AMERICAN UNIVERSITY OF BEIRUT

Designing Physical Layer Security Solutions For Emerging Communication Systems in 5G Networks

by
REEM MOHAMMAD SALIM MELKI

A dissertation
submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy
to the Department of Electrical and Computer Engineering
of the Faculty of Engineering and Architecture
at the American University of Beirut

Beirut, Lebanon
August 2020

AMERICAN UNIVERSITY OF BEIRUT

Designing Physical Layer Security Solutions For Emerging Communication Systems in 5G Networks

by

REEM MOHAMMAD SALIM MELKI

Approved by:

Dr. Karim Kabalan, Professor Electrical and Computer Engineering	Chairperson of Committee
Dr. Ali Chehab , Professor Electrical and Computer Engineering	Advisor
Dr. Rouwaida Kanj, Associate Professor Electrical and Computer Engineering	Member of Committee
Dr. Raphael Couturier, Professor University Bourgogne Franche-Comte, France	Member of Committee

Dr. Yasser Mohanna, Professor — Member of Committee

Lebanese University

Dr. Hassan Noura, Assistant Professor — Member of Committee

Arab Open University

Date of dissertation defense: August 3, 2020

AMERICAN UNIVERSITY OF BEIRUT

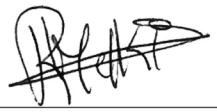
THESIS, DISSERTATION, PROJECT RELEASE FORM

Student Name: *Melki* *Reem* *Mohammad Salim*
Last First Middle

○ Master's Thesis ○ Master's Project ☑ Doctoral Dissertation

☑ I authorize the American University of Beirut to: (a) reproduce hard or electronic copies of my thesis, dissertation, or project; (b) include such copies in the archives and digital repositories of the University; and (c) make freely available such copies to third parties for research or educational purposes.

☐ I authorize the American University of Beirut, to: (a) reproduce hard or electronic copies of it; (b) include such copies in the archives and digital repositories of the University; and (c) make freely available such copies to third parties for research or educational purposes after: **One ___ year from the date of submission of my thesis, dissertation or project.**
Two ___ years from the date of submission of my thesis , dissertation or project.
Three ___ years from the date of submission of my thesis , dissertation or project.



06/08/2020
Signature Date

This form is signed when submitting the thesis, dissertation, or project to the University Libraries

Acknowledgements

I would like to express my sincere and heartfelt obligation towards my advisor Prof. Ali Chehab and co-advisor Prof. Hassan Noura for their active guidance, help, cooperation and encouragement. Without their constant support, this dissertation would not have been possible.

I would like to acknowledge my dissertation committee members, Prof. Karim Kabalan, Prof. Rouwaida Kanj, Prof. Raphael Couturier, and Prof. Yasser Mohanna for their dedicated time and constructive comments.

I also extend my gratitude to the American University of Beirut for this wonderful opportunity and for the continuous support throughout my PhD journey.

Last but definitely not least, special thanks to my mother, siblings, family and friends for their unceasing encouragement, love and care throughout this long academic journey.

An Abstract of the Dissertation of

Reem Mohammad Salim Melki for Doctor of Philosophy
Major: Electrical and Computer Engineering

Title: Designing Physical Layer Security Solutions For Emerging Communication Systems in 5G Networks

Over the past few years, wireless networks have witnessed major advancements in wireless communication technologies, due to 1) the large population growth, 2) the rapid urbanization of different cities around the globe, and 3) the wide deployment of the Internet in people's daily lives. This has triggered a vigorous increase in the amount of traffic in both enterprise and residential networks, and it has motivated researchers and network operators to reconsider current network designs and mobile platforms.

In order to cater for the huge expansion in the wireless industry, the 5G technology has been introduced as the next-generation standard for digital cellular networks. This technology promises vastly increased capacity, reduced latency, better utilization of resources, and faster speeds (data rates). Additionally, the 5G network architecture includes a large heterogeneous panel of interconnected networks and devices, such as Device-to-Device (D2D) and Machine-to-Machine (M2M) networks, small cell access points, network cloud, Internet of Things (IoT), and many more.

One important requirement that needs to be addressed in 5G networks is its security. This is, mainly, attributed to the fact that existing security solutions and cryptographic algorithms can not support the stringent requirements of 5G networks. More specifically, conventional security solutions introduce a considerable overhead in terms of resources and delay (multi-round operations), which is not feasible for constrained devices.

On the other hand, Physical Layer Security (PLS) has, recently, emerged as a promising methodology for enhancing the security of wireless networks, without relying on upper-layer cryptographic techniques. It allows legitimate users to

exchange confidential messages in the presence of adversaries, by simply utilizing the dynamic properties and characteristics of wireless channels. Security in wireless networks has always been addressed separately from the physical layer, due to its uncontrollable random nature. However, with the tremendous advancement in computational capabilities, classical security techniques (static structure) are becoming less secure and the need for new adjustments is becoming more crucial. More and more research has been directed towards studying, understanding and exploiting the highly random nature of wireless networks. As a result, this has paved the way for new security solutions, that are more robust and less complex than current schemes. Moreover, the physical layer is common to all kinds of devices, hence, any security solution at this layer is useful for all heterogeneous devices.

The goal of this PhD dissertation is to design and evaluate novel PLS solutions that guarantee multiple security services with minimum overhead. One important aspect is achieving a good balance between security and performance, in order to ensure the efficient and proper deployment of different state-of-the-art communication systems in 5G networks. Another aspect is providing a complete "security framework" for emerging communication systems and resource-limited devices. This framework consists of several protocols and algorithms, that manage the transferal of information in public wireless networks.

Unlike traditional security solutions, which require multiple rounds of extensive operations, the proposed PLS techniques leverage the random and dynamic properties of wireless channels to achieve robust security using a single round of simple operations. These schemes are classified according to five security services, which are: device authentication, key generation and distribution, data confidentiality, data integrity and source authentication, and data availability. For each security service, several variant schemes are presented and evaluated. The proposed security solutions target different technologies such as NOMA and MIMO. Since OFDM is expected to remain a key enabling technology in emerging and future systems such as 5G networks, PLS schemes (data confidentiality and message authentication) for OFDM systems are also designed and evaluated. Moreover, these methods are compared when applied at two instances; before the Inverse Fast Fourier Transform (IFFT) and after, since it is important to quantify the effect of each case on the security level. The security level and performance gains of the proposed schemes are analyzed using simulations and numerical results. The various obtained results prove the superiority of the proposed solutions over similar existing approaches in the literature.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Wireless communication is one of the most pervasive technologies and is, by far, the fastest growing segment of the communications industry, with nearly 5 billion users accessing only one array of wireless technologies, which is cellular communication [1, 2]. Moreover, this technology has become crucial for a very wide range of applications including 5G networks, Internet of Things (IoT), Wireless Sensor Networks (WSN), banking, social networking, health monitoring and many others [1].

1.1 Problem Formulation

The broadcast nature of wireless transmission has made this technology vulnerable to passive and active attacks, since adversaries are able to capture, decode and recover transmitted signals having sufficient power [3]. Conventionally, communication security is viewed as an independent feature, and it is addressed at the upper layers of the protocol stack by applying traditional cryptographic schemes (data link layer and above) [4]. Some of the well known security protocols are: the Hypertext Transfer Protocol-Secure (HTTPS) which is an adaptation of HTTP for secure communication at the application layer [5], the Transport Layer Security (TLS), which is used to protect the transport layer [6] and the Internet Protocol Security (IPsec), which is designed to secure communication over Internet Protocol (IP) networks [7]. All of the aforementioned security protocols and the available cryptographic algorithms have greatly improved network security, however, it has always been assumed that the physical layer provides an error-free link, which is not the case in practice. For instance, wireless links are more vulnerable to attacks than wired links, since the latter provides dedicated channels between users and, thus, offers better performance in terms of privacy and security. Currently, security protocols operate above the physical layer, which means that the physical layer header is transmitted in plaintext. This allows eavesdroppers to synchronize to the transmitted frames and, hence, recover data.

As such, wireless security is still prone to both active and passive attackers since the underlying structure of the transmitted data is not encrypted and is sent in the clear [4]. Moreover, with the emergence of ad-hoc and decentralized networks, upper layer security techniques have become complex and harder to implement.

1.2 Motivation

Motivated by what state-of-the-art wireless technologies have to offer from increased throughput to enhanced resiliency against failures and by the fact that today's wireless networks still suffer from major security vulnerabilities, Physical Layer Security (PLS) has been recently introduced as a promising candidate to improve the security of wireless communication systems. PLS has received a lot of interest from both academia and industry, and notable progress has been made in terms of 1) understanding the basic physical layer fundamentals and 2) proposing novel ideas and techniques to ensure better wireless network security [8, 1, 9, 10]. Moreover, PLS has the potential to greatly enhance the performance of a very wide range of applications.

The physical layer has the least impact on wireless communication systems (lowest layer) and it is the fastest among all layers (low delay and minimum utilized resources). In other words, any security solution applied at this layer will not modify or affect any functionality at upper layers. Additionally, the physical layer is common to all kinds of devices, which means that any security solution at this layer can be useful to all heterogeneous devices. By default, PLS secures data at the physical layer and all above layers, using only one round of simple operations [11]. Furthermore, the high level of randomness and dynamicity the physical layer possesses, has paved the way for new security solutions, that are more robust and less complex than current schemes. Generally, this is the main motivation behind adopting PLS.

Recently, extensive research work has been presented to design wireless PLS schemes [12, 13]; many challenging but interesting issues remain open for future contributions. Specifically, existing work in this area focuses on exploring different techniques to establish the first line of defense in the security of 5G networks.

1.3 Contributions of Dissertation

In this PhD dissertation, PLS schemes that jointly enhance the security and performance of emerging communication systems in 5G networks, are defined and evaluated. These schemes are divided into five main groups:

- Device (mutual) authentication schemes;
- Key generation and distribution schemes;

- Data confidentiality schemes;
- Source authentication and message integrity schemes;
- Availability schemes.

Examples of communication systems in 5G networks include Internet of Things (IoT) systems, Device-to-Device (D2D) systems, Machine-to-Machine (M2M) systems and many others. Throughout this dissertation, the system model of the IoT system is considered, however, the presented schemes are generic and they can be employed by any other 5G communication system. The contributions of this dissertation are summarized as follows:

- A detailed study on the existing PLS schemes is presented. More specifically, these schemes are categorized into five groups; each group corresponds to one of the previously mentioned security services. Then, each technique is described and discussed in a detailed manner for a better understanding of the PLS method.
- The advantages and weaknesses of each technique are highlighted and ways to overcome the mentioned limitations are suggested. The PLS schemes are summarized in a tabular form for a side-by-side comparison.
- A complete security framework that targets all security services is presented. It is a combination of algorithms and protocols that are, both, secure and efficient.
- Two lightweight and secure device authentication protocols are presented. These protocols are based on multiple factors, mainly a PUF-derived parameter and channel randomness. For each new session, the employed factors are updated (changed) dynamically to minimize the risk of having an exposed key and prevent tracking. Moreover, lightweight cryptographic computations, mainly the XOR operation and a one-way hash function, are utilized, which makes the proposed techniques very efficient compared to other authentication schemes in the literature. In particular, both protocols take into account energy constraints, processing capabilities, and practical limitations of low-power devices. The presented simulation results show that the proposed protocols achieve the desired security and performance (efficiency) requirements, in comparison to existing authentication protocols. Additionally, using the mutual authentication step, users are able to establish a shared secret session key, which will later be used to generate a dynamic key. The resulting key will be used to ensure data confidentiality (encryption), message authentication and system availability.

- A dynamic key generation scheme is presented. Specifically, this key is obtained using channel-derived parameters and the secret session key that has been exchanged between users in the mutual authentication step.

- Two novel encryption/cipher schemes are proposed and assessed (data confidentiality), namely the 2-D permutation scheme and the enhanced phase encryption scheme. These schemes are modified versions of traditional schemes which are the permutation scheme and the phase encryption scheme.

- Four data confidentiality methods are derived and studied. The first scheme secures the OFDM system. The second is a generic scheme that applies to all post-modulation data frames in any system utilizing any multiple access method (not necessarily an OFDM system). The third is a NOMA-based data confidentiality scheme that takes into account the system and transmission models of the NOMA system and the fourth is a MIMO-based scheme. Various security and performance metrics are used to evaluate the proposed schemes. All of the proposed schemes utilize the dynamic key to derive cipher primitives used in the encryption process.

- The effect of encryption before and after the Inverse Fast Fourier Transform (IFFT) in OFDM is explored and analyzed. In particular, the existing and proposed cipher schemes are performed Pre-IFFT and Post-IFFT and several conclusions are drawn out.

- Two PLS source authentication and message integrity schemes are designed and tested. The first technique is a novel hashing technique for OFDM frame symbols (complex symbols), based on the random characteristics of the physical layer (dynamic key). This technique is done in the time and frequency domains, that is, before and after performing the IFFT transformation (two variants). Essentially, complex OFDM symbols are first pre-processed and then, key-hashed to ensure their integrity via a non-linear function (an integer function is used for the Pre-IFFT case and a non-integer function is used for the Post-IFFT case). The two cases (Pre-IFFT and Post-IFFT message authentication) are compared to each other, on one hand, and to popular methods in the literature, on the other. The second scheme is a generic hashing function that can be applied to any system utilizing any multiple access scheme (not necessarily OFDM). It is also based on the random and dynamic parameters of the physical layer (specifically the dynamic key), and it is applied on post-modulation symbols in their complex format. The cryptographic properties such as key and message sensitivities, as well as collision resistance are confirmed. Also, security and performance tests are presented to validate robustness and efficiency in comparison to existing hash functions.

- Finally, two schemes that jointly achieve secrecy and reliability, are explored. These schemes exploit the notions of Random Linear Network Coding (RLNC), multi-homing and PLS, to ensure the availability of data at all times (in case of link failure, error, or availability attacks). Specifically, the joint encryption/encoding processes, which consist of simple and lightweight operations, enable users to decode and recover original data using a subset of the received data. Both schemes are performed at the byte level, however, one scheme considers integer byte values and the other considers binary byte values. The robustness and efficiency of the proposed cryptographic solutions are analyzed using cryptanalysis, security and performance tests.

1.4 Organization of Dissertation

The rest of this dissertation is organized as follows:

Chapter 2 presents the necessary background information related to PLS, the different security services, OFDM, NOMA and MIMO. Chapter 3 presents a comprehensive literature review on the existing research work done in the field of PLS. It also compares the schemes presented in the literature, and identifies their limitations, to highlight on the need for designing novel PLS methods suitable for emerging communication systems. PLS mainly targets three technologies, which are OFDM, NOMA and MIMO. In each category, existing schemes are further divided into several sub-categories, each corresponding to different security services. Chapter 4 presents and evaluates two device authentication protocols. Chapter 5 presents the dynamic key generation scheme. Chapters 6, 7, 8 and 9 present and assess four data confidentiality schemes, each targeting a different system (OFDM, general, NOMA and MIMO). Chapter 10 presents two source authentication and message integrity schemes, and compares them to existing schemes in the literature. Chapter 11 presents the data availability schemes that jointly enhance secrecy and reliability based on RLNC, multi-homing and PLS. Finally, Chapter 12 summarizes and concludes this dissertation, and it presents the open research problems that need further investigation.

1.5 Publications

During the PhD residency, the following publications have been produced.

Journal Papers

1. **R. Melki**, H. Noura, and A. Chehab. "Design and Realization of an Efficient & Secure PLS Cipher Scheme For Multi-Homed Systems." (submitted).

2. **R. Melki**, H. Noura, and A. Chehab. "An Efficient and Secure Cipher Scheme for MIMO-OFDM Systems based on Physical Layer Security." (submitted).

3. **R. Melki**, H. Noura, and A. Chehab. "Physical Layer Security for NOMA: Limitations, Issues and Recommendations." (submitted).

4. **R. Melki**, H. Noura, J. Hernandez Fernandez, and A. Chehab. "Message Authentication Algorithm for OFDM Communication Systems." (submitted).

5. H. Noura, **R. Melki**, and A. Chehab. "Efficient & Secure Cipher Scheme For NOMA Systems." (submitted).

6. H. Noura, **R. Melki**, and A. Chehab. "Secure MIMO D2D Communication Based on a Lightweight and Robust PLS Cipher Scheme." (submitted).

7. **R. Melki**, H. Noura, and A. Chehab. "Efficient & Secure Multi-Homed Systems Based on Binary Random Linear Network Coding." Computers and Electrical Engineering, 2020.

8. H. Noura, **R. Melki**, M. Malli, and A. Chehab. "Lightweight & Secure Cipher Scheme for Multi-Homed Systems." Wireless Networks, 2020.

9. H. Noura, **R. Melki**, A. Chehab, and J. Hernandez Fernandez. "Efficient and Secure Message Authentication Code at the Physical Layer." Wireless Networks, 2020.

10. **R. Melki**, H. Noura, and A. Chehab. "Lightweight Multi-Factor Mutual Authentication Protocol for IoT Devices." International Journal of Information Security (2019): 1-16.

11. H. Noura, **R. Melki**, M. Malli, and A. Chehab. "Design and Realization of Efficient & Secure Multi-Homed Systems based on Random Linear Network Coding." Computer Networks 163 (2019): 106886.

