G network including one macro base station and three micro base stations. All of them had connectivity with the IoT subsystem. That IoT subsystem represents a Smart Building, where Ambient Intelligence and Ubiquitous computing elements and nodes were deployed. All nodes had WiFi and 5G connectivity. The number of nodes was different and varied during the experiments, but in all cases the devices were distributed in tiers as follows: 10% in tier#1, 60% in tier#2 and 30% in tier#3. All devices were supposed to be homogeneously distributed in the geographical area under study. Devices, base stations and the remote server were simulated through software agents, including the proposed new data authentication and anonymization mechanism. Devices were programmed to generate new information in a random manner, and the remote server was programmed to respond to 40% of the received packets, also in a random manner.

Besides, and additional 10% of malicious nodes injecting false information, or capturing valid data packets or manipulating legitimate communications were introduced. They got active for a variable period in a random manner.

As the proposed technology do not consider losses in wireless networks, the employed model was considered perfect (no losses or interferences).

To perform the experiments, the simulation scenario was implemented and executed using MATLAB 2017a software. All simulations were performed using a Linux architecture (Linux 16.04 LTS) with the following 604 hardware characteristics: Dell R540 Rack 2U, 96 GB RAM, two processors Intel Xeon Silver 4114 2.2G, HD 2TB SATA 7,2K rpm.

All simulation represented an operation time or seventy-two (72) hours. Each simulation was repeated twelve times, and final results were obtained as the average of all partial results.

In order to carry out the proposed experiments, the configuration showed in Table 2 was employed. As PUF, a diode was employed, where the threshold voltage is unclonable and unique for each device. The challenging function, then, is a standard electrical signal. In that scheme, the unclonable behavior is caused by uncontrollable phenomena during semiconductor manipulation, atomic structure, and manufacturing. Typically, nonlinear effects are responsible of most uncontrollable behaviors in solid state devices [43].

On the other hand, as most of the proposed security analyses do not have a clear acceptance criterion, it is very important to compare the obtained results to other existing proposals. In this case, we are taking as reference one of the sparse works where a commercial application of chaos-based encryption mechanisms is described [49]. Not all proposed functionalities in our solution are considered in the proposed reference. Thus, the comparison is only valid for the formal security analysis.

TABLE 2. Values for parameters in proposed new technology.

Parameter	Value	Comments
r_1, r_2, r_3	12	Standard value [40]
s_1, s_2, s_3	10	Standard value [40]
ρ_1	0.3	Congestion 30%
ρ_2	0.7	Congestion 70%
ρ_3	0.45	Congestion 45%
m	1024	10 bits is the standard precision for low-cost microprocessors
d	16	Around 10% of the total number of bits
L	512	Length of the standard Ethernet frame
$u_i \ \forall i$	256	256 segments per dimension
r	10 Mbps	The bitrate of physical links among IoT nodes is 10Mbps

In the performance analysis, when possible, the results are compared to an IoT scenario where a standard routing protocol is employed [50].

A. FORMAL VERIFICATION

In order to detect unexpected behaviors and security flaws, we are analyzing the proposed solution using the model checking technique. Model checking is a formal verification technique designed for systems presenting a probabilistic behavior (such as encryption and watermarking schemes), where the objective is to determine the probability of the system to fulfill a given property.

Basically, model checking is a technique to analyze the system behavior against a given cyber-attack, typically a brute force attack, and, eventually, determine if such attack may be successful. This technique needs a system model and a system specification. Then, a specific software tool (known as model checker) uses a system's implementation to automatically prove if the system meets the requirements of the proposed specification.

As chaotic signals, data packets and paths are probabilistic variables in our proposal, we are using a probabilistic model checker. In particular, we have selected PRISM model checker, because of its extensive employment in security formal verification. We are using PRISM 4.6 [53]. PRISM takes two inputs: an abstract model $\mathfrak{M}$ for the system and a set of logic properties $\{r_i\}$ describing the expected system behavior (specification). We are verifying the resilience of the proposed security solution against a brute force attack performed by a privacy adversary.

A privacy adversary is an enhanced adversary being aware of the complete structure of the proposed system (the PRNG algorithm, the chaotic dynamics, etc.), but not about the secret configuration values. Since all the modules present a statistical but computable and structured behavior, we will show that the adversary will not be able to extract the IoT deployment structure and/or the private information from data packets using only statistical tools with no information about the secret key, at least in a finite amount of time.

Formally, an attacker is defined as a privacy adversary where, in addition to the capabilities of a generic brute force adversary, is able to capture data flowing from the IoT deployment to the 5G base station (and also in the opposite direction) with the intention to improve its knowledge about the IoT system and/or the users' private information using parametric or non-parametric statistical tools.

Based on this definition, two different privacy adversaries may be distinguished. On the one hand, an attacker only uses previous packets to identify the IoT deployment structure of the users' private information is a non-parametric adversary. This adversary uses maximum likelihood algorithms and histogram conformation procedures to extract some information from encrypted packets. These attacking algorithms and models have been extensively employed in different previous works [50], [51]. On the other hand, an attacker that is able to calculate the secret key and configuration parameters using packet data is defined as a parametric privacy adversary. This second type of adversary employs ARX (autoregressive with exogenous input) algorithms or ARMAX (autoregressive-moving average with exogenous input) algorithms. These algorithms have been extensively employed in attacks to most common, and even future, security solutions [55].

In order to verify the resilience of the system, the proposed abstract model included an IoT deployment (as previously described) and an attacker module according to the adversary model described above.

PRISM model checker can work with five different system model types: from Markov decision processes to probabilistic automata. In this case, however, as teletraffic theory is based on exponential distributions we have selected continuous-time Markov chains (CTMCs) as verification model.

An abstract model $\mathfrak{M}$ based on CTMC is a tuple (39) where $\boldsymbol{E}$ is a collection of finite states, $\boldsymbol{E_{init}}$ is a collection of initial states, $\boldsymbol{TR}$ is the transition rate matrix (40) and $\boldsymbol{\Lambda}$ is a labeling (41) with atomic prepositions describing the operations and actions performed at each state. Different languages can be employed to describe these temporal logic rules, but in our case, we have selected Probabilistic Computation Tree Logic (PCTL) as it also matches the probabilistic behavior of the proposed technology.

$$(E, E_{init}, TR, \Lambda) \tag{39}$$

$$TR : E \times E \rightarrow \mathbb{R} \tag{40}$$

$$\Lambda : \mathrm{E} \rightarrow 2^{\mathrm{AP}} \tag{41}$$

In order to simplify the creation of this system model, PRISM allows defining the system in a modular way, so the model checker will later run all modules in parallel. Specifically, each IoT device is described as a new module in the PRISM tool, and for each IoT devices, every module in the proposed technology (see Section III) is also described as an independent module.

In this case we are only considering one logic property r_1 representing the situation when a privacy adversary is able to capture some protected information. Using this PRISM tool, the probability of this logic predicate to take the true value is estimated in the context of the proposed abstract model (42). In this case, we are doing different analyses for different attack durations.

Being $\mathcal{W}_{i,j}$ the i-th watermark introduced in the j-th captured packet by the attacker and $\mathcal{E}_{i,j}$ the i-th encrypted message in the j-th captured packet by the attacker, the logic rule representing the success of a privacy adversary is easy to develop (43). In that way, results from PRISM are directly a formal security validation.

$$p_{attack} = Pr\left\{\mathfrak{M} \models r_1\right\} \tag{42}$$

$$r_1 = \bigcup_{i,j}\left(\mathcal{W}_{\mathrm{i,j}} = removed \ \vee \ \mathcal{E}_{\mathrm{i,j}} = decrypted\right) \tag{43}$$

Different analyses are performed for the two introduced adversary models. Besides, different attacking periods (represented by the number of captured packets) are also analyzed. Furthermore, different significance levels will be also considered, in order to determine whether a privacy adversary will be successful or not.

B. FORMAL SECURITY ANALYSIS

The proposed technology, basically, protects two different information pieces: the personal information collected and sent by IoT system, and the configuration information about the IoT deployment itself. However, both may be exposed, and the entire system compromised, if the chaotic encryption and watermarking solution are broken. To analyze how easy this operation would be for an attacker three different approaches are employed.

The first one was based on Kerckhoff's principle [44], showing that no algorithm or key can be secret for an indefinite amount of time. In this first experiment we are assuming the attacker knows perfectly how the proposed technology works, and (then) all the secrecy depends on the key. We are analyzing the following aspects:

- Key space (***sp***): Cardinality of the total set of possible keys to be employed in the proposed technology. It is analyzed from a theoretical point of view.
- Key sensitivity (***sen***): It represents how different the protected information looks when two similar keys are employed to encrypt the same data. In strong crypto solutions, even slightly different keys produce very different encrypted messages. To evaluate this indicator, we are considering a key $\boldsymbol{K}$ and a difference of $\Delta\boldsymbol{K}$ bits, so for a given information $\boldsymbol{X}$ we obtain the key sensitivity just analyzing the numbers of bits that are different (44), as shown at the bottom of the next page, using the function $\boldsymbol{count}(\cdot)$

In this expression, $\Im_{mark}[\boldsymbol{X},\boldsymbol{K}]$ represents the final encrypted and marked packet with key $\boldsymbol{K}$ and containing information X. Different values for $\Delta\boldsymbol{K}$ are considered.

- Resilience against known-plaintext attack and select-plaintext attack (***res***). It represents how different the protected information looks when two similar information pieces are encrypted using the same key. In strong security solutions, this difference should be relevant (near 100%). To evaluate this indicator, an information piece $\boldsymbol{X}$ and a difference of $\Delta\boldsymbol{X}$ bits is considered (45), as shown at the bottom of the page.

The second approach is based on Shannon's proposals [45]. In this approach, a cryptosystem is only secure if no information from the original message is present in the protected packet. A set of different indicators are being employed to analyze the remaining personal information in the final market packets:

- Correlation (***corr***). It represents the strength of the relation and dependency between the original and the market packet. The standard definition of correlation was employed in this analysis. Strong cryptosystems, where statistical attacks are not possible, present values around zero.
- Entropy (***H***) and mutual information (***I***). The medium information entropy of marked packets was evaluated, as well as the mutual information between anonymized and original packets. The objective is to evaluate the amount of private information that is still present in anonymized packets. For both indicators, the standard Shannon's definitions are employed.
- Histogram variance (***var***). Strong cryptosystems produce encrypted messages with an almost uniform histogram, contrary to clear information that is very non-uniform. If variance is zero, it would be the optimum behavior for a security solution. The standard definition of variance, applied to bytes in the final marked packets, was employed.
- Number of byte change rate (NBCR) and Unified Average Changing Intensity (UACI). These indicators represent how different are bytes in the original packets compared to bytes in the final market packet. NBCR (46) calculates the relative number of bytes that are different in both packets (the original and the marked one), while UACI (47) calculates how much both packets differ byte per byte.

$$NBCR = \frac{count\left(\Im\left[X,K\right];\Im_{mark}\left[X,K\right]\right)}{count\left(\Im_{mark}\left[X,K\right]\right)} \tag{46}$$

$$UACI = \frac{\Im\left[X,K\right]-\Im_{mark}\left[X,K\right]}{count\left(\Im_{mark}\left[X,K\right]\right)\cdot I_m} \tag{47}$$

- Sequence test. It is a statistical test indicating how random the final market packet is. It is based on the chi-square distribution, so the cryptosystem passes the test if the significance value $\boldsymbol{\alpha}$ is higher than the value obtained from the sequence test. In this experiment, a standard implementation of sequence test distributed together with the MATLAB libraries was employed.
- NIST Pseudo-random number generators test suite. As the proposed solution includes a PRNG, the quality tests proposed by NIST for these mechanisms should be considered [56]. However, previous works have already proved that Trifork passes all the 15 different tests included in that suite [47]. Therefore, in this paper, we are considering the keys generated by Trifork PRNG are random enough according to NIST formal security analysis. However, in order to analyze the performance of the chaotic encryption, the NIST tests for ciphers are implemented and run.

Finally, the third approach considered in this formal security analysis, is the Diffie-Hellman's view [46]. In this approach, attackers have access to several pieces or information and packets at different stages (clear information, marked packet, encrypted data, etc.). Then, the proposed security solution may be secure only under certain assumptions. Usually, a standard security analysis should consider three different scenarios:

- Known Message Attack (KMA). In this scenario, the attacker has access to a set of final market packets and their corresponding clear information. Attackers in this scenario will employ a Maximum Likelihood Estimator (MLE) to discover the content of new and future marked packets.
- Known Original Attack (KOA). In this scenario, the attacker has access to a set of final market packets and the corresponding non-marked (but encrypted) packets. In this case, attackers can employ a signal processing technique known as blind source separation (BSS) with no noise. Many papers [48] and implementations of this technique has been reported.
- Watermarked Only Attack (WOA). In this scenario, the attacker has only access to a set of final market packets. In this case, attackers are typically employing the same BSS technology, but for noise environments.