12. **R. Melki**, H. Noura, M. Mansour, and A. Chehab. "Physical Layer Security Schemes for MIMO Systems: An Overview." Wireless Networks (2019): 1-23.

13. H. Noura, **R. Melki**, A. Chehab, and M. Mansour. "A Physical Encryption Scheme for Low-Power Wireless M2M Devices: A Dynamic Key Approach." Mobile Networks and Applications 24.2 (2019): 447-463.

14. **R. Melki**, H. Noura, M. Mansour, and A. Chehab. "An Efficient OFDM-based Encryption Scheme Using a Dynamic Key Approach." IEEE Internet of Things Journal 6.1 (2018): 361-378.

15. **R. Melki**, H. Noura, M. Mansour, and A. Chehab. "A Survey on OFDM Physical Layer Security." Physical Communication 32 (2019): 1-30.

International Conference Papers

1. **R. Melki**, H. Noura, and A. Chehab. "An Efficient and Secure Cipher Scheme for Filter Bank Multi-Carrier Systems." International Conference on Wireless Networks and Mobile Systems (WINSYS), 2020

2. **R. Melki**, H. Noura, and A. Chehab. "Lightweight and Secure D2D Authentication & Key Management Based on PLS." IEEE Vehicular Technology Conference (VTC2019-Fall). IEEE, 2019.

3. H. Noura, **R. Melki**, and A. Chehab. "Secure and Lightweight Mutual Multi-Factor Authentication for IoT Communication Systems." IEEE Vehicular Technology Conference (VTC2019-Fall). IEEE, 2019.

4. **R. Melki**, H. Noura, and A. Chehab. "Efficient & Secure Physical Layer Cipher Scheme for VLC Systems." IEEE Vehicular Technology Conference (VTC2019-Fall). IEEE, 2019.

5. **R. Melki**, A. Hussein, and A. Chehab. "Enhancing Multipath TCP Security Through Software Defined Networking." IEEE International Conference on Software Defined Systems (SDS). IEEE, 2019.

6. H. Noura, **R. Melki**, A. Chehab, M. Mansour, and S. Matrin. "Efficient and Secure Physical Encryption Scheme for Low-Power Wireless M2M Devices." IEEE International Wireless Communications & Mobile Computing Conference (IWCMC). IEEE, 2018.

7. **R. Melki**, M. Mansour, and A. Chehab. "A Fairness-based Congestion Control Algorithm for Multipath TCP." IEEE Wireless Communications and Networking Conference (WCNC). IEEE, 2018.

Chapter 2

Background

This chapter presents and discusses the necessary background information related to physical layer security, Orthogonal Frequency Division Multiplexing (OFDM), Non-Othogonal Multiple Access (NOMA) and Multiple-Input Multiple-output (MIMO) systems.

2.1 Preliminaries and Notation

Appendices A and B list all of the used abbreviations and notations in this dissertation.

2.2 Physical Layer as a Security Solution

In general, there are two approaches for designing security techniques: 1) computational security and 2) information-theoretic security [14]. In computational security, the security level of a specific technique is indicated by the amount of time it takes to break a code [15, 16]. Information-theoretic security, on the other hand, does not rely on computational power, but rather on the physical properties of the radio channel. It is based on Claude Shannon's early work on the mathematical theory of communication. Shannon's work mainly focused on symmetric-key encryption systems [17]. In contrast, the work of Aaron Wyner, in this field, is more relevant to PLS, in which he used the wiretap channel model to prove that secrecy can be attained using the communication channel itself, without the need of shared secret keys [18]. As a result, physical layer security techniques, based on information-theoretic security, have attracted a lot of attention, lately [14, 19, 20].

Physical layer security methods can be either 1) key-based [17, 21] or keyless based on Wyner's wiretap channel [18]. While, keyless security schemes require full or partial Channel State Information (CSI) of the eavesdropper's

channel (unpractical in most situations), key-based security exploits the randomness of common wireless channels to establish secure keys between two legitimate users [20]. In [18], Wyner analyzes the wiretap channel model and proves that a transmitter can send information, securely, if the Signal-to-Noise Ratio (SNR) of the legitimate receiver is higher than that of the eavesdropper (channel between transmitter and receiver is better than the channel between the transmitter and the adversary). In this case, the maximum achievable secrecy rate, in which data can be transmitted secretly from the legitimate transmitter to the legitimate receiver, is referred to as the secrecy capacity. The keyless model proposed in [18] can be realized when using PLS schemes such as wiretap coding [18, 22], beamforming [23], and power allocation [24]. However, in all of the previously mentioned techniques, it is assumed that the transmitter knows the CSI between the eavesdropper and itself, which is unrealistic [3]. To ensure robust security, PLS solutions should be independent of the CSI of the adversary (key-based).

The applications of PLS are numerous. It can be applied to all types of applications (Machine-to-Machine (M2M), Point-to-Point or Link-to-Link), especially those adopting wireless communications. M2M is a generic class of applications in which a variety of devices communicate with each other or through a network. More specifically, it refers to the direct communication of machines (objects) with each other [25]. Few state-of-the-art technologies that fall under "Machine-to-Machine (M2M) communications" are Device-to-Device (D2D), Internet-of-Things (IoT) and Vehicular Communication. Other wireless systems and technologies that can benefit from PLS include Visible Light Communication (VLC), Body Area Network (BAN), Power Line Communication (PLC), Radio Frequency Identification (RFID), Vehicular Ad-Hoc Network (VANET), smart grid, Ultra-Wide-Band (UWB), Unmanned Aerial Vehicle (UAV), mm-Wave, cognitive radio, index modulation, Multiple-Input Multiple-output (MIMO) systems, Orthogonal Frequency Division Multiplexing (OFDM) and new multiple accessing schemes such as Non-Orthogonal Multiple Access (NOMA) [26].

2.3 Security Services

In order to design and assess various security schemes, it is important to understand the different security services.

The main goal of information security is to protect information and mitigate its associated risks, by preventing unauthorized use, access, modification, disruption, and inspection of data. Generally, there are five main security elements [27]:

- Availability and Utility
- Integrity
- Authenticity

- Confidentiality
- Non-Repudiation

The first security element/service, which is availability, refers to the ability of users to access information at all times. This is important since whenever a system is not functioning properly or is not able to deliver data efficiently and in a timely manner, information availability is compromised. Typically, illegitimate users aim to interrupt the availability of data through the Denial-of-Service (DOS) attack and the Distributed Denial-of-Service (DDOS) attack. One conventional way to overcome these attacks is by using intrusion detection/prevention systems or Security Information Event Management [28].

Utility may not be considered as a main security element, however, it is of great benefit. It is somehow related to availability, in the sense that utility is mandatory for achieving proper availability. For example, if the encryption key, used to encrypt valuable information, is lost or accidentally deleted, the ciphered information will still be available, however, it will be useless. Therefore, in some cases utility, which refers to something being useful, is a must.

On the other hand, integrity is a major and basic security component, which refers to the correctness of received data (unaltered data). Specifically, users should be able to verify that the received data has not been changed or modified by untrusted/unauthorized entities, during transfer. Popular data integrity verification mechanisms include the checksum and cryptographic hash functions.

There are two types of authentication: device authentication and source authentication. The former is performed at the beginning of a communication session, while the latter is often associated with data/message integrity. Device authentication enables communicating users to confirm and verify each other's identities, based on multiple factors which are: you have, your are, you know. This step is crucial at the beginning of each communication session, where the transmitter should be able to ensure that the receiver is, indeed, a legitimate user, and vice versa (legitimacy of users). To achieve device (or user) authentication, several processes can be employed, few of which are: usernames and passwords, biometrics (retina or fingerprints), tokens, randomly generated numbers and digital certificates (using private keys). Differently, source authentication validates the origin/source and legitimacy of the message (correctness). It is usually associated with message integrity and it is realized using keyed-hashing algorithms (Message Authentication Code (MAC)).

Following the device authentication step, users should achieve data confidentiality and information secrecy/privacy. In particular, only authorized users should gain access to sensitive information, which should be well protected and private [29, 30, 31, 32]. Generally, data confidentiality is achieved using cryptography; symmetric encryption or asymmetric encryption. The former is used more in practice since it is more efficient, whereas the latter is is used for exchanging symmetric keys.

Finally, non-repudiation confirms that the transmitted message was, truly, issued by the intended transmitter and both, the transmitter and receiver, can not deny it. In other words, it assures that both, the transmitter and receiver, can not deny the validity of something. This security element can be attained using digital signatures and/or encryption, which proves the proper delivery and reception of data.

To guarantee the desired security level, cryptographic and non-cryptographic algorithms are, currently, employed. Most cryptographic techniques require multiple rounds, in which they apply the same round function multiple times. This, in turn, ensures two important security properties: 1) diffusion, which obscures the relationship between the utilized encryption key and the resulting ciphered data, and 2) confusion, which guarantees that any change in the input data affects the obtained output, significantly [30]. On the other hand, conventional multi-round schemes introduce a considerable amount of overhead in terms of latency and resources, which has motivated the search for alternative solutions, namely PLS. In the literature, several PLS schemes have been presented, each addressing one or more of the mentioned security services.

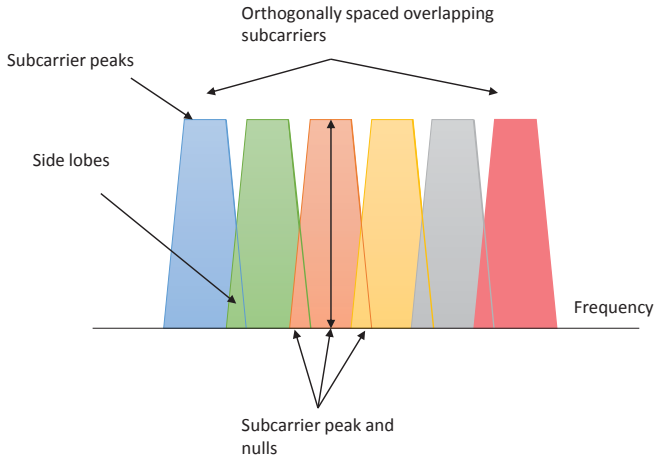


Figure 2.1: OFDM frequency spectra

2.4 OFDM and its Variants

Orthogonal Frequency Division Multiplexing (OFDM) was first introduced in the late 1950's [33, 34]. Since then, it has been widely adopted as the basic building block for many modulation schemes in different technologies such as 802.11 WLAN, 802.16 WiMAX, and 3GPP LTE. In principle, the frequency band is divided into multiple narrow sub-bands, each modulated with a conventional

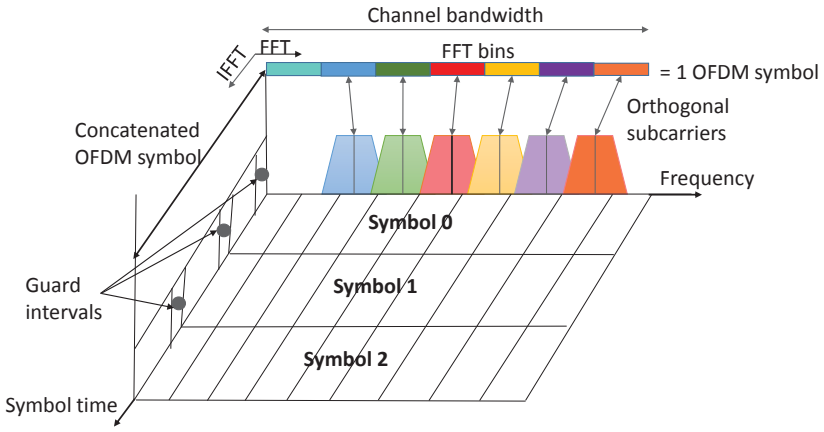


Figure 2.2: OFDM representation in the time-domain and frequency-domain

digital modulation scheme. The large number of closely-spaced overlapping sub-carriers, transmitting in parallel, leads to an increase in the spectral efficiency (Figure 2.1). This is one of OFDM's main advantages. Another advantage is overcoming the effect of frequency selective fading, which results from multi-path propagation [35]. This requires each sub-band to have a bandwidth (BW) satisfying [36]:

$$BW < \frac{1}{2\pi DS_{avg}}, \tag{2.1}$$

where DS_{avg} is the average delay spread. Therefore, the problem of having a frequency selective fading channel is simplified to having multiple flat fading sub-channels which can be easily mitigated by equalization.

Another issue in OFDM is Inter-Symbol Interference (ISI) and Inter-Carrier Interference (ICI). ISI is the interference caused by adjacent symbols due to the delay spread (DS), while ICI is interference caused by adjacent sub-carriers. To overcome ISI and ICI, a guard interval, also knwon as Cyclic Prefix (CP), is inserted between consequent OFDM symbols and its length is set to be more than the channel delay spread. The cyclic prefix is simply an extension of the signal itself appended at the beginning of the OFDM symbol [37].

Figure 2.2 illustrates the main concepts of an OFDM signal, which is represented in both time- and frequency-domains. Conceptually, a combination of Fast Fourier Transform (FFT) and Inverse Fast Fourier Transform (IFFT) digital signal processing is required for OFDM implementation. These transforms are important from the OFDM perspective because they can be viewed as mapping digitally modulated input data (data symbols) onto orthogonal sub-carriers [37, 38, 39]. Figures 2.3 and 2.4 show the detailed OFDM transmitter and receiver block diagrams, respectively.

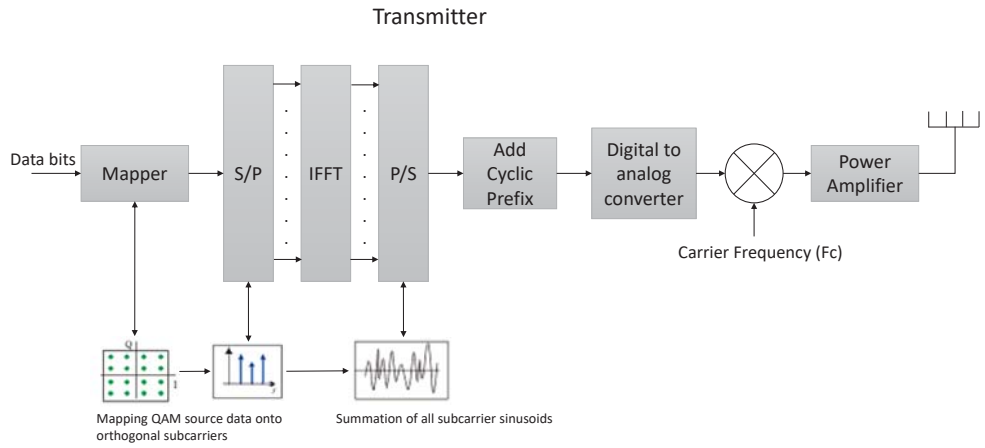


Figure 2.3: A detailed OFDM transmitter block diagram

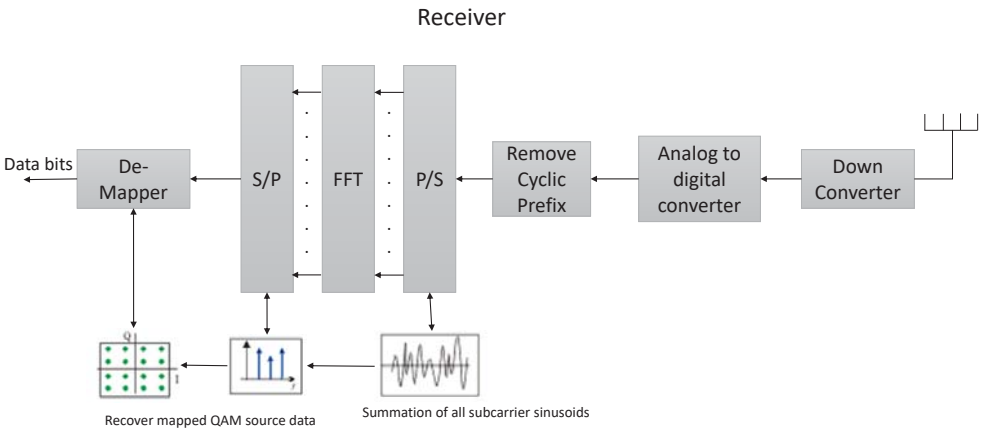


Figure 2.4: A detailed OFDM receiver block diagram

At the transmitter's side, the frequency-domain input bits are first grouped, and then mapped to complex data symbols, each representing a different modulation constellation point [38, 39]. After serial-to-parallel (S/P) conversion, the IFFT block modulates frequency-domain data symbols onto a specific number of orthogonal sub-carriers, which make up a single OFDM symbol. In particular, the IFFT block transforms frequency-domain data to time-domain data. The resulting time-domain OFDM symbols are concatenated after inserting the guard intervals for each symbol to create the final OFDM burst signal. The output of the IFFT is basically the sum of the orthogonal sinusoids representing the transmitted time-domain signal across the radio channel (Figure 2.3) [37, 38, 39].

At the receiver's side, a reversed operation of the previously mentioned technique is performed. Here, the time-domain signal is transformed to the frequency-domain using the FFT block, from which data bits are recovered (Figure 2.4).