In strong cryptosystem, any of these attacks will be only successful if a very large number of marked and clear packets

$$sen = \frac{count\left(\Im_{mark}\left[X,K+\Delta\mathbf{K}\right];\Im_{mark}\left[X,K\right]\right)+count\left(\Im_{mark}\left[X,K-\Delta\mathbf{K}\right];\Im_{mark}\left[X,K\right]\right)}{2\cdot count\left(\Im_{mark}\left[X,K\right]\right)} \tag{44}$$

$$res = \frac{count\left(\Im_{mark}\left[X+\Delta\mathbf{X},K\right];\Im_{mark}\left[X,K\right]\right)+count\left(\Im_{mark}\left[X-\Delta X,K\right];\Im_{mark}\left[X,K\right]\right)}{2\cdot count\left(\Im_{mark}\left[X,K\right]\right)} \tag{45}$$

are captured. Therefore, the success rate of attackers in each one of these scenarios, for different numbers of captured packets is analyzed, according to the attacking techniques described before. The minimum number of captured packets required to perform a successful attack is usually known as security level.

C. PERFORMANCE EVALUATION

After analyzing the security level provided by the proposed mechanism, we are investigating if operational behavior of this solution is compatible to 5G and IoT scenarios.

Two different experiments were performed. The first one was focused on studying the behavior of the proposed technology from the operational point of view. The second one was focused on evaluating the performance of the new mechanism, in terms of resource consumption.

The first experiment evaluates some relevant security indicators. To do that, different simulations varying the number of nodes in the IoT deployment were developed. In this experiment, information about the success rate in the detection of malicious or manipulated packets was collected and the success rate was calculated.

The second experiment evaluates the performance of the proposed technology, in terms of resource consumption. Information about the network congestion caused by the pseudo-random routing protocol, the processing delay and the memory consumption was collected. The global temporal and spatial order of the proposed technique was evaluated. Besides, the number of operations per time unit was also measured, so the relation between the provided security level and the computing complexity is also analyzed.

V. RESULTS

In this Section, results for the previously described experiments are provided.

A. FORMAL VERIFICATION

Results obtained from PRISM model checker are presented in Table 3. Table 3 represents the higher significance for which the PRISM model checker shows the model do not fulfill the proposed specification (i.e. the attacker access to the private information and then the proposed solution is secure).

As can be seen, a privacy adversary will not be successful with a probability 99.95% even if large amounts of packets are captured. This conclusion applies to both adversary types: parametric and non-parametric. This good behavior is due to the randomly nested encrypted and watermarked packets. This scheme guarantees that packets cannot be constructively analyzed together, as they may have followed different paths and, as a consequence, they do not share any information and are independent events. However, any attacker cannot know which packets are related or not, so the privacy adversary may rarely be successful.

The only difference appears for parametric privacy adversaries, and very large number of captured packets. This kind of adversary, as more information accumulates, more it can enrich the ARX/ARMAX algorithms, reaching slightly greater successful rates. However, even in that case, the privacy adversary will not be able to break the proposed security mechanism in 99.9% of cases.

To address the potential risk of parametric privacy adversaries, the challenge introduced in the PUF may be updated periodically, to ensure the secrecy of the system configuration.

TABLE 3. Results from the formal verification experiments.

Captured packets (10^2)	Adversary model	
	Non-parametric privacy adversary	Parametric privacy adversary
1	***	***
5	***	***
10	***	***
25	***	***
50	***	***
100	***	***
350	***	***
750	***	***
1000	***	**

Significance level: * 0.01 ** 0.001 *** 0.0005

B. FORMAL SECURITY ANALYSIS

We are first analyzing results from experiments considered in the Kerckhoff's approach.

In the proposed scheme, the secret key is composed of two three-dimensional vectors, where each component is an integer number where the maximum value is $m - 1$. Then, the key space for each node sp_i may be easily calculated as the number of permutations with repetition that may be created (48). As a final market packet goes through γ nodes, the final key space for a market packet can be obtained as a product (49).

$$sp_i = m_i^6 \tag{48}$$

$$sp = \sum_{i=1}^{\gamma} m_i^6 \tag{49}$$

These expressions are similar to previous reported technologies [49], [51], where the key space follows a potential function, or even exponential if different nested algorithms are employed. We can conclude, then, that the key space is strong enough and secure.

Figure 13 shows the evolution of key sensitivity for different values of $\Delta\mathbf{K}$ and γ parameters.

As can be seen, as the difference in considered keys goes up, the difference in the market packets grows exponentially. However, the growing rate is higher as the number of nodes involved in the packet transmission, γ, goes up. As several nested encryption processes are developed, the difference in the final market packet is obviously higher when more nodes process the packet. Any case obtained results are coherent with previously reported works, which are considered to present a good security behavior. As can be seen, if only one node is involved, the performance is equivalent to commercial technologies employed nowadays where only one encryption

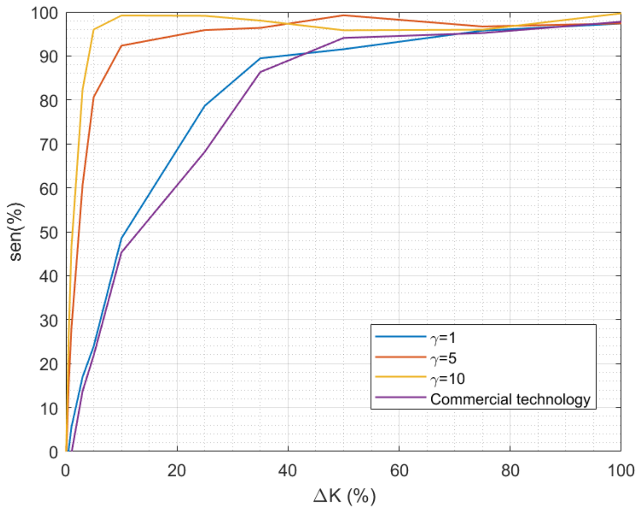


FIGURE 13. Key sensitivity test: results.

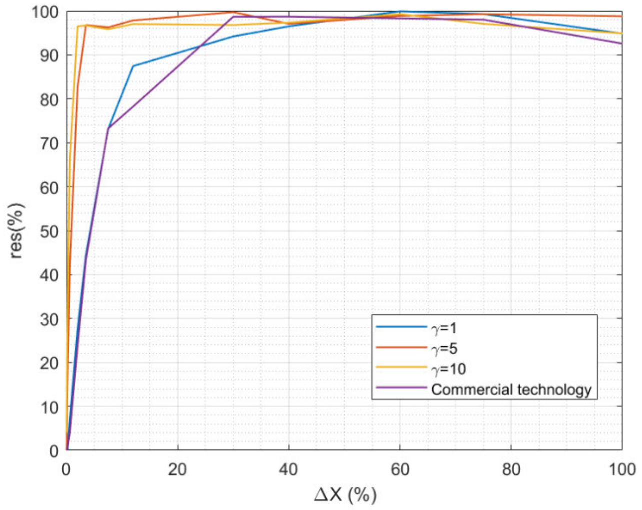


FIGURE 14. Resilience against known-plaintext attack and select-plaintext attack: results.

and watermarking process is included. For a standard difference between similar keys of 20%, in this initial situation we are obtaining a key sensitivity of 65% (approximately).