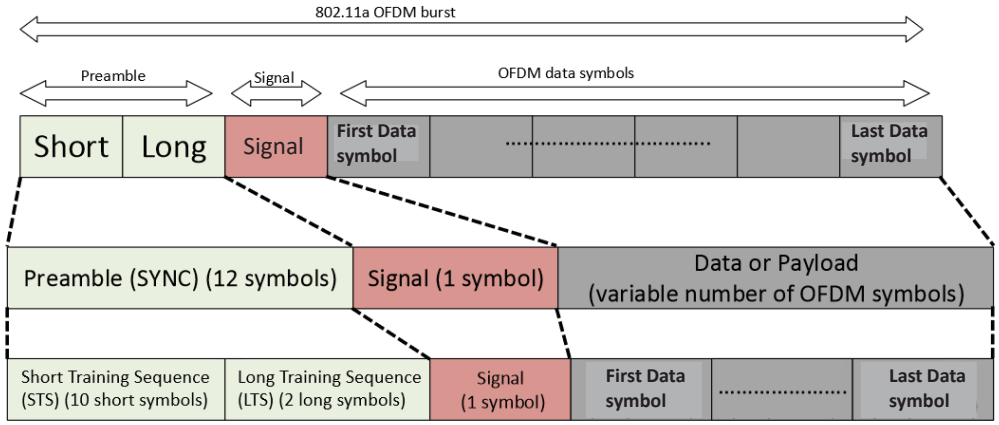


Figure 2.5: 802.11a frame structure

OFDM is adopted in the IEEE 802.11 a/g/n standards for signal modulation [40]. The corresponding structure of the physical layer packet of IEEE 802.11 OFDM is shown in Figure 2.5. The packet consists of a preamble, a SIGNAL field and a DATA (payload) field [10]. The DATA field includes the transmitted data bits. The SIGNAL field, which is equivalent to one OFDM symbol, carries information related to the coding rate, the mapping scheme and the length of the DATA field. The preamble, on the other hand, is 16 μs long and is divided into two fields: the Short Training Sequence (STS) field and the Long Training Sequence (LTS) field (Figure 2.5) [10, 41].

- **STS Field**: Consists of ten repetitions of STS, each having a duration of 0.8 μs. The STSs are mainly used for signal detection, coarse frequency offset estimation, Automatic Gain Control (AGC) diversity selection and time synchronization [37].

- **LTS Field**: Includes a guard interval (1.6 μs) and two repetitions of LTS (3.2 μs each). The LTSs are used for channel estimation (CSI), fine frequency offset estimation and channel equalization [37].

The CP in OFDM introduces redundancy in transmitted signals and degrades the overall performance in terms of data rate, spectral efficiency and power efficiency [42]. Additionally, OFDM systems suffer from two major drawbacks which are: i) high Peak-to-Average Power Ratio (PAPR) and ii) high Out-of-Band (OOB) emissions. Basically, all Multi-Carrier Modulation (MCM) waveforms experience high PAPR, however, frequency confinement varies, significantly, from one MCM waveform to another. OFDM uses a rectangular pulse shape which results in poor confinement in frequency-domain leading to high OOB emission [43].

This has lead to the emergence of other MCM methods, mainly the Filter Bank Multi-Carrier transmission scheme (FBMC), the Universal Filter OFDM (UF-OFDM) and the Generalized Frequency Division Multiplexing (GFDM), as promising candidates for the future 5G mobile communication system. These methods are briefly discussed below:

- **FBMC:** This technology eliminates the CP and introduces filter banks to the OFDM system. FBMC is one of the key technologies of future networks; mainly 5G networks. It has been designed to overcome the drawbacks of OFDM systems and to enhance the system's performance, efficiency and flexibility. More Specifically, instead of using a CP, FBMC uses an array of filters equal to the number of sub-carriers (sub-carrier level) and OQAM (Offset Quadrature Amplitude Modulation) modulation to reduce the OOB power leakage and increase the spectral efficiency with low costs [44]. The OQAM pre-processing block is based on a two-step operation. The first step is converting complex data into real data by separating the real and imaginary components of a complex-valued symbol into two symbols. This increases the sample rate by a factor of 2 [45]. Afterwards, the two symbols are multiplied by a specific sequence. Accordingly, this technique avoids the interference between consecutive sub-channels since in each time interval, either the real or the imaginary part of the original symbol is transmitted on a sub-carrier. At the transmitter side, the symbols are first modulated using offset QAM, and then, filtered using a Synthesis Filter Bank (SFB), which includes the IFFT block and the poly-phase network. Similarly, at the receiver, a reversed operation is performed, in which time-domain symbols are recovered using an Analysis Filter Bank (AFB) (which includes an FFT block and PPN) and then demodulated (OQAM post-processing). The SFB and AFB consist of an array of filters equal in number to available sub-carriers.There are two types of FBMC implementations: frequency spreading (FS-FBMC) and poly-phase network (PPN-FBMC). The latter is more common in the literature, since it reduces the high complexity that results from extra filtering [46, 47].

- **UF-OFDM:** UF-OFDM groups sub-carriers to sub-bands (sub-groups) and then applies filtering to each sub-group, separately. Hence, UF-OFDM can be seen as a compromise between OFDM and FBMC since it requires less overhead and low complexity compared to FBMC [48].

- **GFDM:** GFDM also uses the filter bank multi-carrier concept. Basically, GFDM spreads the available spectrum for each user into multiple spectral segments, each having more or less bandwidth [48].

It should be noted that the majority of the work in the literature targets the security of OFDM systems and very minimal work tackles the security of

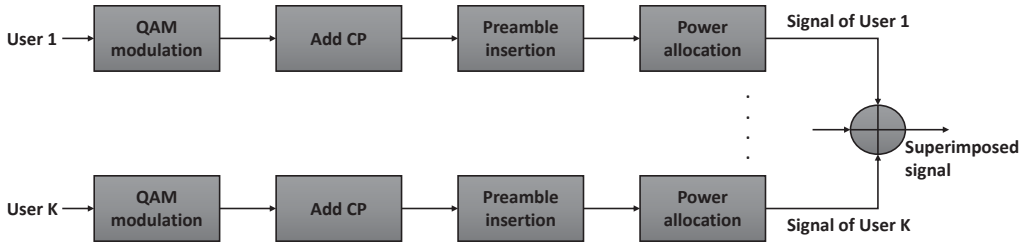


Figure 2.6: The NOMA transmitter block diagram

other variant systems. However, any OFDM PLS technique can be adapted and integrated into any OFDM-variant.

2.5 NOMA Technology

In the literature, NOMA schemes are divided into two main classes: Power-Domain NOMA (PD-NOMA) and Code-Domain NOMA (CD-NOMA). In the former case, different users, sharing the same time-frequency resources, are assigned different power coefficients based on their channel conditions and distance from the Base Station (BS). Following the allocation of power coefficients, the resulting signals are superimposed and transmitted. At the receiver, Successive Interference Cancellation (SIC) is used to decode the signals, one by one, until the desired signal is obtained [49]. The block diagram of the NOMA transmitter is shown in Fig. 2.6.

In CD-NOMA, different signals are also multiplexed over the same time-frequency resources, however, using unique codes. Specifically, each user is allocated a unique non-orthogonal code with low cross-correlation, or a sparse sequence. Examples of CD-NOMA include Multi-User Shared Access (MUSA), Sparse Code Multiple Access (SCMA), and Low-Density Spreading (LDS). The concept of CD-NOMA is similar to that of Code Division Multiple Access (CDMA), except that CDMA utilizes orthogonal codes. In addition to PD-NOMA and CD-NOMA, there are other, less popular, NOMA schemes such as Pattern Division Multiple Access (PDMA) and Bit Division Multiplexing (BDM). Generally, more attention is being drawn to PD-NOMA than CD-NOMA, due to its simplicity, efficiency, and applicability in current systems. Moreover, it does not require additional bandwidth nor major changes, to improve spectral efficiency.

Also, from a PLS viewpoint, PD-NOMA has many challenges and limitations that should be addressed and highlighted. The security vulnerabilities of PD-NOMA are attributed to multiple factors which are [50]:

- The superimposed messages of multiple users are sent at the same time, over the same bandwidth. Hence, these information, which are sent in the

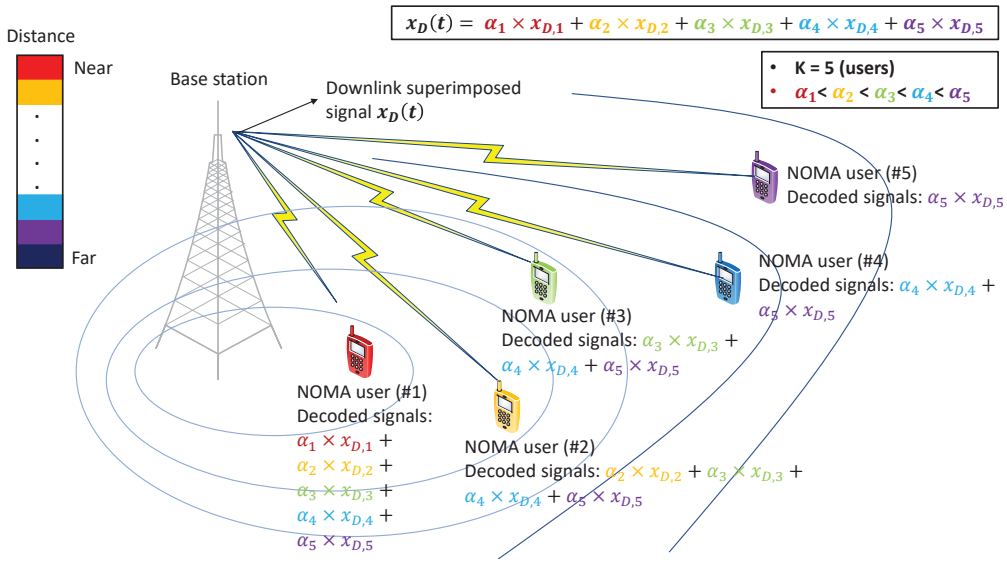


Figure 2.7: NOMA downlink system with five NOMA users

clear (broadcast nature of wireless transmission), are vulnerable to being captured and leaked to illegitimate users.

- To recover the signal of each legitimate user, Successive Interference Cancellation (SIC) is applied. This process allows users to decode all of the transmitted signals (superimposed) to obtain the desired one. Hence, superimposed signals are not secure and are exposed to both, legitimate and illegitimate users.

2.5.1 Downlink PD-NOMA System Model

At the transmitter's side, the Base Station (BS) superimposes all individual information signals into a single waveform, using different power coefficients (Fig. 2.7). In particular, the User Equipment (UE) that is farthest (FU) from the BS is allocated maximum power, while the nearest UE (NU) is allocated minimum power. Power allocation is also related to the quality of the channel and its conditions. All users in the network receive the same signal that contains the information of all UEs. For the recovery of individual signals, each UE performs SIC to decode the strongest signal, and then subtract it from the received signal. This operation is iterated successively until the UE finds its signal. In contrast, the Far User (FU), which has the highest power coefficient, can recover its desired signal directly without performing SIC, since other signals are treated as noise [51]. The transmitted downlink signal ($x_D(t)$) can be written as:

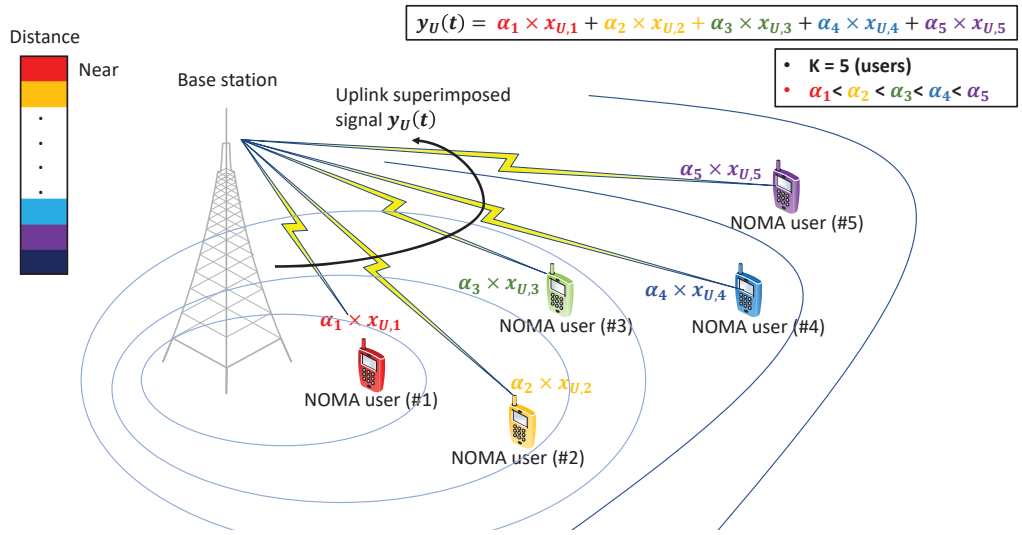


Figure 2.8: NOMA uplink system with five NOMA users

$$x_D(t) = \sum_{k=1}^{K} \sqrt{\alpha_k P_{BS}} x_{D,k}(t), \tag{2.2}$$

where $x_{D,k}(t)$ is the individual downlink information of the k^{th} UE, P_{BS} is the transmission power at the BS, K is the number of users in the network, and α_k is the power allocation coefficient for user k. The received signal $(x'_{D,k}(t))$ at k^{th} user is expressed as follows:

$$x'_{D,k}(t) = h_k x_D(t) + no_k(t) = h_k \sum_{k=1}^{K} \sqrt{\alpha_k P_{BS}} x_{D,k}(t) + no_k(t), \tag{2.3}$$

where h_k is the channel coefficient of the k^{th} user, and $no_k(t)$ is the Additive White Gaussian Noise (AWGN) at the k^{th} UE.

2.5.2 Uplink PD-NOMA System Model

In the uplink PD-NOMA network, each user transmits its signal to the BS. These signals are multiplexed into one signal (Fig. 2.8). At the BS, SIC is applied to detect and distinguish the signal of each user [51]. More specifically, the received signal at the BS, which includes the signals of all users, is represented as follows:

$$y_U(t) = \sum_{k=1}^{K} x_{U,k}(t) h_k + no_k(t). \tag{2.4}$$

Here, UEs optimize their transmit powers according to their locations, as in the downlink NOMA case.

For the CD-NOMA case, the same concept is applied, however, instead of using unique power coefficients, users utilize unique non-orthogonal codes. Sparse Code Multiple Access (SCMA) is the most popular CD-NOMA technique. Using this technique, encoded bits get mapped to complex codewords, directly. Particularly, the SCMA encoder combines two operations in one step: modulation and IFFT/FFT (Inverse Fast Fourier Transform and Fast Fourier Transform) [52].

The above-discussed system model is suited for Single-Input Single-Output (SISO) systems, where channels are represented by scalars. Similarly, NOMA can be applied with Multiple-Input Multiple-Output (MIMO) systems. Unlike SISO-NOMA, the channels of MIMO-NOMA systems are represented by matrices. Currently, there are two main designs for MIMO-NOMA, which are:

1. Beamformer-based MIMO-NOMA: In this design, different beams are directed towards different users and SIC is performed by users sharing the same resource block.

2. Cluster-based MIMO-NOMA: Cluster-based MIMO-NOMA divides users into several clusters, and each cluster is served by a single beam. Afterwards, SIC is used among users within the same cluster to recover the desired signals [50].

2.6 Massive MIMO Technology

Multiple-Input Multiple-Output (MIMO) is a radio communications technology that promises increased link capacity and spectral efficiency along with improved link reliability. Moreover, it is being adopted in various technologies such as Wi-Fi, LTE, 5G and many others. Typically, a signal travels along many paths (previously viewed as interference) due to reflection, refraction and scattering. The MIMO system exploits this phenomenon to carry data on different paths, thus, achieving diversity gains [53]. MIMO can be used with many air interfaces such as Time-Division Multiple Access (TDMA) and Code-Division Multiple Access (CDMA), however, MIMO-OFDM is the most popular and dominant air interface for 4G and 5G communication systems.

Conceptually, diversity provides the receiver with multiple copies of the same message in order to improve the robustness and reliability of radio-communications, and reduce the error rate (increased diversity gains) [53]. The different diversity modes are:

- **Time diversity**: The same message is transmitted at different times (same copy at different time slots).

- **Frequency diversity**: The same message is transmitted over different frequencies (different sub-carriers in the case of OFDM systems).

- **Spatial diversity**: The same message is transmitted over different antennas. This will result in redundant data on different paths. At the receiver's side, the various versions of the received data are combined to enhance the reliability of data-transfer. Several schemes are used to allow the transmission of several copies of the same message across a number of antennas such as space time block coding and Alamouti coding. This mode is further divided into: receive diversity, in which more antennas are used at the receiver's side than on the transmitter's side, and transmit diversity, in which more antennas are used at the transmitter's side than on the receiver's side.

The diversity gain is equal to the product of the number of antennas at the transmitter and the number of antennas at the receiver.

On the other hand, in order to improve the data rate rather than the system's robustness, MIMO spatial multiplexing is considered. Here, data is divided and transmitted across separate antennas as independent streams (different data on different antennas), thus, increasing the degrees of freedom or multiplexing gain, which is equal to the minimum number of antennas (either the number of antennas at the transmitter or the number of antennas at the receiver).