For this same difference between keys, if at least five nodes are involved in the packet transmission, difference between final market packets is around 95%. This result is also coherent with previously reported works [51], showing that the proposed solution is secure and key sensitive.

Figure 14 shows the resilience against known-plaintext attack and select-plaintext attack in the proposed solution, for different values of $\Delta\mathbf{K}$ and γ parameters.

As can be seen, the tendencies and evolution of system resilience are pretty similar to key sensitivity evolution. This is coherent with XOR encryption, where key and private information are combined through a binary operation meeting the commutative property. Thus, the system behavior changes in the same way if key or private information changes. Variation in the particular values seen in Figure 14 and Figure 13 may be explained as information packets are larger than keys, so percentages are lower. The same results are obtained in other previous works [51], where the system resilience for a standard variation of 20% is always above 90%. As a conclusion, the proposed system is resilient against the known-plaintext attack and the select-plaintext attack.

TABLE 4. Shannon's cryptoanalysis for the proposed solution.

Parameter	Theoretical value	Experimental value
Information entropy	10	9.689
Mutual information	0	0.435
Correlation	0	-0.097
Histogram variance	0	3.82
NBCR	100	99.61
UACI	--	32.94
Sequence test (α = 3)	≤ 3	0.985

Table 4 shows the results obtained for all indicators considered in the Shannon's cryptoanalysis, as well as the expected optimal values.

The information entropy may be theoretically calculated through the expression proposed by Shannon (39). As we have defined a symbol length of 10 bits, and considering that (in a totally encrypted data flow) all symbols have the same probability, the expected information entropy for the strongest possible encryption scheme is $H = 10$. Using the same theoretical procedure, the mutual information may be calculated (40). In this case, the expected value is zero. The same reasoning may be done for correlation and histogram variance. For NBCR, the ideal value is 100%, while for UACI the cryptosystem is better as higher this value is.

As can be seen, deviation of experimental values from idea theorical values is, for all cases, around 3%. This error is small enough to consider negligible the amount of private information that is still present in the encrypted and watermarked packets. Besides, the value obtained from the sequence test is much smaller than the proposed significance level, so randomness in the final market packets is high enough. The values, besides, are similar to the ones provided by other chaos-based mechanisms [49], [51], although an improvement around 1% may be identified. The conclusion is that the proposed anonymization and authentication technology is strong enough from the security point of view.

Finally, Table 5 shows the results obtained from the tests defined in the NIST suite. As can be seen, the proposed solution passes all (15) the proposed tests, as obtained scores are always above 95%, and results are similar and coherent to previously reported works (see Table 4). Considered significance level was $\alpha = 0.01$. Thus, any behavior far away

TABLE 5. Results from NIST statistical tests.

Test	Score (proportion)	
	Proposed technology	Commercial technologies
Random Excursions	120/122	119/122
Cumulative sums	197/200	195/200
Random Excursions Variant	121/122	122/122
FFT	195/200	197/200
Runs	196/200	196/200
Rank	200/200	199/200
Longest Run	196/200	197/200
Block Frequency	197/200	196/200
Approximate entropy	198/200	197/200
Non-Overlapping Template	197/200	198/200
Linear Complexity	197/200	198/200
Serial	197/200	195/200
Frequency	197/200	195/200
Universal	195/200	194/200

from perfectly random protected packets has been detected. In conclusion, we can confirm the proposed solution is secure against statistical attacks.

Finally, Figure 15, 16 and 17 show the results for Diffie-Hellman's view. As can be seen, the worst scenario is caused by Known Original Attacks, where only five thousand captured packets are required to perform a successful attack (success probability around 90%). Then, the security level of the proposed solution against KOA is $5 \cdot 10^3$. This number, any case, is coherent with other previously reported works [52] and commercial solutions (see Figure 17), so we can confirm the proposed mechanism is robust in KOA scenarios.

The same situation may be seen in KMA scenarios, where obtained results are coherent with the stat of the art although in this case the security level grows up a 300%, up to $15 \cdot 10^3$, approximately. However, in WOA scenarios we have reached an important improvement in the security level, around 30% compared to previously reported technologies in the state of the art. Globally, the security level (approximately, $10 \cdot 10^3$) is between the one for KMA and the one for KOA scenarios. Any case, it is a high enough value to consider the proposed solution is secure.

In summary, according to Diffie-Hellman's view the proposed technology is also secure and valid.

C. PERFORMANCE EVALUATION

Figure 18 shows the evolution of the detection success rate. Failure rate is disaggregated, distinguishing between false positive detections and false negative detections.

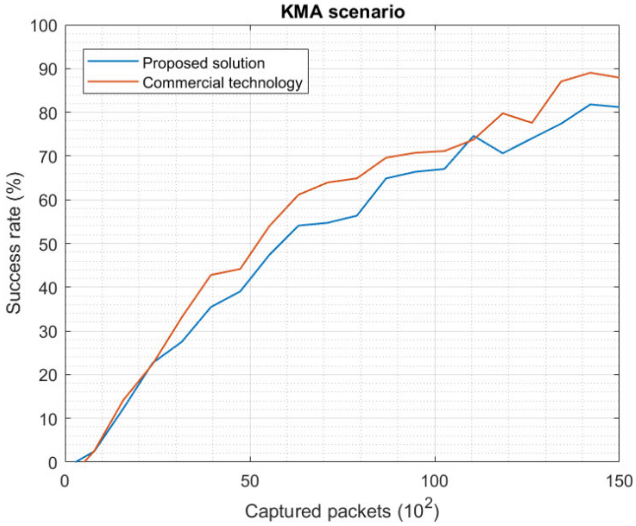


FIGURE 15. Diffie-Hellman's analysis. KMA scenario: results.

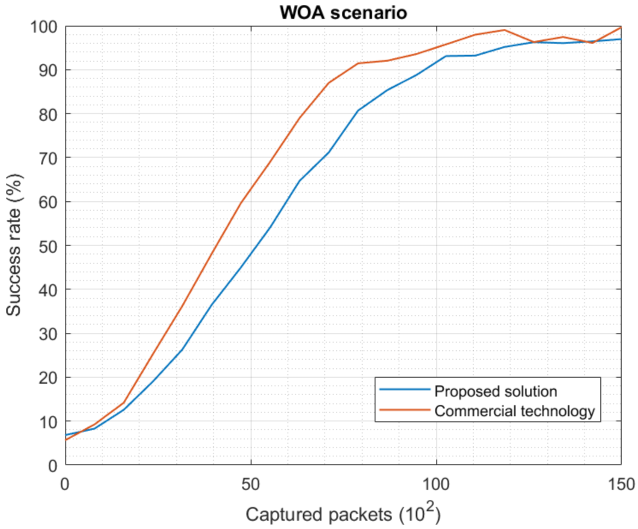


FIGURE 16. Diffie-Hellman's analysis. WOA scenario: results.

As can be seen, the success rate (packets correctly classified) is around 90% for all cases. It slightly decreases when the number of devices in the IoT subsystem goes above one hundred (100). Before this limit, the success rate is near 100% (exactly 97%), but for values above one hundred, the success rate is in the environment of 100%. This evolution may be caused by numerical errors, as the average tend to fluctuate as the number of realizations goes up, until the final and stable value is reached.

On the other hand, false positive detections represent around 2%, and false negative are around 10%. Thus, most wrongly classified packets are, in fact, false negative detections, i.e. malicious datagrams that are not correctly authenticated.

Any case, the obtained values are coherent with other authentication techniques for IoT deployments, improving previously reported results up to 10% [8].

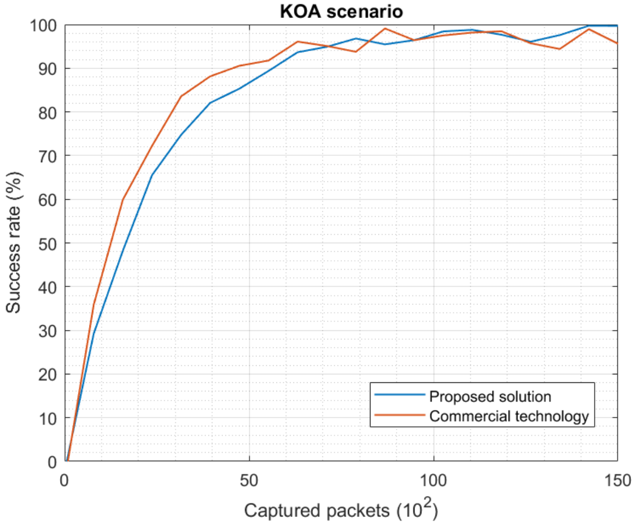


FIGURE 17. Diffie-Hellman's analysis. KOA scenario: results.

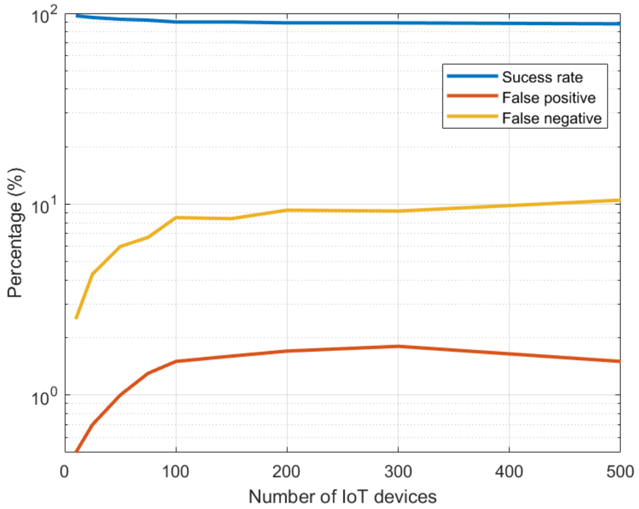


FIGURE 18. First experiment: success rate, false positive and false negative.

Figure 19 shows the evolution of the congestion level in all tiers in the IoT subsystem, as the number of devices in the deployment goes up. Results are compared to congestion in a standard IoT network, where a common routing protocol is employed [50].

As can be seen, the proposed mechanism to guarantee the different tiers are not congested above the security levels works perfectly, and (in all cases) the level of congestion remains below the proposed limits.

It is interesting to see that, in tier#1, congestion grows up with the number of IoT nodes, although tends to get stable between 0.25 and 0.3 when the number of devices is large enough. This behavior may be explained as packets in the tier are only generated (or received) by devices in tier#1. Thus, as the number of devices goes up, more packets are produced, and congestion tends to grow.

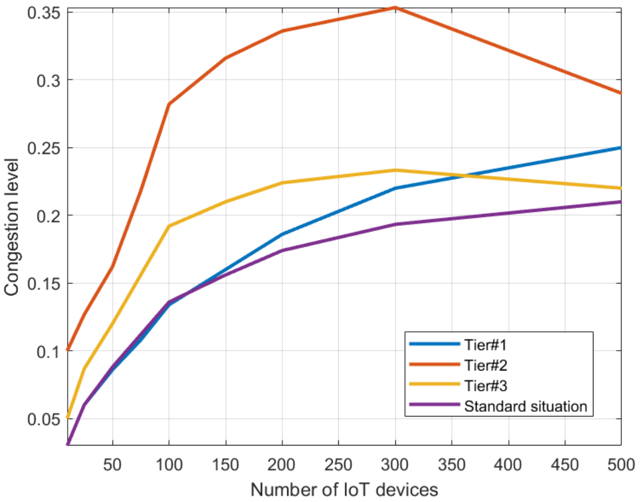


FIGURE 19. Second experiment: congestion level.

On the other hand, congestion in tier#2 and tier#3 presents a convex behavior: the curve reaches a maximum and, after that, goes down. This behavior is especially relevant in tier#2. In fact, as the number of devices goes up, more packets tend to circulate in the network, and congestion is algo higher. However, after exceeding a certain critical value, the available resources are higher than the increase in the number of packets in the deployment, and congestion decreases. This phenomenon is clearer in tier#2, as it manages packets from all tiers in the IoT subsystem. However, the same reasoning may be applied to devices and congestion in tier#3.

As the number of simulations employed to calculate the results is above ten, the internal validity of the experiment is very high, and alternative hypotheses are scarcely probable.

Compared to a standard situation, where a common routing protocol is employed, the congestion level is obviously lower, as packets are not circulating around the network. In fact, the behavior of tier#1 is pretty similar to a standard IoT network, as devices in this tier are focused on transmitting packets outside the layer as soon as possible. Any case, the reached congestion levels are acceptable, especially if we consider the great improvement in security and privacy that is obtained.

Figure 20 shows the evolution in the processing and transmission delay for the proposed mechanism (i.e. time from the information is generated until it is received in the remote server). As previously, the result is compared to the performance of an IoT network where a standard routing protocol is employed.