Another important aspect in MIMO is beamforming in which signal strength is maximized along a specific direction only [53].

2.6.1 MIMO System Model

In general, a narrow-band flat fading MIMO system is modelled as:

$$y = H \cdot x + no, \tag{2.5}$$

where y and x are the receive and transmit vectors, respectively. H and no are the channel matrix and noise vectors, respectively.

Moreover, the channel matrix, H, can be decomposed into three sub-matrices using Singular Value Decomposition (SVD) as shown below:

$$H = U \cdot \Lambda \cdot V^H, \tag{2.6}$$

where U and V are orthogonal matrices such that $U \cdot U^H = I$ and $V \cdot V^H = I$. The operation $(\cdot)^H$ represents the Hermitian operation and I is an identity matrix. Also, Λ is a diagonal matrix satisfying the above equation [54].

Chapter 3

Literature Review

The main purpose of the literature review in this chapter is to survey and assess existing research work related to PLS. The majority of the PLS schemes in the literature fall under three principal categories, which are: PLS for OFDM, PLS for NOMA and PLS for MIMO.

3.1 PLS Schemes for OFDM

The PLS techniques, which target the OFDM system, can be further divided according to the previously listed security services, mainly, device authentication, key generation and distribution, data confidentiality, source authentication and message integrity and data availability.

3.1.1 Device Authentication

Typically, in traditional key-based authentication, the transmitter either sends a random number as a challenge and the receiver sends back the hash of both, a shared secret key and the challenge, or the transmitter encrypts a random number (or nonce) using a secret key and the receiver sends back the random number, incremented and encrypted using the same secret key (or a function of the nonce, encrypted using the secret key).

Recently, several physical layer authenticating schemes have been proposed for authenticating users at the physical layer. These schemes can be classified as keyless or key-based authentication schemes. The former exploits specific features of the legitimate devices or specific features of the shared channel between the users. However, this technique is considered unpractical and weak since there should exist some level of trust between any two users in order to identify and share these features. In other words, both communicating entities should be able to 1) confirm that the exchanged device and channel features are, indeed, legitimate (device not subject to impersonation) and 2) prove that the exchanged

authentication messages have not been manipulated or forged. Additionally, PLS authentication mechanisms can not only rely on channel characteristics, since these characteristics and parameters can be acquired by illegitimate users, which compromises the authentication process and makes it vulnerable to malicious attacks. For example, an illegitimate user is able to extract the CSI of a certain user if he is able to synchronize to the LTS in its corresponding OFDM packet preamble. Hence, the keyless approach is not considered secure and robust in real environments. As such, to strengthen the authentication mechanism and achieve the required security level, a secret key or parameter should be introduced along with the channel-based parameters. This is more practical and closer to the traditional challenge-response mechanism [13].

In [13], the authors propose an enhanced version of the scheme presented in [55], in which Tikhonov-distributed artificial noise is added to interfere with the phase-modulated key, used in the authentication process. Similarly, the authors of [56] integrate multipath delay characteristics to the channel impulse response for authentication.

The research presented in [57] experimentally investigates Carrier Frequency Offset (CFO) monitoring as an authentication method. Assuming two users are communicating, user authentication is verified if the difference between the CFOs at the transmitter is equal to that at the receiver.

The time bounded anti-spoofing technique is presented in [58] to enhance Wi-Fi authentication. Specifically, this technique leverages the CSI of the shared channel for the purpose of mutual authentication. Moreover, this technique is based on the facts that 1) different transmitting locations will likely result in different wireless channels, and 2) the channel state drift within a short time interval should be bounded. The proposed authentication method works as follows: the receiver continuously estimates the CSI upon the reception of a new packet. If, within a short period of time, the difference between the channel state of two consecutive packets, having the same address, is large, then the receiver concludes that one of these packets is spoofed and, hence, drops them. This is similar to the authentication mechanism presented in [59].

In [60], authors explore the potential possibility of using physical-layer channel responses as authenticators between each communicating pair. The proposed authentication scheme: physical layer assisted authentication for VANETs (Vehicular Ad hoc Networks), exploits the advantage of having unique physical layer channel responses for each communicating pair, so that the receiver would be able to identify the transmitter. More specifically, the sender appends the channel response estimated at its side, which is referred to as the authenticator signal, with the transmitted data. Consequently, the receiver authenticates the sender by comparing the authenticator signal with the estimated channel response at its side. If the two are close, then the sender is authenticated, otherwise the message is ignored. The same concept is applied in [61]. Similarly, in [62], the receiver compares the channel matrices of two consecutive frames; if the differ-

ence is small, then the sender is authenticated; if the difference is larger than a predefined threshold, then the communication is terminated.

Differently, the authors in [63] present a distributed authentication model in which several receiving nodes and a third party authority are involved. Whenever a receiving node estimates the channel response, it relays the information to the third party authority, which is responsible for the decision making process (authentication of users).

The authors of [64] proposed *PriLA*, PRIvacy-preserving Location Authentication systems in OFDM-based Wi-Fi networks. The protocol works as follows: First, the mobile user and the location-based service provider exchange handshake frames and extract both the CSI and the CFO information, which will be used to generate the secret key for the encryption of the following frames. Upon receiving the encrypted frames, the location-based service provider performs decryption and extracts the user's media access control address and location information. Then, the location-based service provider uses the CSI obtained from the received frames to construct a multipath profile, which is compared to the already stored profiles. Accordingly, the location-based service provider authenticates the sender and delivers the service.

Authors in [65], on the other hand, consider a two-hop wireless network that involves a relay and present two physical layer challenge-response authentication mechanisms. These mechanisms require a random number, channel reciprocity and a pre-shared secret key. The first mechanism, which assumes that the relay is trustworthy, works as follows: the sender first generates a random number and sends it to the relay, which forwards the signal to the receiver. The receiver calculates the inverse of the received signal, multiplies the result with a shared key and sends the obtained signal to the relay. Afterwards, the relay forwards the received signal back to the sender. Having the random number and shared key, the sender can authenticate the receiver. An obvious weakness of this protocol is that the relay, or even an eavesdropper, are able to obtain the shared secret key by simply multiplying the first signal with the second one. The second mechanism assumes that the relay is not trustworthy, and thus, it is more complicated and can be summarized as follows: the sender generates a random number and sends it along with the first shared key in two different OFDM symbols. The relay forwards the signal to the receiver, which divides the two signals and extracts the random number, having the first secret key. Afterwards, the receiver sends the extracted random number and another shared key in two different OFDM symbols to the relay, which forwards them to the sender. The same operation is done at the sender's side to extract the random number and thus verify the receiver. This technique requires not one, but two secret keys between users. Moreover, messages are sent in plaintext, which will allow the eavesdropper to extract the random number and the secret keys.

The authors of [66] use the concept of fingerprint embedding for message and user authentication. Here, it is also assumed that there exists a secret key between

legitimate users. First, the sender generates a tag from the secret key and the data, and superimposes the resulting tag onto the modulated message. At the receiver's side, the data is, first, estimated and then, encrypted using the secret key. The resulting chipertext is compared to the sent tag. If there is match, then the sender is authenticated.

The technique presented in [67] is similar to the one discussed previously expect that the tag is generated from the message and a secret, via a keyed-hash (MAC) operation. After generating the tag, the sender appends the tag to the message, and sends it to the receiver which, in turn, extracts the message, performs hashing using the secret and compares the generated hash to the one generated by the sender. Similarly, the techniques in [68] and [69] use the concept of tags.

The concept of hashing is also used in [70]. The receiver sends a random signal to the transmitter, who estimates the channel and generates a hash using the response of the multipath channel and the secret key. The resulting hash is then sent to the receiver, who estimates the channel and uses the extracted information to generate a hash in a similar manner to the transmitter. The generated and received hash digests are compared.

A pilot authentication scheme for a two-user multi-antenna OFDM system, is presented in [71]. It is based on the "Code-Frequency Block Group" coding mechanism, in which sub-carrier blocks are 1) encoded to authenticate pilots and 2) reused for channel estimation.

In contrast, authors in [72] present a secure PUF-based device authentication protocol for wearable devices, **independent of PLS**. The presented scheme allows wearable devices and mobile terminals, worn or carried by the same user, to mutually authenticate each other and share a secret session key, which will later be used to secure communication. Lightweight cryptographic computations, mainly the hash function and XOR operation, are utilized towards achieving high security and low complexity, simultaneously. However, this scheme is considered inefficient since it requires a large number of computational operations and a large execution time. In fact, 17 hash functions are needed to ensure secure authentication between the two devices, which is quite exhaustive for resource- and power-limited devices. A different PUF-based approach is applied in [73], where a three-factor anonymity authentication scheme is presented for Wireless Sensor Networks (WSNs) in Internet-of-Things (IoT) systems. This scheme mainly depends on two simple operations which are multiplication and hashing, however, it suffers from high computation costs (21 hash functions). Similar, but less efficient user authentication schemes, are also presented in [74] and [75]. The presented protocols require the exchange of four messages and a total of 31 and 19 hash functions, respectively. All of the aforementioned PUF-based authentication schemes have been proven to be secure in the literature, however, these schemes suffer from high computational complexity and communication costs.

Figure 3.1 and Table 3.1 summarize the techniques used in PLS for device

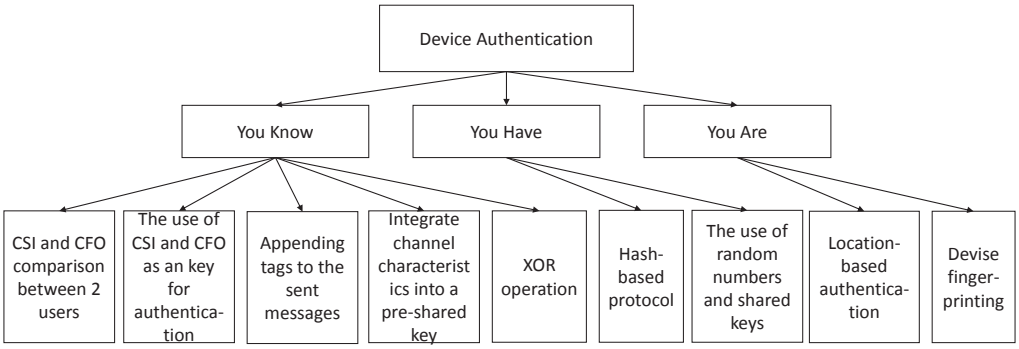


Figure 3.1: The proposed classification of the PLS device authentication schemes in the literature

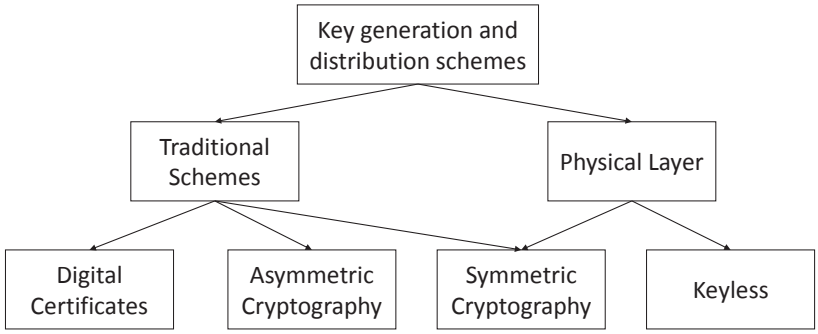


Figure 3.2: General classification of key generation and distribution techniques

authentication. Table 3.1 also presents the advantages, limitations, complexity and cost of each of the presented schemes.

3.1.2 Key Generation and Distribution

Key generation and distribution schemes are divided into: 1) keyless security and 2) secret key-based security [76] (Figure 3.2). Although keyless security requires no shared keys between communicating entities, legitimate users should have partial/full knowledge of the eavesdropper's CSI, which is considered very complex in terms of implementation, and rather impractical. Secret key-based security, on the other hand, is further discussed throughout this section due to its inevitable importance to PLS and due to the many research advances in this field.

Note: In theory, channel reciprocity is the same frequency spectrum that the uplink and downlink share. It is also when the coherence time of the channel is greater than the OFDM packet period, which is the case of a channel with low doppler spread [37].

Table 3.1: A summary of the PLS device authentication schemes for OFDM systems

Device authentication schemes	**Comparison of the channel properties of two consecutive frames (CSI,CFO)**	**Third party authority (relay) using XOR and simple multiplication operations**	**Using secret keys derived from the channel for encryption**	**Concept of "tags" which can be generated using encryption or hashing**
Advantage	No additional cost and overhead	Non-repudiation	Utilizing the notion of a secret key	Utilizing the notion of a secret key
Limitation	Ineffective when attacker is near legitimate user, that is when both experience same channel conditions	The third party authority is vulnerable to being impersonated. Moreover a high level of trust should exist between legitimate users and the third party	A secret key derived from the channel is a weak proposition since it can be easily generated and acquired	Computationally complex. It requires additional operations at the transmitter and receiver
Resource and communication cost	No additional cost and overhead	Authentication is verified through multiple rounds of communications (3 rounds) and performing simple operations such as XOR	Two steps are done prior to data transmission: 1) Key extraction. 2) Encryption and decryption	1) Key extraction and 2) "tag" generation. 3) Append "tags" to the transmitted messages
Complexity	Not computationally complex: performing comparison	Not computationally complex: performing XOR operations	Computationally complex: encryption	Computationally complex: encryption

In network security, major emphasis is targeted towards key generation and distribution (public, private and session keys) [77] due to its importance in many applications such as IoT, D2D and military communications. However, with the emergence of PLS, active research has shifted towards finding alternative techniques to public key cryptography where two users are able to exploit the un-predictable characteristics (randomness) of their shared wireless channel to generate a symmetric session key. This technique is low cost and simple, and exploits the most important feature in the physical layer which is randomness, as the building block of the generated encryption/authentication key [78]. Theoretically, a less random key will compromise the security of the system due to the small search space when conducting brute force attacks. Several techniques related to this issue are introduced and further discussed, next.

Assumption: In the following techniques, legitimate users are considered to be $\frac{\lambda}{2}$ *away from each other so that channel reciprocity is ensured, and the eavesdropper is considered to be at a distance* $> \frac{\lambda}{2}$.

A straightforward method for generating shared session keys is to extract them from the CSI directly. Such a scheme is presented in [79] where legitimate users exchange several dummy data packets, and for each received packet, the CSI is extracted and stored in a matrix. The columns of the matrix correspond to the sub-carrier index on which the packet was sent and the rows correspond to the packet number. Afterwards, the CSIs of each packet in a single column are checked; if the CSIs are in ascending order, one bit in the secret key is set to one otherwise, it is set to zero. In this manner, the secret key between two users is constructed without exchanging any information publicly.

Similarly, the approach presented in [80] includes the following four phases: channel estimation, public discussion, secret key extraction and verification. The key is directly generated from the channel frequency response of the legitimate users. This technique is simple but not fully secure since the generated key solely depends on the channel between the two parties, which is accessible by others (attackers and eavesdroppers).

The authors of [78] also use the same concept above, which is exploiting channel randomness and extracting the CSI. Here, the keys are extracted from the channel responses of individual OFDM sub-carriers over time. The paper also provides a thorough theoretical modelling of the system and channel, which will lead to the optimal probing rate and maximal key generation rate.

Another simple technique is introduced in [81], where two legitimate users exchange data to estimate the channel between them. Then, a transmitter pre-equalizes the sent message that contains the secret key. This technique decreases the probability of signal interception at the eavesdropper, and it allows perfect decoding at the receiver's side. The presented technique is simple and requires no further action at the receiver's side, however, in case the eavesdropper is able to retrieve the secret information from the shared channel between users, he will be able to recover the real data.

In [82], the authors use a new mechanism for generating encryption keys. The keys are obtained from the bipolar real OFDM samples at the output of the optical OFDM systems. The mechanism can be summarized as follows: 1) the transmitter multiplies the OFDM symbol, from which the initial session key will be derived, with a key only known to the transmitter. 2) The output is sent to the receiver, which multiplies the received signal with its own key and re-sends the result to the transmitter. 3) Again, the transmitter multiplies the obtained message with its own key. 4) Afterwards, the output sent to the receiver, which will in turn multiply with its key to recover the initial OFDM symbol and then extract the initial session key. The multiplication procedure mentioned in this technique refers to the element-wise multiplication operation. The initial session key is only used for the encryption of the first bipolar OFDM signal. The following encrypted signals are encrypted with another subsequent keys, which are obtained from the cyclic prefix of previous signals. The cyclic prefix is a copy of the last samples of the data included in the payload. The presented mechanism suffers from several major weaknesses which are: first, from step 3) the transmitter is able to acquire the secret key of the receiver, using its own key and the initial OFDM symbol. Second, from the exchanged messages of steps 2) and 3), the secret key of the receiver can be obtained by the eavesdropper. Third, from the exchanged messages of steps 2) and 4) the secret key of the transmitter can be obtained by the eavesdropper. Finally, using all exchanged messages the initial OFDM symbol which is used to generate the initial session key can be recovered.