As can be seen, the processing delay follows also a convex evolution, induced by the evolution of congestion in the different tiers and the IoT subsystem in general. The observed values (from 3 seconds in the lower point, to 10 second in the worst case) are coherent with the performance of resource constrained devices. Any case, the temporal order of the solution with respect the number of IoT nodes may be analyzed. Among well-known functions, the closets is $n \cdot log\,(n)$. However, it seems that the processing delay tends

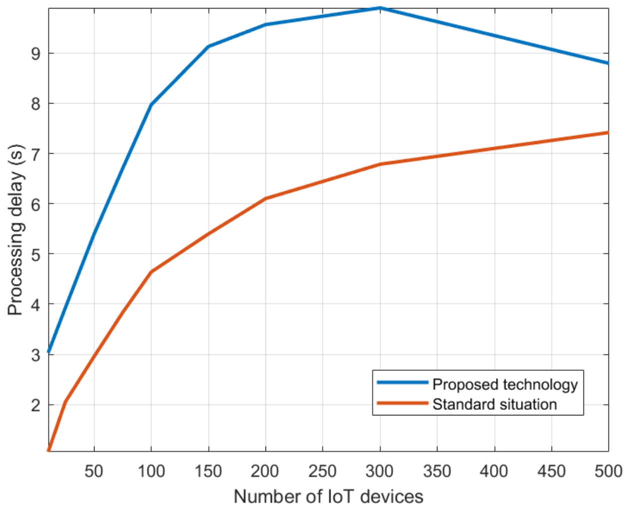


FIGURE 20. Second experiment: processing delay.

to a stable value after going above a certain value, although in the proposed experiment this behavior is not clearly shown.

On the other hand, processing delay is clearly above the required processing time for an IoT network where no encryption or anonymization technologies are deployed. The processing delay increases up to 50% in the worst case, compared to a common IoT deployment, although this percentages reduces as the number of IoT goes up. Any case, it is expected that there is going to be always below the results for the proposed mechanism. Nevertheless, although these values may seem high, the absolute numbers are acceptable, and the proposed solution is adequate for practical scenarios.

The obtained results, any case, are analyzed at network level. If additional layers (application or session layers, for example) are considered, the final Quality-of-Experience may vary (and, in general, decrease) because of noel protocol overheads, new congestion control mechanisms, etc. Besides, although the physical protocol has been also considered in the proposed simulations (and, so, effects such as interferences, Signal-to-Noise Ratio, electromagnetic noise, etc.), other phenomena may affect the obtained results. For example, a geographically unfavorable environment, complex climatic conditions, sparse deployments, or isolated devices can cause the network congestion and, overall, the processing delay to deteriorate.

Any case, although quite exhaustive, the proposed model at physical level considers a limited number of phenomena and variables. Therefore, the performance of the proposed solution in real scenarios is expected to be slightly lower, and (mainly) more fluctuant. This is probably the highest threat to the external validity. However, the average behavior is expected to be very similar to results showed in this paper.

Finally, it is interesting to analyze the spatial order or the algorithm. However, as the proposed solution is datagram oriented and totally anonymous, the memory consumption is agnostic with respect to the number of nodes in the subsystem. Table 6 shows the results of memory consumption.

TABLE 6. Performance indicators.

Use of RAM	Use of program space	Mathematical operations per minute
19%	9%	43809

As can be seen, the RAM usage is relevant (as many temporal variables must be managed), although is tolerable. Besides, the use of the program space is the average of security solutions for IoT applications [40]. Finally, the number of performed mathematical operations per minute, although high, is similar to values obtained for other encryption or signal processing mechanisms [51]. The required processing capacity, any case, meets the special characteristics for IoT devices and, thus, the proposed solution is adequate for those scenarios.

VI. CONCLUSION

In this paper we proposed a new mechanism to protect, authenticate and anonymize data in IoT systems supported by future 5G networks. The proposed solution employs both digital watermarking techniques and lightweight cryptographic technologies; as well as a pseudo-random routing protocol. To generate keys in a secure and simple manner, physical unclonable functions and pseudo-random number generation based on Lagged Fibonacci generators are employed. Besides, to reduce as much as possible the computational cost of algorithms, chaotic dynamics are considered, specifically, the Lorenz dynamic.

Results show the proposed solution provides a good security level and the detection success rate is around 90%, and the solution guarantees that secure congestion levels are never exceeded in the IoT subsystem.

Future works will evaluate the performance of the proposed solution using real devices and applications. Besides, the dependence of the proposed mechanism on the selected PUF should be also analyzed.

REFERENCES

[1] B. B. Sanchez, A. Sanchez-Picot, and D. Sanchez De Rivera, "Using 5G technologies in the Internet of Things handovers, problems and challenges," in *Proc. 9th Int. Conf. Innov. Mobile Internet Services Ubiquitous Comput.*, Jul. 2015, pp. 364–369.

[2] B. Bordel, R. Alcarria, T. Robles, and D. Martín, "Cyber–physical systems: Extending pervasive sensing from control theory to the Internet of Things," *Pervas. Mobile Comput.*, vol. 40, pp. 156–184, Sep. 2017.

[3] B. Bordel, R. Alcarria, T. Robles, and D. Sánchez-de-Rivera, "Service management in virtualization-based architectures for 5G systems with network slicing," *Integr. Comput.-Aided Eng.*, vol. 27, no. 2, pp. 77–99, 2020.

[4] B. Bordel, R. Alcarria, D. Sánchez-de-Rivera, and A. Sánchez, "An inter-slice management solution for future virtualization-based 5G systems," in *Proc. Int. Conf. Adv. Inf. Netw. Appl.* Cham, Switzerland: Springer, Mar. 2019, pp. 1059–1070.

[5] B. Bordel, R. Alcarria, T. Robles, and A. Sanchez-Picot, "Stochastic and information theory techniques to reduce large datasets and detect cyberattacks in ambient intelligence environments," *IEEE Access*, vol. 6, pp. 34896–34910, 2018.

[6] J. C. Gondim, R. de Oliveira Albuquerque, A. C. A. Nascimento, L. G. Villalba, and T.-H. Kim, "A methodological approach for assessing amplified reflection distributed denial of service on the Internet of Things," *Sensors*, vol. 16, no. 11, p. 1855, Nov. 2016.

[7] B. Bordel, R. Alcarria, D. Sánchez-de-Rivera, and T. Robles, "Protecting industry 4.0 systems against the malicious effects of cyber-physical attacks," in *Proc. Int. Conf. Ubiquitous Comput. Ambient Intell.* Cham, Switzerland: Springer, Nov. 2017, pp. 161–171.