Differently, authors in [83] exploit the inherent randomness that exists within an integrated circuit (such as an FPGA (Field-Programmable Gate Array) or RFID chip) [84, 85] to implement a PLS scheme based on Physical Unclonable Functions (PUFs). Optical scattering-based PUF devices are primarily used for creating identification and authentication keys [86, 87]. The mechanism aims to generate a secret session key used for encryption and it behaves as follows: the transmitter and receiver connect their devices and each one generates an equal number of optical scattering-based communication PUF using input spatial light modulator pattern which illuminates a volumetric scattering medium with a random coherent optical wavefront. Each combination of key-mixtures (using XOR operation) is saved in a digital electronic dictionary corresponding to all spatial light modulator patterns. The dictionary is assumed to be public and available to all local devices. Afterwards, whenever a secure message is sent to the receiver, the transmitter picks a key from the cluster which contains all available pre-generated keys and XORs this key with the message. The receiver receives the encrypted message and generates both the key-mixture and its key to recover the original message. The key-mixture will eliminate the two keys, thus, obtaining the message. In this technique, the transmitter sends the XORed message and the corresponding pattern, in order to help the receiver recover the message. However, with the key-mixtures and the patterns publicly available,

any eavesdropper is able to break this system using the chosen ciphertext attack.

In contrast, in [88], encryption keys are obtained from the bipolar real OFDM samples (the cyclic prefix) at the output of the optical OFDM systems. This technique is considered weak since encryption keys should never depend on the transmitted payload due to fading and channel noise.

In [89], the authors present a novel technique, iJam, that uses cooperative jamming for key distribution among users. The intuition behind this technique is that there is no need for any pre-shared information between the transmitter and receiver. The iJam technique works as follows: the transmitter sends two copies of each OFDM symbol back-to-back. The receiver, who is also the jammer in this case, randomly jams complimentary samples in the original signal and its repetition. Upon reception, the receiver picks out the correct samples from the signal and its repetition and re-arranges them to get the intended signal since only the jammer knows which samples are clean and which ones are jammed. The eavesdropper, on the other hand, can't differentiate between the clean and jammed samples, thus, he is not able to detect data, correctly. In this technique two OFDM symbols are sent back-to-back, which is inefficient (low data rate). In principle, each symbol should be acknowledged, separately, and there should exist a time slot between each pair.

A general overview of the PLS for the Internet of Things (IoT) technology is presented in [90]. More specifically, the authors highlight on the secret key generation issue and present one technique that enables two entities to exchange the secret session key, securely. The procedure is as follows: a transmitter sends a public pilot signal to the receiver, which will enable the latter to estimate the CSI. After a certain time, the receiver sends a pilot, which enables the transmitter to estimate the corresponding CSI. These steps are repeated several times, until both users get enough measurements to generate a set of keys. Since the measurements of both users might not be equal, users can exchange and compare the time stamps of the measurements. These time stamps might not be equal, however, their difference should be equal to the sampling delay in time-division duplex mode. In turn, both users will exchange their time stamps and each will keep the common ones only. Now both have the paired measurements and thus, are able to extract the secret key used for encryption.

Non-reciprocity is also addressed in [91]. A novel channel gain complement algorithm is presented. This scheme can mitigate the CSI disparity between a pair of wireless devices by removing the non-reciprocity component, which is obtained from a small number of probe packets. The presented procedure is as follows: After collecting a certain number of CSI samples, each of the two users send the extracted CSI samples, along with the corresponding time stamps to the other user. Then, the time stamps of both users are compared. Only the samples with time stamps on both sides satisfying the following requirement are utilized for non-reciprocity learning: the difference between the time stamps should be less than the threshold of the coherence time. Here, it should be noted that time

stamps can also be retrieved by an eavesdropper making the presented technique not completely secure against attackers.

In [92], a robust key generation technique is presented. This technique is mainly divided into two steps. In the first step, users estimate their channel gains which are considered primary random processes and compare them to a preset threshold. If the channel gain exceeds the threshold, the location is stored in a vector (initially all zeros). From the primary random process, a secondary random process is derived which is, in turn, used to generate the secret keys. After setting the vector, the moving increments, which are the difference between each two adjacent locations, serve as the realizations of the second random process. Finally, both users generate the secret key from the secondary random process. Again, in this technique, key generation mainly depends on the channel between users and, thus, suffers from the previously mentioned drawbacks (channel information can be acquired by adversaries).

In [93], the authors use a different approach to establish session keys between legitimate users by relying on keyless cryptography. The presented protocol consists of:

- **Initialization**: A public trusted authority generates the training sequence.
- **Training**: Legitimate users move into proximity (move their devices close to each other) and exchange the training sequences, which in principle will have different shifts in amplitude, phase, and frequency. After receiving the training sequences, the corresponding mismatch is evaluated.
- **Signal Transmission**: Users exchange several rounds of random analog signals to mask the mismatch in such a way that if the first signal in a specific round belongs to the first user, the corresponding bit is set to "1", otherwise it is set to "0". At each round, a secret bit of the key is obtained.
- **Key Establishment**: In this step, the secret key is obtained.

In [76], the authors introduce three metrics that are essential for the evaluation of key generation systems: 1) randomness: it is the most important feature in key generation and it is tested using the randomness tests provided by the National Institute of Standards and Technology (NIST), 2) Key Generation Rate (KGR): it is the amount of secret bits generated in one second, and 3) Key Disagreement Rate (KDR): it is the percentage of different bits in the generated keys of two communicating users. The key generation procedure is discussed in the survey [94].

The objective of the survey presented in [94] is to discuss the different aspects and fundamentals of secret key generation in PLS. First, the authors discuss the common sources of randomness, which are listed below:

- **Channel estimates**: The most popular randomness metrics used in PLS, are the channel gain and the channel phase, which fall under this category. These metrics can be easily estimated and they result in a high key generation rate. However, one drawback of using the channel gain or the channel phase in key generation, is the AWGN, which affects the reciprocity of the channel between two users.

- **Received Signal Strength (RSS) indicator**: The RSS is the received signal's power. This common metric can be implemented easily; however, in order to generate a key with acceptable entropy/randomness, a highly mobile scenario should be considered (not a practical assumption).

- **Distance**: Similar to the RSS indicator, this metric is best suited for mobile scenarios. However, a major weakness is that a key generated based on distance is vulnerable to being recovered if an eavesdropper is equipped with Angle of arrival (AoA) estimation capabilities.

- **Angle of Arrival (AoA)**: One advantage of using AoA as a common source of randomness in secret key generation is its high estimation accuracy at low SNR levels [94]. However, it requires additional hardware and is more computationally complex.

Afterwards, the authors present and discuss the following steps for key generation:

1. **Initialization**: In this step, users exchange beacons or dummy data.

2. **Common source of randomness estimation**: Legitimate users estimate the physical layer channel characteristic based on the received signal from the other legitimate node.

3. **Quantization**: The users convert the estimated common source of randomness to bits. This topic is discussed thoroughly in [95].

4. **Encoding**: To avoid any mismatch in bit rate between two users, each quantized value is encoded.

5. **Information reconciliation**: This step is crucial since there may exist some differences in the generated bit streams between two users due to interference, noise and hardware limitations, which leads to inconsistency in the generated secret key. In this step, users make sure that the keys generated at both ends are the same.

6. **Privacy amplification**: This step is directly linked to the previous one since information reconciliation leaks some information, which will in turn be useful to an eavesdropper for the recovery of the secret key. To avoid

this issue, privacy amplification reduces the length of the output bits and prevents the leakage of any information related to the agreed key. This can be realized through a set of hashing functions [76, 94].

Three of the above steps are evaluated and simulated in [96], using Matlab: the distillation phase (common source of randomness estimation), the reconciliation phase and the privacy amplification phase.

Finally, the authors in [94] divide the commonly used metrics in secret key evaluation into information theoretic metrics and statistical metrics. Information theoretic metrics include: secret key rate, secret key capacity and outage secret key capacity. Statistical metrics include: frequency test, serial test, poker test, run test, auto-correlation test and bit mismatch rate.

However, a more accurate categorization of evaluation metrics is shown in Fig. 3.3. Basically, evaluation is based on two important factors, the efficiency of a specific security algorithm and its security level. Efficiency includes the secret rate, latency and required resources, while security level includes resistance against attacks, independence and uniformity. The last two metrics are evaluated using statistical randomness tests. Unlike [94], Fig. 3.3 is more general.

In [97], the authors analyze and evaluate the secret key and privacy leakage rate of a binary secret key scheme, called fuzzy commitment. This scheme is different from previously discussed schemes, since a transmitter encrypts the secret key and then XORs it with the transmitted message. The resulting is referred to as the public data helper, and it is sent to the receiver through an authenticated, noiseless channel. The transmitter then sends the message through the noisy channel. Next, the received signal is XORed with the public data helper and the secret key is, thus, estimated. However, the assumption of having an authenticated link between two legitimate users is not practical.

The technique in [98], on the other hand, relies on a third party authority for session key distribution. This technique can't be generalized since in some cases a trusted public authority does not exist. Generally, end-to-end key generation and distribution protocols are more desirable.

In [99] a key extraction protocol for D2D communication is presented and is studied experimentally. It has been shown that adjacent or nearby sub-carriers have similar physical characteristics, thus, their corresponding CSI measurements may have strong correlations, which is a major vulnerability that should be addressed. For this purpose, the authors presented a fast secret key extraction protocol, which combines the information of all sub-carriers. This validation-recombination mechanism prevents attackers from obtaining the secret keys, and hence, achieves a high security level and a fast key-generation rate.

All of the schemes presented so far, mainly, depend on the channel and, more specifically, assume that the channel responses of both the sender and receiver are identical, which might not be the case at all times. The authors of [100] study, experimentally, the non-reciprocity factors of CSI from the perspective of

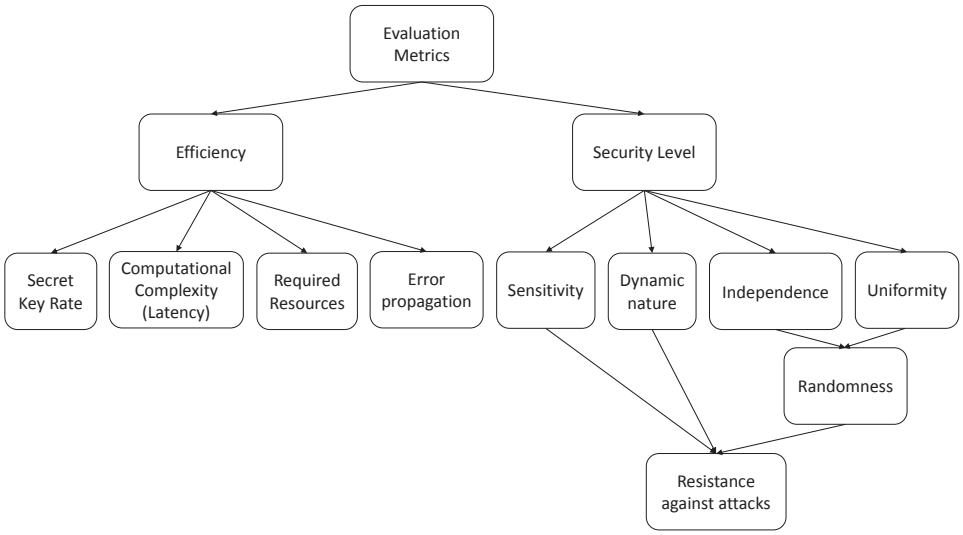


Figure 3.3: Chosen metrics used for the evaluation of key generation and encryption techniques

hardware devices mismatches and time delay. The evaluation is done using the Mean Square Error (MSE) algorithm.

In [10], the authors verify the feasibility of a key generation technique through implementation on the wireless open-access research platform [101] running an 802.11 OFDM system. The investigated scheme simply generates secret keys from the channel responses of individual sub-carriers in OFDM systems.

Unlike the techniques that preceded, [102] and [9] analyze the security of the shared session keys between users rather than introducing a new key generation and distribution technique. Authors in [9] mainly consider sophisticated attacks that enable an attacker to manipulate the key generation process and go unnoticed. More specifically, the following two types of attacks are considered: 1) different-key attacks and 2) low-rate key attacks. The former attack occurs when an insider tries to force different realizations of the shared secret key at different nodes. The latter occurs when an insider tries to reduce the secret-key rate by decreasing the channel variations over time. Whereas in [102], authors prove, through information-theoretic analysis, that the secret key capacity of the side-channel is lower than that of the wiretap channel. In addition, the authors analyze the electronic devices during randomness capturing and quantization and, consequently, show that the keys generated from the physical layer are susceptible to many threats.

The above mentioned key generation and distribution techniques can be summarized into five groups, shown in Figure 3.4.

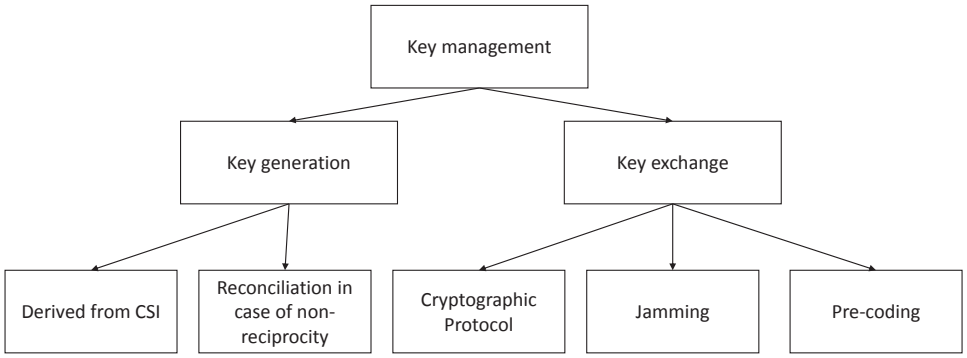


Figure 3.4: Existing key generation and distribution methods

3.1.3 Data Confidentiality

Figure 3.5 illustrates the existing PLS OFDM cipher schemes in the literature. These schemes are listed and described in what follows (eight sub-classes).

Encryption in OFDM-based systems is divided into two main classes: Pre-IFFT encryption and Post-IFFT encryption. In the former class, frequency-domain symbols are encrypted (before passing through the IFFT block), while in the latter, time-domain symbols are encrypted (after the IFFT block). Hence, data confidentiality schemes can be applied at different instances (locations) in the OFDM system as shown in Fig. 3.5, where either the time-domain symbols or the frequency-domain symbols or both, are encrypted and secured. However, there is no clear study in the literature that shows the effect of each case on the security level of data confidentiality techniques. Moreover, the domain of the data (frequency or time), which is subject to encryption using different schemes, is not justified. One question that directly comes to mind is: which of the following schemes is more secure, "Pre-IFFT" encryption or "Post-IFFT" encryption? Therefore, in Chapter 6, various data confidentiality schemes are tested under two scenarios: performing encryption before the IFFT transformation and afterwards. Results have shown that frequency-domain encryption (Pre-IFFT) performs better in terms of performance (lower Bit Error Rate (BER)), while time-domain encryption (Post-IFFT) is more secure.

Permutation:

One simple approach to secure transmitted data is through permutation and interleaving. In [103], the "cyclic delay perturbation on effective channel" scheme is presented for the MISO (Multiple-Input Single-Output) single-antenna-eavesdropper wiretap channel. The scheme improves PLS in OFDM systems by introducing a random cyclic delay (perturbation) to the transmitted signals. The random perturbation, which changes on symbol-by-symbol basis, depends on the CSI between the legitimate users and the total transmit power, both of which are

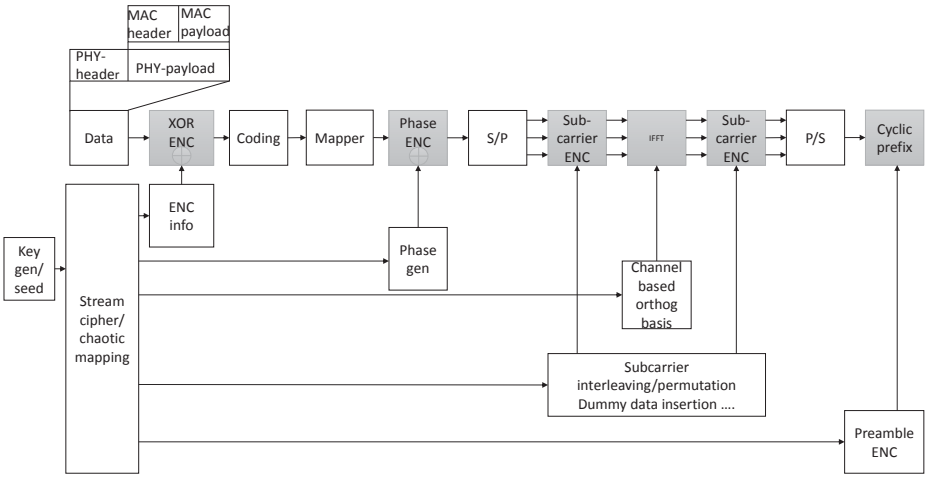


Figure 3.5: Existing OFDM PLS encryption schemes

available at the intended transmitter and receiver. Basically, the scheme is summarized as follows: before appending the cyclic prefix (CP), the time-domain signal, resulting from the IFFT block on a specific antenna, is cyclically shifted by a specific delay. As a result, the effective channels experienced by both the intended receiver and the eavesdropper are changed into linear combinations of phase-rotated channel gains. At the receiver's side, the cyclically shifted signal is multiplied by a weighting factor in the frequency-domain, such that only the intended receiver is able to generate this factor and recover the real signal. For this scheme, no additional information about the eavesdropper's channel is required at the transmitter's side and no additional information with the transmitter is necessary at the receiver's side.

Another technique for enhancing PLS is through signal interleaving, in which the time- and/or frequency-domain signals are shuffled based on chaotic pseudo-random sequences [104]. In [105] and [106], interleaving is done before and after the IFFT block based on a key stream (pseudo-random sequence) generated by a chaotic map, which results in a two-level data encryption since both the time- and frequency-domain signals are interleaved. This method is similar to the one presented in [107], except for the fact that securing the OFDM system is achieved using time-domain scrambling, only, which is based on a pre-shared secret key. On the other hand, chaotic-based encryption solutions have proven to be inefficient and not fully secure due to their poor cryptographic structure, in addition to having several disadvantages such as high computational complexity, memory and energy consumption.

In contrast, a new technique that provides practical secrecy is presented in [108]; OFDM sub-channels are shuffled based on the intended user's channel. Basically, the transmitter extracts the unitary matrices from the amplitude of the

channel frequency diagonal matrix of the legitimate user. The resulting unitary matrices are then decomposed using Singular Value Decomposition (SVD) and used as channel frequency-based pre-coder and post-coder. In this method, the authors rely on the channel randomness for achieving OFDM system security.

Moreover, random permutation can also be realized when the real and imaginary components of a symbol are interleaved, as is the case in [109]. Specifically, this is done when the channel phase of a sub-carrier symbol is larger than a predefined threshold.

The main advantages of the techniques presented in this subsection are simplicity, low computational complexity and low energy. However, these techniques lack the notion of secrecy, in which data shuffling is done based on known parameters that can be acquired easily.

Phase Encryption:

In [110], a PLS encryption scheme based on pseudo-random phase permutation is presented at the time-domain signal level. Basically, two pseudo-random sequences (Seq_1 and Seq_2) are generated using secure stream ciphering, where the first sequence is multiplied by the real part of the time-domain signal, and the second sequence is multiplied by the imaginary part. This, in turn, represents pseudo-random phase shuffling of the in-phase and quadrature symbol components. The two utilized sequences can either be $+1$ or -1, consequently, each symbol has only four possible combinations, making this scheme vulnerable to brute force or dictionary attacks, especially if a static secret key is used.

Channel-Based Data Encryption:

In order to achieve PLS, most researchers rely on the physical properties of wireless channels to derive shared secret keys.

One way of doing so is through active sub-carrier index selection. In the index selection scheme of [111], the data bits are divided into two groups. The first group of bits represents the indices of the active sub-carriers that will carry the second group of data bits, which are part of the real data. The selection of active sub-carriers depends on the CSI between the transmitter and receiver. These sub-carriers are arranged in descending order of their channel gains observed at the transmitter. Here, the eavesdropper receives incorrect symbols due to the unknown CSI used for index selection and data modulation. However, this is not always true since in case the eavesdropper is able to synchronize to the transmitted data, the CSI can be easily estimated. The authors of [112], also use sub-carrier index selection for securing physical layer data. The difference here is that only the sub-carriers experiencing high channel gains are used to transmit data. The main motivation behind this scheme is that while the channel gains of some sub-carriers are high between two users, these sub-carriers might experience low channel gains between another set of users. The main drawback of this approach is that it affects the data rate, dramatically.

A novel OFDM physical layer encryption scheme using dummy data insertion is presented in [4] and [14]. The main idea is to obfuscate the encrypted data streams by randomly inserting dummy data at random OFDM sub-carriers. As such, the encrypted data will be secured and only legitimate users will be able to 1) know the locations of the dummy data, 2) remove them in order to decrypt the sent signals and 3) recover the real data at the intended receiver. In [4], the location of dummy data differs between OFDM symbols. These locations depend on a secret shared "seed" between legitimate users. Thus, the seed serves two purposes; first, it is used for generating a stream cipher that represents the sub-carrier location of dummy data and second, it is used to generate another stream cipher, which represents the dummy data itself. At the receiver's side, data can be correctly recovered, since only the legitimate users share the encryption information. In this technique, the data rate is inversely proportional to the security level. Basically, the security of the presented scheme improves with the increased number of dummy data, which in turn, leads to reduced data rates. The authors also introduced an added security feature: training sequence re-arranging, which prevents an eavesdropper from performing synchronization and channel estimation, since the eavesdropper has no access to the new training sequence (only shared between the transmitter and receiver).

The authors of [14] design secure pilot signals to help the legitimate receiver differentiate between real and dummy data and in turn recover them. In particular, secure pilot signals are designed in such a way that only the intended user is able to locate the sub-carriers carrying real data. In the two previous methods, the eavesdropper has to guess the location of the real data and extract them from the received signal. This is only true when the eavesdropper has absolutely no knowledge about the channel between the legitimate users.

In [8], the authors present a novel scheme in which the traditional IFFT/FFT blocks are replaced with new ones. The new blocks are based on the channel between the legitimate transmitter and receiver and they are used to perform the modulation operation in a secure manner. Basically, new orthogonal bases are extracted from the channel, decomposed into new matrices using Singular Value Decomposition (SVD), and used to transmit and receive data. As such, only the legitimate receiver is capable of retrieving the sent data since he is able to estimate the channel and derive the basis used for transmission. Like previous schemes, the security of this method degrades when the eavesdropper is able to correctly estimate the channel, through preamble synchronization. In addition, this scheme requires matrix multiplication which introduces considerable overhead in terms of resources and latency.

Similarly, the transmitter in [113] estimates the channel between itself and the intended receiver and then, modifies the transmitted signal according to the CSI using pre-equalization. At the receiver, the pre-equalized signal will be received undistorted, only when passing through the intended channel.

The data encryption scheme presented in [114] works as follows: legitimate

users estimate the CSI between them, share transform orders and modulate data, accordingly. In this scheme, only users with valid transform orders are able to demodulate the data.

On the other hand, the authors of [115] exploit the fact that the imaginary part of transmitted symbols is usually unloaded, to introduce a new physical layer encryption method for OFDM/OQAM systems based on intrinsic interference. Consequently, the imaginary part of the symbol can be used as an encryption key to obfuscate the transmitted data symbols. For eavesdroppers, it is very difficult to recover the data since a secret key is required. In contrast, legitimate receivers are able to recover data by eliminating the interference completely.

Different from what preceded, the authors of [116] introduce the idea of each sub-carrier having a distinct initial phase. In particular, a new mechanism to achieve angle-range-dependent physical layer security for point-to-point communications is presented, in which the transmitted OFDM symbols are well preserved in a specific location, while symbols are scrambled at all other locations. The algorithm produces a unique phase for each sub-carrier of the OFDM symbol in baseband. Here, a MISO system is considered. In traditional beamforming, each element is equipped with a phase shifter and each symbol is weighted with a specific weight to target a desired transmission direction [116]. At the receiver's side, the modulated OFDM symbols are recovered, correctly, along a specific direction but not elsewhere. However, this method cannot guarantee good security when an eavesdropper is equipped with a sufficiently sensitive receiver. For this reason, the authors resort to an enhanced and more secure technique than conventional beamforming.

Two types of encryption techniques are presented in the literature, Exclusive OR (XOR) and phase encryption. In [117], these techniques are compared in terms of decoding symbol rate, where the first technique is a conventional encryption scheme: stream cipher encryption using the XOR operation, and the second is achieved by multiplying the real and imaginary components of the time-domain OFDM samples by two binary key streams $\{1, -1\}$.

The chaos I/Q-encryption (Inphase/Quadrature) technique presented in [118] is also a step forward towards a more secure physical layer. The technique can be summarized as follows: after serial-to-parallel conversion and QAM mapping, QAM symbols are split into two parts: In-phase and Quadrature-phase. Each part is then multiplied separately by a phase sequence, which is generated using a chaotic map. The technique presented in [119] is also based on a chaotic system, which generates 3-D chaotic sequences. However, in this scheme the generated sequences are used to form the training sequences of the OFDM frames, perform OFDM sub-carrier masking and control the fractional order of the FFT operation. Similarly, this approach suffers from the previously mentioned limitations and challenges of chaotic systems.

Another technique that utilizes chaotic mapping is presented in [120]. Here, authors apply logistic chaotic maps to enhance the security of OFDM systems

in Visible Light Communication (VLC). More specifically, the proposed scheme exploits the random nature of the physical channel, mainly CSI, and the symbol's CP to:

1. Permute frequency-domain symbols using column/row permutation,

2. Encrypt time-domain signals using a secret key.

Typically, a chaotic system introduces a significant overhead in latency and resources, since it is based on floating-point arithmetic. Moreover, the secret key depends on the symbol CP, which copies the last bits of the previous symbol and multiplies them with the channel-based chaotic sequences to encrypt the current symbol. This is considered a major weakness since a secret key should be independent of plaintext/ciphertext. In fact, the secret key should never depend on the transmitted payload and consequently shouldn't depend on the CP due to interference and transmission errors.

In contrast, authors in [121] exploit both, channel characteristics and the OFDM symbol structure (CP), to enhance the security of OFDM systems using the channel shortening method. More specifically, they propose using a smaller CP and applying channel shortening at the transmitter's side, in such a way that the effective channel at the receiver's side does not cause ISI, while effective channel of the eavesdropper causes ISI.

What is common between all of the presented techniques, is that the extracted CSI is used as a secret key between legitimate users to encrypt/decrypt data (data confidentiality). The main motivation behind this approach is that wireless channels are unique, dynamic (always varying) and pseudo-random. Hence, channel characteristics and parameters can be used in the encryption process, specifically, in the secret-key generation process to enhance the security of exchanged data. However, the above techniques suffer from a major drawback since they all rely on the estimated CSI, solely, and assume that the eavesdropper is unable to estimate the CSI between legitimate users. In reality, the eavesdropper can acquire the CSI if he is able to synchronize to the transmitted data, which is achieved through preamble detection and synchronization.

Artificial Noise and Artificial Fast Fading:

The technique presented in [122] requires the cooperation of both the sender and receiver, on one hand, and the addition of Artificial Noise (AN) to the transmitted time-domain signal, on the other. Also, an OFDM single-antenna relay system is considered. Unlike most existing AN techniques, which only consider MIMO systems, the authors consider a system where all users are equipped with a single antenna. The mechanism is summarized as follows: the sender and receiver estimate the channel and obtain the CSI. Using the IFFT block, the sender transforms the frequency-domain signal to time-domain, appends the CP at the

beginning of each OFDM symbol, adds the AN, which is derived from the estimated CSI, and transmits the signal to the relay. Then, the receiver sends a jamming signal to the relay, which is regarded as self-interference and which, in turn, cancels the AN. Consequently, the eavesdropper receives the distorted signal and is unable to recover it.

In [123], the authors use the CFO as a new feature to incorporate AN in a PLS scheme. In principle, CFO is the difference between the frequencies generated by the transmitter's and receiver's local oscillators. The basic idea is to pre-compensate the CFO in the transmitted data in such a way that when it passes through the legitimate receiver's channel, the signal will be received without ICI.

Additionally, Artificial Fast Fading (AFF) is exploited in [3] to improve the energy efficiency of MISO systems using OFDM. This method is also beneficial to PLS since the pre-introduced weights at the transmitter's side, which are derived from the CSI of the channel shared between two users, will be cancelled out when propagating through the intended channel. Thus, only the intended receiver will be able to recover the real data.

As it can be inferred, the mentioned schemes use CSI to pre-equalize or pre-compensate the transmitted signal such that this added noise would cancel out when passing through the legitimate user's channel. Considering that the eavesdropper is able to access the needed information (eavesdropper is near legitimate user, same channel conditions), the performance of these techniques would suffer greatly.

Preamble Encryption:

In order to prevent an eavesdropper from estimating the channel between two users and extracting the CSI through signal synchronization, preamble security should be enhanced.

Generally, especially in OFDM-based systems, the receiver doesn't know the exact signal of the preamble but knows its structure. This has motivated the authors in [41] to construct several new, but compliant preamble waveforms. To generate such waveforms, a new technique, called preamble modulation (P-modulation), is presented. P-modulation is a combination of two signal processing techniques: a shift in time-domain and a phase rotation in frequency-domain. Differently, the mechanism presented in [124] manipulates the preamble sequence based on the CSI between two users. Here, two power efficient algorithms are introduced, enabling the intended receiver to correctly estimate the channel.

Moreover, another way to improve preamble security is through embedding user-specific data, or sequence permutation. In [4], as mentioned earlier, the LTS is re-arranged in a manner only known to the legitimate users, which will prevent CSI and CFO estimation at the attacker's side.

Securing the packet preamble is necessary to prevent eavesdroppers from synchronizing with legitimate users, thus, acquiring the channel state information. However, this is just one layer of security which should complement other data

confidentiality schemes that secure the transmitted data itself.

Power Allocation:

A technique referred to as "frequency diverse array beamforming" is presented in [125] and [126]. The main idea is to maximize the secrecy rate by carefully designing the following parameters: frequency offsets across the antennas and the transmit beamformer. First, the channels of both the legitimate user and the eavesdropper should be maximally decoupled by optimizing the frequency offsets across the array elements. Essentially, the frequency offset values affect the phase lags among different array elements, which consequently leads to independent channel characteristics among different users. Given the set of frequency offsets, the transmit beamformer can be optimized.

The commonly used PLS schemes such as beamforming and AN-insertion fail in certain environments due to the high correlation between the eavesdropper's channel and the legitimate users' channel, such as the case in Line-of-Sight (LOS) mmWave communications. To solve this issue, several techniques related to power allocation for securing communication are presented in the literature [127, 128].

One of the main features in OFDM-based networks is that each sub-carrier experiences different gains values. In [129], a two-step PLS scheme is presented which includes optimal power distribution and multi-user sub-carrier allocation. Initially, the transmitter estimates and then allocates the optimal transmit power for each sub-carrier. Afterwards, two algorithms are introduced to effectively allocate sub-carriers among several users. Accordingly, both the secrecy rate and the performance in terms of fairness among users are greatly improved.

In [130], authors jointly optimize sub-carrier power allocation and the covariance matrix of the time-domain artificial noise to improve the secrecy rate (non-concave function).

The resource allocation problem to achieve PLS in heterogeneous networks is also addressed in [131]. The main idea is similar to the one presented above: finding transmit power vector (also referred to as sub-carrier allocation vector), which is modeled as an optimization problem. In other words, the main target is to find the optimal power for each sub-carrier, in addition to finding the optimal set of active sub-carriers.

A similar concept to power allocation is optimal resource allocation for secure communication networks. The work presented in [132, 133, 134] is based on the fact that the security in a multi-user OFDMA-based system can be significantly improved by well-designed resource allocation schemes, which enhances greatly the secrecy rate of users.

On another note, the basic idea behind the approach presented in [135] is that the cooperative jammer uses the harvest-then-jam protocol to ensure the protection of confidential information transmission. The transmission block is divided into two time slots. In the first slot, the source sends dedicated energy signals to the receiver, which acts as the jammer. In the second slot, the jammer

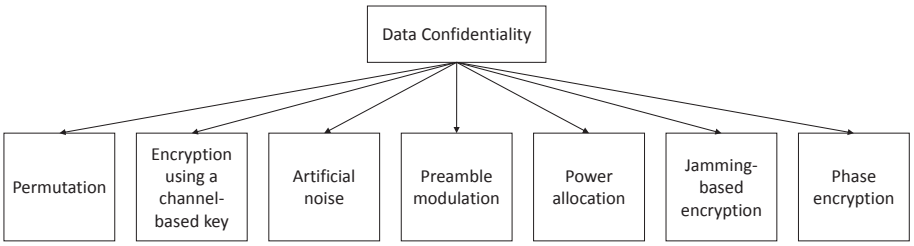


Figure 3.6: Proposed data confidentially technique classification for OFDM systems

uses the harvested energy to interfere with the eavesdropper.

In this type of PLS schemes, transmitted data is not secured (encrypted), but rather sent according to an optimal resource/power allocation mechanism. This is not enough to ensure robust data confidentiality in wireless networks.

Jamming-Based Encryption:

In order to avoid the challenge of key generation and distribution between users, authors in [136] design and implement a keyless acoustic short-range communication system, PriWhisper, which exploits a friendly jamming technique from radio communication. The system allows the receiver to send a random jamming signal while the transmitter is transmitting, and since the receiver knows the jamming signal it is emitting, it can simply remove the noise and receive the intended data. However, jamming cancellation can be a very exhaustive and computationally complex task, which can be done by either generating an antidote signal or estimating the jamming signal from the received signal. Both techniques require a lot of resources and, thus, are not feasible for implementation.

Joint PLS Enhancement and PAPR Reduction:

A major disadvantage of OFDM is having the overall instantaneous power more than the average power, leading to high PAPR. In principle, high PAPR values result from high time-domain peaks that are produced when independently modulated sub-carriers are coherently added. Therefore, several works in the literature have been presented to jointly reduce PAPR and enhance PLS such as in [118] and [137]. These techniques have already been discussed earlier.

The different schemes that target data confidentiality are summarized and compared in Figure 3.6 and Tables 3.2 and 3.3.