[8] T. Robles, B. Bordel, R. Alcarria, and D. Sánchez-de-Rivera, "Enabling trustworthy personal data protection in eHealth and well-being services through privacy-by-design," *Int. J. Distrib. Sensor Netw.*, vol. 16, no. 5, May 2020, Art. no. 155014772091211.

[9] M. Pérez-Jiménez, B. Sánchez, A. Migliorini, and R. Alcarria, "Protecting private communications in cyber-physical systems through physical unclonable functions," *Electronics*, vol. 8, no. 4, p. 390, Apr. 2019.

[10] R. H. Weber, "Internet of Things-new security and privacy challenges," *Comput. Law Secur. Rev.*, vol. 26, no. 1, pp. 23–30, 2010.

[11] J. H. Castellanos, D. Antonioli, N. O. Tippenhauer, and M. Ochoa, "Legacy-compliant data authentication for industrial control system traffic," in *Proc. Int. Conf. Appl. Cryptogr. Netw. Secur.* Cham, Switzerland: Springer, Jul. 2017, pp. 665–685.

[12] A. J. Perez, K. G. Rivera-Morales, M. A. Labrador, and I. Vergara-Laurens, "HR-auth: Heart rate data authentication using consumer wearables," in *Proc. 5th Int. Conf. Mobile Softw. Eng. Syst.*, May 2018, pp. 88–89.

[13] H.-W. Ferng and N. M. Khoa, "On security of wireless sensor networks: A data authentication protocol using digital signature," *Wireless Netw.*, vol. 23, no. 4, pp. 1113–1131, May 2017.

[14] J. Sun, W. Wang, L. Kou, Y. Lin, L. Zhang, Q. Da, and L. Chen, "A data authentication scheme for UAV ad hoc network communication," *J. Supercomput.*, vol. 76, pp. 4041–4056, Jun. 2017.

[15] A. Hassan, M. Nihad, and N. Nife, "The Internet of Things privacy," *J. Comput. s Nanosci.*, vol. 16, no. 3, pp. 1007–1018, 2019.

[16] M. Xie, M. Huang, Y. Bai, and Z. Hu, "The anonymization protection algorithm based on fuzzy clustering for the ego of data in the Internet of Things," *J. Electr. Comput. Eng.*, vol. 2017, pp. 1–10, 2017.

[17] J. J. V. Nayahi and V. Kavitha, "Privacy and utility preserving data clustering for data anonymization and distribution on Hadoop," *Future Gener. Comput. Syst.*, vol. 74, pp. 393–408, Sep. 2017.

[18] C. Dwork and A. Roth, "The algorithmic foundations of differential privacy," *Found. Trends Theor. Comput. Sci.*, vol. 9, nos. 3–4, pp. 211–407, 2013.

[19] C. Yin, J. Xi, R. Sun, and J. Wang, "Location privacy protection based on differential privacy strategy for big data in industrial Internet of Things," *IEEE Trans. Ind. Informat.*, vol. 14, no. 8, pp. 3628–3636, Aug. 2018.

[20] M. Malekzadeh, R. G. Clegg, A. Cavallaro, and H. Haddadi, "Mobile sensor data anonymization," in *Proc. Int. Conf. Internet Things Design Implement.*, Apr. 2019, pp. 49–58.

[21] C. Perera, C. McCormick, A. K. Bandara, B. A. Price, and B. Nuseibeh, "Privacy-by-design framework for assessing Internet of Things applications and platforms," in *Proc. 6th Int. Conf. Internet Things*, Nov. 2016, pp. 83–92.

[22] *Internet of Things: Privacy and Security in a Connected Worldm*, Federal Trade Commission, Washington, DC, USA, 2015.

[23] C. Fontaine and F. Galand, "A survey of homomorphic encryption for nonspecialists," *EURASIP J. Inf. Secur.*, vol. 2007, no. 1, 2007, Art. no. 013801.

[24] S. Spiekermann and L. F. Cranor, "Engineering privacy," *IEEE Trans. Softw. Eng.*, vol. 35, no. 1, pp. 67–82, Jan. 2009.

[25] L. Brankovic and V. Estivill-Castro, "Privacy issues in knowledge discovery and data mining," in *Proc. Austral. Inst. Comput. Ethics Conf.*, Jul. 1999, pp. 89–99.

[26] J.-Z. Sun, "Adaptive determination of data granularity for QoS-constraint data gathering in wireless sensor networks," in *Proc. Symp. Workshops Ubiquitous, Autonomic Trusted Comput.*, 2009, pp. 401–405.

[27] R. Roman, J. Zhou, and J. Lopez, "On the features and challenges of security and privacy in distributed Internet of Things," *Comput. Netw.*, vol. 57, no. 10, pp. 2266–2279, Jul. 2013.

[28] E. Damiani, F. Pagano, and D. Pagano, "iPrivacy: A distributed approach to privacy on the cloud," *Int. J. Adv. Secur.*, vol. 4, no. 2, pp. 1–8, 2011.

[29] C. Perera, M. Barhamgi, A. K. Bandara, M. Ajmal, B. Price, and B. Nuseibeh, "Designing privacy-aware Internet of Things applications," *Inf. Sci.*, vol. 512, pp. 238–257, Feb. 2020.

[30] L. Davoli, Y. Protskaya, and L. Veltri, "An anonymization protocol for the Internet of Things," in *Proc. Int. Symp. Wireless Commun. Syst. (ISWCS)*, Aug. 2017, pp. 459–464.

[31] X. Duan and X. Wang, "Authentication handover and privacy protection in 5G hetnets using software-defined networking," *IEEE Commun. Mag.*, vol. 53, no. 4, pp. 28–35, Apr. 2015.

[32] E. Dubrova, M. Naslund, and G. Selander, "CRC-based message authentication for 5G mobile technology," in *Proc. IEEE Trustcom/BigDataSE/ISPA*, Aug. 2015, pp. 1186–1191.

[33] A. Sk and V. Masilamani, "A novel digital watermarking scheme for data authentication and copyright protection in 5G networks," *Comput. Electr. Eng.*, vol. 72, pp. 589–605, Nov. 2018.

[34] P. Mareca and B. Bordel, "An intra-slice chaotic-based security solution for privacy preservation in future 5G systems," in *Proc. World Conf. Inf. Syst. Technol.* Cham, Switzerland: Springer, Mar. 2018, pp. 144–154.

[35] P. Mareca and B. Bordel, "Robust hardware-supported chaotic cryptosystems for streaming commutations among reduced computing power nodes," *Anal. Integr. Circuits Signal Process.*, vol. 98, no. 1, pp. 11–26, Jan. 2019.