3.1.4 Source Authentication and Message Integrity

Message integrity is the absence of inconsistent data, which should be correct, valid, and accurate at all times [138]. The main rationale behind the concept of message integrity is to make sure that the sent data has been correctly received

Table 3.2: A summary of the OFDM data confidentiality schemes presented in the literature

Data confidentiality for OFDM systems	**Permutation**	**Phase encryption**	**Artificial noise (AN) and artificial fast fading (AFF)**	**Preamble encryption**
Location	Pre- and Post-IFFT	Pre- and Post-IFFT	Pre- and Post-IFFT	Pre- and Post-IFFT
Advantage	Low computational complexity and low energy consumption	Simplicity and low cost	Automatic cancellation of AN and AFF when passing through the intended channel. No further actions done at the receiver	Eavesdropper can not estimate CSI
Limitation	Weak level of security since it is based on channel properties only	Vulnerable to brute force attack since there are only four possible combinations that can be used for encryption	Eavesdropper can recover transmitted data if it is aligned with the legitimate receiver	Additional operations and costs are introduced at both ends. More delay
Resource and communication cost	No additional cost and overhead	Generation of two pseudo-random sequences	Generation of AN and AFF matrices from the decomposition of the channel matrix	Generation of alternate LTS sequences
Complexity	Not computationally complex: performing shuffling and permutation	1) Divide each complex symbol into real and imaginary, 2) multiply them with the generated sequences and then 3) form again the complex symbol	1) Extract information from channel. 2) Generate AN and AFF. 3) Include these signals in transmitted data	1) Generate sequence. 2) Encrypt packet preamble

Table 3.3: A summary of the OFDM data confidentiality schemes presented in the literature (continued)

Data confidentiality for OFDM systems	**Channel state information (CSI)**	**Jamming-based encryption**	**Encryption/ PAPR reduction**
Location	Pre-, within and Post-IFFT	Post-IFFT	Pre-IFFT
Advantage	Ciphertext is resilient to cryptanalysis	Data hiding	Joint PAPR reduction and enhanced security levels
Limitation	CSI can be easily obtained if the eavesdropper is synchronized with the legitimate users	Data distortion. Synchronization between transmitter and receiver is required (jamming signal and real signal should be transmitted at the same time to confuse the adversary). Reduction in data rate	Large delays and computationally complex. Also the receiver has to detect the sequence used for the transmitted data
Resource and communication cost	No additional resource and communication cost	Additional resources are needed to send jamming signals (more power and delay)	Generation of several sequences
Complexity	Data multiplication with extracted channel information	Generate and send jamming signals (computationally exhaustive)	1) Generate several sequences, 2) choose one with the lowest PAPR

with no alterations or modifications, which might result from different factors such as malicious attempt, human error or hardware failure. Usually, message integrity is realized using hash functions. A MAC (Message Authentication Code) [139], on the other hand, ensures source authenticity along with message integrity, in which the intended sender and the correctness of the message are both verified. The difference between a hash and a MAC is that a MAC requires a key. In a MAC operation, a user hashes the intended message and then encrypts the resulting digest with a key. At the receiver's side, the same steps are followed to generate the keyed or un-keyed hash digest. The obtained value should be equal to the received value.

Generally, keyed hash functions employ HMAC [140], or a robust block cipher such as AES (Advanced Encryption System) with CBC mode (CMAC [141] and GMAC [142]). The AES technique can be employed using various key sizes: 128, 192 and 256 bits [143], which determines the number of required encryption/decryption rounds. For instance, 10, 12, and 14 iterations are required for a 128-, 196-, and 256-bit key, respectively. Typically, block ciphers are faster to execute while hash functions are more secure. Hence, there exists a trade-off between security level and efficiency.

In contrast, un-keyed hash functions, which are referred to as modification detection codes, only provide message integrity (proving that the received data is unaltered, without authenticating the origin of received data) [144]. Examples of traditional un-keyed hash functions include: MD4 [145], MD5 [146], SHA-1 [147], SHA-2 [148] and SHA-3 [149]. This class of hash functions also requires multiple rounds, for example, the SHA-256 and the SHA-512 require 64 and 80 rounds, respectively. As a result, conventional keyed and un-keyed hash functions suffer from large delays and high complexity, which is not suitable for resource-limited devices.

Regarding source authentication and message integrity at the physical layer, very few work have been presented, so far. In particular, message authentication techniques based on PLS are divided into two classes: Channel State Information (CSI)-based message authentication and Channel Pre-coding (CPC)-based message authentication. Both of these methods suffer from major drawbacks [150, 151]. The first class mainly depends on the CSI between two users to authenticate exchanged messages. However, CSI is a public parameter and it can be acquired by illegitimate users, hence, this method is not completely secure. In contrast, CPC-based message authentication leverages hash functions and secure channel codes to achieve information-theoretic security and guarantee the needed requirements. The issues associated with this type of solutions are: low efficiency and high computational complexity. Also, these schemes always assume that the legitimate receiver is closer to the legitimate transmitter than the adversary which is not necessarily true. Therefore, new solutions should be studied and proposed in this field [150, 151].

Recently, a different paradigm has been proposed and studied for the con-

struction of hash functions. It is based on chaotic systems, which are complex non-linear transformations that possess a high level of randomness. In general, chaotic maps can be either based on an integer transformation or a non-integer transformation [152].

3.1.5 Data Availability

Jamming attacks are one type of Denial-of-Service (DoS) attacks that threaten the performance of wireless communications, where jamming signals are intentionally emitted to disrupt communication in terms of throughput degradation and network availability, by simply occupying the channel and blocking legitimate communication [153].

In [154], the authors present a novel jamming-resistant scheme that estimates the channels of the legitimate receiver and the jammer using an unused orthogonal pilot sequence. The transmitter picks a specific pilot sequence from a pilot codebook after a certain number of symbols in a coherence block. It is assumed that there is at least one unused pilot sequence and that the transmitter uses a pilot hopping scheme (pseudo-random) such that the jammer is unable to know the user's current pilot sequence. The communication between the transmitter and receiver is divided into two phases. In the first phase, the transmitter sends a pilot sequence to enable channel estimation at the receiver, and in the second phase, the transmitter sends its payload data. It is assumed that the jammer performs jamming in both phases. To eliminate jamming in the first phase, the receiver (BS) projects the received signal (legitimate user's pilot sequence and jamming signal) onto the unused orthogonal pilot sequence, so, the user's pilot signal is eliminated leaving the jamming signal only. Note that the jammer chooses a jamming signal not orthogonal to the signal present in the pilot codebook. Now, the receiver has estimated the channel of both the legitimate user and the jammer and is able to amplify the desired signal on one hand and mitigate the jamming signal on the other. It should be noted that this technique is ineffective if the jamming targets the whole band.

Another defense mechanism to achieve jamming resilient OFDM systems is presented in [153]. This scheme tracks the direction of the jamming signal using multiple pilots and then cancels it out. The authors rely on the fact that the jamming signal is most effective when the angle between itself and the transmitted signal is zero, that is, when both signals are aligned, and is least effective when both signals are orthogonal. In other words, the angle between the transmitted signal and jamming signal should be close to 90 degrees. As a result, the presented scheme aims to project the received signal to a direction orthogonal to that of the jamming signal. In order to do so, multiple pilots are inserted in the payload. Upon jamming, the inserted pilots, which are already known, will be corrupted (jammed). Afterwards, the jammer's channel is estimated by comparing the original pilots to the jammed ones. Consequently, the receiver will be able to

know the direction of the jammed signal and will be able to rotate the received signal in a direction orthogonal to that of the jammed signal.

In [155], an anti-jamming technique is presented whereby a friendly jammer blocks unauthorized wireless transmission whenever unauthorized users are detected (by jamming) and stays silent, otherwise. Specifically, the friendly jammer has to identify whether the ongoing transmission is authorized or not using a special preamble, that is generated by the authorized user using a shared secret key between itself and the friendly jammer. Consequently, the friendly jammer verifies the special preamble using the secret key and stays silent, otherwise, launches a jamming attack to prevent unauthorized communication.

Table 3.4 summarizes and compares the existing anti-jamming solutions present in the literature, in terms of multiple factors.

3.2 PLS Schemes for NOMA

In this section, various PLS schemes for NOMA systems (PD-NOMA and CD-NOMA), are reviewed. It should be noted that all of the listed schemes address data confidentiality, and no work targeting other security services have been presented, so far.

3.2.1 Data Confidentiality for PD-NOMA

In [156] and [157], authors have recommended using two user-specific parameters, which are the media access control address and the international mobile equipment identity, to improve the security of the SIC algorithm in PD-NOMA. Generally, conventional SIC provides no privacy or data protection, since near users can sequentially decode the information of farther users. Hence, the security of users with weaker channel conditions (FU), is compromised. The proposed scheme modifies the conventional SIC scheme by introducing two keys or two verification steps. The first verification step is applied before selecting the signal with the highest strength, and it is used to ensure that the user performing this procedure is a legitimate user ("Key-1"). The second modification adds a second key, "Key-2", after the "decode selected user" step, to restrict the correct decoding of messages to legitimate users, who own the corresponding media access control address and the international mobile equipment identity. The proposed technique prevents internal users from acquiring all messages except their own, which also guards against illegitimate users. It modifies the conventional SIC algorithm and increases the delay for recovering the intended messages by using hash functions. Although authors in this work claim that the presented scheme enhances the confidentiality and security of exchanged messages, the presented scheme only proves that communicating users are legitimate users (user authentication only).

On the other hand, the work in [158] evaluates the secrecy performance of

Table 3.4: A summary of the OFDM anti-jamming schemes presented in the literature

Anti-jamming scheme	**Pilot-index encryption**	**Jamming signal estimation**	**Pilot insertion within payload**	**Jamming the jammer**
Advantage	Low latency and complexity	No additional overhead in terms of exchanged messages and latency	Accurate estimation of the jamming signal within the jammed pilot	Prevents the jammer from sending jamming signals in real time instead of waiting for the authority's response
Limitation	If the attacker has enough power, the whole bandwidth would be affected (no need to know the pilots)	Jamming signals can not be estimated or known since these signals are random	The jamming signal can't be fully known from the jammed pilots within the payload. This technique decreases the data rate	An extra node monitoring the network is required. The friendly jammer and legitimate user should be able to share a secret key securely
Resource and communication cost	Encryption of pilot indices and exchanging them which requires additional communication cost	No additional costs in terms of resources and communication	Using additional resources for pilot insertion in payload since pilots replace part of the intended data	Additional devices to constantly monitor the network are needed in addition to power resources
Complexity	Not computationally complex: encrypting pilot indices not the pilot signal itself	Computationally complex to estimate a jamming signal or even a part of it	Computationally complex: 1) pilot insertion upon transmission. 2) Jammed pilot extraction upon reception and comparison with original pilots	1) The friendly jammer should first detect whether unauthorized users are present 2) Generate jamming signals

a cooperative PD-NOMA network. The legitimate users are paired randomly such that one user is the strong user (NU) and the other is the weak user (FU). Also, two cases are considered: 1) security is provided for the message of the stronger user (when the data rate of the stronger user is greater than that of the weaker user) and 2) security is provided for the messages of both users. The lower and upper bounds of the secrecy outage probability are derived for cases 1 and 2, respectively. Consequently, two strategies are presented: in the first, the stronger user acts as a relay and allocates all of its transmitting power for sending the message of the weaker signal, whereas, in the second, the stronger user acts as a friendly jammer, in which it allocates some of its transmitting power for sending the message of the weaker user and the rest of its power for sending noise. Here, it should be noted that this technique is rather complex since it requires additional resources, overhead, and synchronization between users.

A cooperative relay network is also considered in [159]. The network includes two transmitters and two receivers and a set of relays, where one relay is selected to decode-and-forward the information to the two destinations, respectively. Here, PD-NOMA strategies are applied in the uplink and downlink. Specifically, two users send their information to the selected relay, which performs SIC to recover both signals. Then, the relay re-generates the superimposed composite signal from the signals of users 1 and 2, and transmits it to both receivers. First, the stronger user decodes the signal of the weaker user, then it obtains its own. The weak user directly decodes its signal, treating the other signal as noise. Here, the strong user is still able to acquire the information of the weak user, which is a major drawback.

Authors in [160] present a new and secure PD-NOMA transmission strategy for cognitive radio inspired NOMA networks, having multiple primary and secondary users. Using this approach, both primary and secondary users are paired (superimposed) according to their channel gains to perform PD-NOMA. The power allocation and transmit rate control techniques are investigated to prove that the SIC process has been performed successfully at primary users. Besides, closed-form expressions for connection outage probability, secrecy outage probability, and effective secrecy throughput are also derived.

Another approach for enhancing the security of MISO-NOMA systems (Multiple-Input Single-Output) is presented in [161]. This solution is based on an AN-aided (Artificial Noise) beamforming problem, which is studied under a practical non-linear energy harvesting model. In particular, the transmission beamforming and AN-aided covariance matrix are jointly optimized to minimize the transmit power of the base station, while the secrecy rates of users and the energy harvesting requirement are satisfied. Two algorithms have been presented to solve this non-convex problem. Similarly, an interference-alignment transmitting zero-forcing-beamforming approach is presented in [162] for downlink MIMO-NOMA in cognitive radio networks. Here, two base stations design their transmitting zero-forcing-beamforming vectors in order to improve PLS for cell-center and

cell-edge users in both cells. The main drawback of this technique is that whenever the adversary is near the legitimate user (or aligned), he will be able to intercept the transmitted signals, easily.

Artificial noise is also utilized in the scheme presented in [163], where a network, consisting of a relay and two users, is considered. The relay is assumed to be operating in full-duplex mode to receive signals from the legitimate users, while radiating jamming signals simultaneously, to guard against eavesdroppers. The main advantage of introducing artificial noise as jamming signals is to confuse the eavesdropper without interfering with the signals of the legitimate users (exploiting the null space of the legitimate channels). This technique is beneficial only when illegitimate users are far from legitimate users. Authors in [164] also follow a similar methodology.

A different technique for securing PD-NOMA using AN is presented in [165]. The presented technique is summarized as follows: first, users send uplink training sequences to the base station (minimum mean-squared-error estimation at the base station). Based on the estimated Channel State Information (CSI), the base station pre-codes the confidential information and injects AN. This noise will be canceled out when passing through the intended channel.

Currently, beamforming is a popular technique used to ensure and enhance security in PD-NOMA systems. Authors in [166] present a novel hybrid beamforming scheme for PD-NOMA systems, where the utilized beamforming vectors of the weak and strong users are linear functions of their channel vectors, respectively. In this scheme, AN is also used to further improve the secrecy performance of PD-NOMA systems.

Authors in [167] utilize the null space approach of interference cancellation in their proposed system. Specifically, this scheme cancels the interference between different multi-cast groups, which are mainly divided into three groups: the "NOMA group" (high-security group), the low-security group and the "non-NOMA group". The signals of the "NOMA group" users and the low-security group users are superimposed. On the other hand, the low-security group subspace is orthogonal to the "non-NOMA" group subspace. Once the relationship is established, the null space of every group can be utilized. To cancel the interference among different multi-cast groups, the authors in this work have proposed a projection matrix for a certain group by null space. In this way, the signals of the "NOMA group", which have a higher security priority, will not be obtained by the low-security group. Moreover, the "NOMA group" employs SIC to obtain signals of the low security users, and its own signals. In contrast, the low-security group does not employ SIC, hence it will only be able to obtain its signals. The main drawback of this scheme is that the signals of the low-security group are still being obtained by another group of users, and no security measures are employed to solve and overcome this issue. In principle, all users should be able to protect their information (confidentiality and privacy) from other legitimate and illegitimate users. A similar technique is applied in [168], where the base

station degrades the eavesdropper's channel using a jamming signal, that is injected into the system by means of null-steering beamforming, without affecting the legitimate users.

A lot of research work in the literature focuses on deriving and optimizing analytical expressions for the secrecy outage probability, connection outage probability and secrecy capacity, to evaluate the security performance of a power allocation scheme, a resource allocation scheme or a scheduling algorithm [169, 170, 171, 172]. However, very few papers in this area target the confidentiality and privacy of data transmitted in networks utilizing the PD-NOMA technology. In particular, authors in [173] have proved that solving the joint optimization problem of power allocation, user pair selection, and time-frequency resource allocation amounts to solving a so-called iterated function without a closed form, where the algorithm reaches optimality with fast convergence. Also, in [174] novel beamforming strategies for the direct transmission PD-NOMA and cooperative jamming PD-NOMA with a helper are presented. The problem is formulated as the worst-case sum power minimization subject to secrecy rate constraint. An iterative algorithm, which is based on successive convex approximation is presented to transform the non-convex problem into convex approximations.

In contrast, the presented cluster-based downlink PD-NOMA system model in [175] groups legitimate users into two sub-categories: security-required users and QoS-required users (Quality-of-Service). Then, several clusters that consist of one security-required user and multiple QoS-required users are formed. Users in each cluster are served by a common beam using PD-NOMA. Since random classification leads to poor performance, authors design an effective user scheduling mechanism to achieve better secrecy performance and spectral efficiency.

In [176], authors utilize cooperative PD-NOMA to improve the performance of weak users. Since users with better channel conditions have prior information about the messages of other users, strong users can help in enhancing the performance and security of users with weaker channel conditions.