[36] A. Akhshani, S. Behnia, A. Akhavan, H. A. Hassan, and Z. Hassan, "A novel scheme for image encryption based on 2D piecewise chaotic maps," *Opt. Commun.*, vol. 283, no. 17, pp. 3259–3266, Sep. 2010.

[37] S. Behnia, A. Akhshani, H. Mahmodi, and A. Akhavan, "A novel algorithm for image encryption based on mixture of chaotic maps," *Chaos, Solitons Fractals*, vol. 35, no. 2, pp. 408–419, Jan. 2008.

[38] J. Mallet-Paret and H. Smith, "The Poincaré-Bendixson theorem for monotone cyclic feedback systems," *J. Dyn. Differential Equ.*, vol. 2, no. 4, pp. 367–421, 1990.

[39] M. P. Mareca and B. Bordel, "Improving the complexity of the lorenz dynamics," *Complexity*, vol. 2017, Jan. 2017, Art. no. 3204073.

[40] B. Bordel, A. B. Orue, R. Alcarria, and D. Sanchez-De-Rivera, "An intra-slice security solution for emerging 5G networks based on pseudo-random number generators," *IEEE Access*, vol. 6, pp. 16149–16164, 2018.

[41] P. Mareca and B. Bordel, "A robust implementation of a chaotic cryptosystem for streaming communications in wireless sensor networks," in *Proc. World Conf. Inf. Syst. Technol.* Cham, Switzerland: Springer, Apr. 2017, pp. 95–104.

[42] S. Shin and T. Kwon, "A privacy-preserving authentication, authorization, and key agreement scheme for wireless sensor networks in 5G-integrated Internet of Things," *IEEE Access*, vol. 8, pp. 67555–67571, 2020.

[43] T. Potteiger and W. H. Robinson, "A one zener diode, one memristor crossbar architecture for a write-time-based PUF," in *Proc. IEEE 58th Int. Midwest Symp. Circuits Syst. (MWSCAS)*, Aug. 2015, pp. 1–4.

[44] A. Kerckhoffs, "La cryptographie militaire," *J. Sci. Militaires*, vol. IX, pp. 5–38, Jan. 1883.

[45] C. E. Shannon, "Communication theory of secrecy systems," *Bell Syst. Tech. J.*, vol. 28, no. 4, pp. 656–715, Oct. 1949.

[46] W. Diffie and M. Hellman, "New directions in cryptography," *IEEE Trans. Inf. Theory*, vol. IT-22, no. 6, pp. 644–654, Nov. 1976.

[47] A. B. Orue, F. Montoya, and L. H. Encinas, "Trifork, a new pseudorandom number generator based on lagged Fibonacci maps," *J. Comput. Sci. Eng.*, vol. 2, no. 2, pp. 46–51, 2010.

[48] C. Jutten and J. Karhunen, "Advances in blind source separation (BSS) and independent component analysis (ICA) for nonlinear mixtures," *Int. J. Neural Syst.*, vol. 14, no. 5, pp. 267–292, Oct. 2004.

[49] G. Vidal, R. Becheikh, R. Rhouma, and S. Belghith, "A commercial application of a chaos-based-stream cipher: Performance and security analysis," in *Proc. 11th Int. Conf. Internet Technol. Secured Trans. (ICITST)*, Dec. 2016, pp. 39–44.

[50] H. Kim, J. Ko, D. E. Culler, and J. Paek, "Challenging the IPv6 routing protocol for low-power and lossy networks (RPL): A survey," *IEEE Commun. Surveys Tuts.*, vol. 19., no. 4, pp. 2502–2525, 4th Quart., 2017.

[51] S. Lian, J. Sun, and Z. Wang, "Security analysis of a chaos-based image encryption algorithm," *Phys. A, Stat. Mech. Appl.*, vol. 351, nos. 2–4, pp. 645–661, Jun. 2005.

[52] P. Bas and G. Doárr, "Practical security analysis of dirty paper trellis watermarking," in *Proc. Int. Workshop Inf. Hiding*. Berlin, Germany: Springer, Jun. 2007, pp. 174–188.

[53] M. Kwiatkowska, G. Norman, and D. Parker, "PRISM 4.0: Verification of probabilistic real-time systems," in *Proc. Int. Conf. Comput. Aided Verification* Berlin, Germany: Springer, Jul. 2011, pp. 585–591.

[54] R. Ye, "A novel chaos-based image encryption scheme with an efficient permutation-diffusion mechanism," *Opt. Commun.*, vol. 284, no. 22, pp. 5290–5298, Oct. 2011.

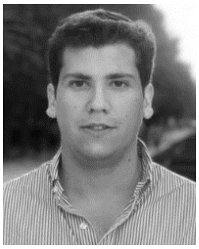
[55] M. Elboukhari, M. Azizi, and A. Azizi, "Analysis of quantum cryptography protocols by model checking," *Int. J. Universal Comput. Sci*, vol. 1, pp. 34–40, May 2010.

[56] A. Rukhin, J. Soto, J. Nechvatal, M. Smid, and E. Barker, "A statistical test suite for random and pseudorandom number generators for cryptographic applications," *Booz-Allen Hamilton*, Mar. 2001.

BORJA BORDEL received the B.S. and M.S. degrees in telecommunication engineering from the Technical University of Madrid, in 2012 and 2014, respectively, and the Ph.D. degree in 2018. He is currently an Assistant Professor with the Computer Science School. His research interests include cyber-physical systems, wireless sensor networks, radio access technologies, communication protocols, and complex systems.

RAMÓN ALCARRIA received the M.S. and Ph.D. degrees in telecommunication engineering from the Technical University of Madrid, in 2008 and 2013, respectively. He is currently an Associate Professor with the Department of Geospatial Engineering, Technical University of Madrid. He has been involved in several Research and Development European and National projects related to Future Internet, the Internet of Things, and Service Composition. His research interests include service architectures, sensor networks, human–computer interaction, and prosumer environments.

TOMAS ROBLES received the M.S. and Ph.D. degrees in telecommunication engineering from the Technical University of Madrid, in 1987 and 1991, respectively. He is currently a Full Professor of telematics engineering with the ETSI Telecommunication, Technical University of Madrid. His research interests include advanced applications and services for wireless networks, also on blockchain-based infrastructures.



MARCOS SÁNCHEZ IGLESIAS received the B.S. degree in IT engineering and the M.S. degree in telecommunication engineering from the Universidad de Castilla-La Mancha, in 2017 and 2020, respectively. He is currently pursuing the Ph.D. degree in IT with the IT Systems Engineering School, UCLM. His research interests include cyber-physical systems, wireless sensor networks, radio access technologies, communication protocols, and complex systems.