Authors in [177] benefit from public/private key encryption to secure the data of each user in two steps. Specifically, the data of each user is encrypted using its public key, such that other superimposed users can decode, but not recover, signals expect theirs. Also, the base station encrypts the entire transmitted signal (composed of multiple superimposed signals) with its private key (symmetric encryption) so that only legitimate users can correctly recover the transmitted signals. However, public/private cryptography is complex and requires a significant overhead (multi-round operations), hence, it is not practical and un-preferable in today's systems.

On the contrary, authors in [178] adopt the concept of a timing channel, which refers to the interval between every two transmitted symbols. This feature is exploited, here, to secure data. Specifically, authors have designed a time constellation, in which every point represents several symbols and the intervals

between these symbols. Different points carry different information using several intervals of varying lengths. Accordingly, one-time constellation point is transmitted to different users, carrying different data since different mapping rules are allocated to different users.

Finally, authors in [179] utilize several techniques to secure primary users in cognitive radio networks. The presented scheme is divided into two phases (two time-slots). In the first time-slot, the primary signal is transmitted from the primary transmitter to the relay. Simultaneously, the relay uses some of its antennas to generate AN to disrupt the eavesdropper. In the second time-slot, the relay superimposes and transmits the signals of the primary and secondary users (the primary user is assumed to be farther than the secondary user). Primary security is guaranteed by the modified decoding order and beamforming optimization, which is converted to convex and solved by an iterative algorithm. This method is only applicable in a subset of networks, which have relays.

The PLS schemes for PD-NOMA are summarized in Tables 3.5 and 3.6, along with their advantages, limitation, overhead, and complexity.

3.2.2 Data Confidentiality for CD-NOMA

In [180], authors present a novel chaos scrambling NOMA scheme that outperforms Multi-User Shared Access (MUSA) and Sparse Code Multiple Access (SCMA). In particular, each scrambling sequence is generated by a specific chaotic equation and it is user-specific (secure). At the transmitter, the modulated symbols of each user are, first, scrambled by a unique chaotic sequence (acts as the unique code of each user), and then sent on the same frequency spectrum. However, this technique is considered rather unpractical since generating chaotic sequences requires a lot of resources and overhead. Hence, the presented method is not efficient to implement.

Differently, in [181], authors target downlink Space Code Multiple Access (SCMA) systems to enhance the security and confidentiality of exchanged messages, in the presence of external eavesdroppers. Specifically, a novel and secure transmission approach, based on a highly structured SCMA codebook is presented using physical layer security. Using the extracted channel phases from the Channel State Information (CSI) of each user, the base constellations are rotated pseudo-randomly, hence, data confidentiality is achieved. The security of this technique, mainly, depends on public parameters extracted from the shared wireless channel between users. Since the CSI of users can be acquired by illegitimate users, the security of the presented technique is compromised (limitation). To overcome this issue, the security of the SCMA coding process can be enhanced by using pseudo-random non-orthogonal matrices that depend on a dynamic secret key. This solution can be based on the work in [182, 183]. The main advantage of this method is that the encryption and the coding processes are combined in one step, which means less computational resources and requirements are needed.

Table 3.5: A summary of the PLS data confidentiality PD-NOMA schemes

Data confidentiality schemes	**Beamforming**	**Optimization problem of power/resource allocation and scheduling schemes**	**Securing transmitted data using asymmetric encryption (public/private keys)**	**Cognitive radio inspired NOMA**
Advantage	Simple and it does not require any additional overhead	Theoretical analysis and mathematical derivations	No one except the legitimate user can decode its signals due to the confidentiality of the private key	Primary and secondary users can transmit data over the same time-frequency resources, simultaneously
Limitation	In case of alignment, illegitimate users will be able to recover the sent data	These schemes do not secure transmitted data itself	Asymmetric encryption is unpractical requires additional resources and overhead	Data is not protected from internal users
Resource and communication overhead	It does not require additional resources or communication overhead	Additional resources and communicated messages are required for scheduling algorithms between the BS and users	1) Secure exchange of public keys. 2) Encryption and decryption using public and private keys, respectively	No additional cost and overhead
Complexity	Not computationally complex	Most optimization problems in the literature are non-convex, hence, they are very complex algorithms	Asymmetric encryption requires multiple rounds in the encryption/decryption process (high complexity)	Not computationally complex

Table 3.6: A summary of the PLS data confidentiality PD-NOMA schemes (continued)

Data confidentiality schemes	**Hashing unique identifiers/parameters of users**	**Relay**	**Cooperative jamming**	**Artificial Noise (AN)**
Advantage	Authentication of users since only legitimate users will be able to generate the same hash digest	Non-Repudiation between devices	Complete data hiding from eavesdroppers	Automatic cancellation of AN when passing through the intended channel. No further actions done at the receiver
Limitation	Modifying the conventional SIC algorithm and increasing the delay for recovering the intended messages by using hash functions	Securing data requires two-time slots (delay) and the relay is vulnerable to being impersonated	Synchronization between users is required. Besides, more resources, power, and overhead are introduced	Eavesdropper can recover transmitted data if it is aligned with the legitimate receiver. Transmitted data is not encrypted
Resource and communication overhead	During the SIC process, users need to perform two hashing operations	Confidentiality is achieved through multiple rounds of communication. Additional resources are required	More power is needed to generate and send jamming signals	Generation of AN
Complexity	Computationally complex: hashing	Not Computationally complex: decode and forward operation	Computationally complex: jamming is an exhaustive operation	1) Channel estimation. 2) generation of AN. 3) Appending AN to the transmitted data

3.3 PLS Schemes for MIMO

The MIMO-based PLS solutions in the literature are classified into: device authentication schemes, key generation and distribution schemes, data confidentiality schemes and data availability schemes. It should be noted that no PLS source authentication and message integrity schemes exist for MIMO systems.

3.3.1 Device Authentication

In [184], authors proposed two mechanisms to overcome the fake client attack and the fake access point attack. In the first attack, the attacker provides the access point with forged CSI upon receiving the training sequence, while in the second, the attacker impersonates the access point and sends fake training symbols to the client, which generates wrong CSI and reports them back to the access point. To overcome the Fake client attack the following procedure is proposed:

1. The access point sends a training frame.

2. The client receives the training frame multiplied by the channel coefficient, estimates the obtained channel coefficient and sends it back to the access point.

3. The access point performs beamforming based on the received estimated channel coefficient and sends a pre-coded verification code.

4. Again, the client estimates the verification code and sends back the result to the access point.

5. The AP verifies the authentication code and authenticates the client.

In case the attacker sends a forged channel coefficient to the transmitter, the transmission of the verification code will result in a received signal at the attacker equal to null. However, if the eavesdropper is physically located near the receiver or transmitter, he will be able to obtain the verification code.

The second method which prevents the fake access point attack is summarized as follows:

1. The access point broadcasts a random sequence instead of the known training sequence.

2. The client receives the random number multiplied by the channel coefficient, and sends back an estimated channel coefficient value that is equal to the random number multiplied by the real channel coefficient value and divided by the training sequence.

3. Having the random value, the training sequence and the estimated channel coefficient, the access point estimates the real channel coefficient value.

It should be noted that the random number and the estimated channel coefficient are not encrypted, hence, any eavesdropper is able to recover the channel coefficient value in the same way as the access point.

On the other hand, the authors of [185] specify and compare three authentication schemes, that depend on the channel randomness between two users. The first scheme, physical layer authentication, compares the pilots transmitted at different instances to determine the authenticity of the source based on the similarity between these pilots. The asymmetric-key based authentication scheme exploits the channel as a random number generator to derive the public/private key pairs for each user, which will be used to encrypt and decrypt the information. The third scheme, symmetric-key based authentication, also extracts the common channel information between two legitimate users to derive a symmetric key, which is used in the authentication process. Here, it is assumed that the channel is secure, private and unique which is not the case since any user is able to synchronize to transmitted frames, perform channel estimation and extract channel-based information.

In [186] and [187], a channel-based MIMO authentication scheme based on game theory is presented. Basically, the spoofing node chooses its transmission rate in the zero-sum game. The receiver specifies its test threshold in order to detect spoofing and compares the estimated channel with the channel record of the transmitter.

The aforementioned authentication techniques, which are summarized in Table 3.7, exploit channel characteristics to authenticate communicating parties, especially those sharing the same channel. However, these techniques can't rely on channel characteristics solely and should depend on a shared secret between the users to enhance the security robustness of these methods and prevent forged messages.

3.3.2 Key Generation and Distribution

In [188], the authors present a key extraction scheme for PLS in a 2×2 MIMO-OFDM system. Basically, the channel matrix is clustered into groups, which are converted into key bits. This technique extracts the secret key between two users, directly, from the channel. However, this technique is considered weak since the channel matrix can be known by any eavesdropper. The authors of [189, 190, 191, 192, 193] also rely on channel reciprocity between any two users to generate the secret shared key. Having a secret key depending on the channel solely is inefficient since it can be easily intercepted by eavesdroppers.

Similarly, the presented phase randomization scheme in [194] evaluates the CSI between two users and consequently embeds key bits within a set of pre-

Table 3.7: A summary of the PLS device authentication schemes for MIMO systems

Device authentication schemes	**Beamforming based on channel characteristics**	**Generating a symmetric encryption key from channel characteristics**	**Generating an asymmetric encryption key from channel characteristics**	**Pilot comparison**	**Estimating the channel frequently**
Advantage	No additional cost and overhead	Achieving authentication and confidentiality at the same time	Achieving authentication and confidentiality at the same time	No additional cost or overhead	No additional cost or overhead
Limitation	Exchanged information are sent in plaintext	Generating encryption keys based on channel characteristics only is a weak assumption since acquiring channel information is not an impossible task	Asymmetric encryption is complex and can not only rely on channel characteristics	Can not rely on pilot comparison for the authentication process only	Additional channel estimations are required
Resource and communication cost	Authentication is verified through multiple rounds	Two steps are done: 1) Key extraction. 2) Encryption and decryption operations are performed	Two steps are done: 1) Key extraction. 2) Encryption and decryption operations are performed	For every specific period pilots are compared with previously acquired ones which are saved	Estimating the channel multiple times
Complexity	Not computationally complex	Not computationally complex: performing XOR operations	Computationally complex: asymmetric encryption	Not computationally complex: performing comparison	Not computationally complex

shared phase randomizing sequences, which are indexed based on a specific channel metric (key establishment). More specifically, the transmitter and receiver, first, collect the needed information for the generation of the phase randomization vectors. The extracted information, mainly, include the number of sub-carriers, the channel state information and the phase randomization vector size. Then, the indices of the obtained phase randomization vectors are re-ordered based on the capacity response, to achieve indexing privacy. This step is referred to as prerequisite planning and it is done at each of the transmitter and receiver, separately. Finally, key exchange and privacy amplification is applied, where the transmitter generates pseudo-modulated symbols to convey the indices of the phase randomization vectors to the legitimate receiver, who will, thus, be able to detect the secret key. In particular, the transmitter multiplies the modulated symbol vector with a shared phase vector. At the receiver, a maximum likelihood algorithm is applied to detect the phase randomization vector index bits. Only users having access to the legitimate channel information will have sequences indexed in the same order.

The authors of [195] and [196] adopt the same concept as above but use a different approach in which multiple incorrect symbols are transmitted, simultaneously, along with the correct symbol to confuse eavesdroppers. The legitimate receiver, on the other hand, can retrieve the correct symbol using the shared CSI with a legitimate transmitter.

Differently, a key generation method based on channel quantization with SVD is presented in [197]. This technique is divided into four main steps:

1. **Estimation of the complex channel coefficients' matrix**: Both users exchange reference signals to estimate the channel between them.

2. **Decomposition of the channel matrix**: First, the magnitudes and phases of the channel matrix are identified. Then, these matrices are decomposed using SVD.

3. **Generation of random matrices**: Based on the decomposed matrices, random matrices are derived.

4. **Reshaping to generate phase randomization vector**: From the random matrices, secret bits are obtained.

As it can be inferred, this technique suffers from the same fate as the previously mentioned techniques in which the secret key becomes publicly known if the channel matrix is recovered.

A secret key exchange scheme based on private random pre-coding in a MIMO system is presented [198, 199, 200]. In this scheme, the conventional MIMO codebook is manipulated based on pre-coding, where the pre-coding matrix index feedback is used to exchange secret keys among users. This procedure is divided

into three steps. First, the users find a common source of randomness through exchanging private preambles, which are jointly assigned to each of the already generated pre-coding matrices that are provided by the codebook. Based on the receiver's feedback, which conveys the optimal pre-coder index, a private version of the codebook is created after applying index rotation to the subspaces' indices. Finally, the secret bits are generated and shared in a secure manner between the sender and the receiver.

In contrast, the scheme presented in [201], divides the antennas at the transmitter's sides to two groups. The first group sends the secret key to the legitimate user while the second transmits jamming signals to the eavesdropper, simultaneously. Here, the location of the eavesdropper is considered to be known, also the antennas at the transmitter's side are considered to utilize different sub-carriers.

Finally, the main idea behind key exchange scheme presented in [202] and [203] is to rotate the constellation mapping based on the antennas channel gain between legitimate users.

3.3.3 Data Confidentiality

All of the PLS schemes, that are presented next, exploit the random properties/characteristics of the physical channel, to secure transmitted data and achieve data confidentiality using different techniques. In particular, three schemes (scrambling matrix, encryption based on CSI, artificial noise and artificial fast fading) require a channel-based key (extracted from the channel characteristics and properties) to achieve data privacy/secrecy, while the other three don't. Hence, data confidentiality can be achieved with or without a channel-based key.

Scrambling Matrix:

A channel scrambling scheme is presented in [204] to enhance the security of visible light communication systems against eavesdroppers and malicious attacks. First, the data stream is multiplexed into parallel independent data streams by a serial-to-parallel converter. These parallel data streams are then scrambled using a scrambling matrix, which is constructed from the location of the legitimate user and includes the angle of incidence, the angle of irradiance and the distance between the legitimate users. After constructing the scrambling matrix, iterative orthogonal non-negative matrix factorization is applied to reduce the correlation between the scrambling matrix and the channel gain matrix. At the receiver's side, the construction of the inverse scrambling matrix is straightforward since both the transmitter and receiver share the same location information and thus, the receiver is able to retrieve the needed information even when the CSI is imperfect. The concept of information scrambling is also applied in [205] and [206].

Artificial Noise and Artificial Fast Fading:

The authors in [207] present a new PLS technique for multi-user MIMO-OFDM systems. The technique assumes a non-existent receiver (imaginary receiver/imaginary channel response). The base station generates a pre-coding matrix in consideration of the imaginary users. When passing through the legitimate user's channel, the pre-coded signal will be successfully demodulated without interference only at the legitimate user. The advantage of this scheme is that the imaginary receiver does not feedback its channel response to the base station and hence, the eavesdropper is not able to recover the pre-coding matrix. Also, the authors modify the AN and the AFF schemes originally proposed for OFDM systems to cope with multi-user MIMO-OFDM systems.

The work in [208] and [209] also leverages the concept of AN to enhance the security of MIMO systems; however, the Alamouti scheme and artificial noise are combined. The main rationale is to generate the AN such that it lies in the null space of the receiver's channel.

AN is also utilized in [210] to provide secure communications between two legitimate users. More specifically, the main concept is based on the idea of using a portion of the antennas to send secret information to the intended receiver while using the other portion to transmit AN. The AN is chosen such that it lies in the null space of the receiver's channel. In other words, the receiver's channel can null out the AN, unlike the eavesdropper's channel, which is different from that of the receiver. This scheme is beneficial if the eavesdropper is at a different location, specifically, if the eavesdropper is not aligned with the receiver. Otherwise, both will experience the same channel conditions and the security of this scheme is jeopardized.

In addition, AN is used in [211] but in a different manner whereby AN is split and added to the message before and after the IFFT blocks.

Similar to [207], an AFF generation scheme for MIMO-OFDM systems is presented in [212] and [213]. The proposed AFF weight matrix is composed of two parts: a random weight matrix based on complex Gaussian random variables representing the AFF and the canceling weight matrix. The AFF is generated in such a way that it is successfully canceled out at the legitimate receiver, only.

Differently, the authors in [214] redesign AN signals in the form of constructive interference to the intended receiver and destructive signals to the eavesdropper. The AN signal will be constructive if the intended receiver moves the received symbols away from the decision thresholds of the constellation, that is the angles of both the received signal and the desired symbol should be aligned. Using the AN signals, the authors push the decision symbols towards the constructive regions of the modulation constellation. On the other hand, the eavesdropper will observe a distorted version of the transmitted signal.

Channel-based Data Confidentiality:

A mode selection scheme for MU-MIMO downlink networks is presented in [215]. Particularly, the MIMO BS broadcasts a pre-coded training vector to

a number of single antenna secure users. Then, each user calculates its received Signal-to-Interference-plus-Noise Ratio (SINR) over each beam and feedbacks the index of the beam with the highest SINR to the BS. Consequently, the BS selects the optimal user with the highest SINR for each beam. In each time slot, the BS communicates with a fixed number of users.

In [216], the CSI of the legitimate users' links and the buffer state information of the relay are leveraged to provide PLS to the considered relay system. Here, the users are assumed to be equipped with multiple antennas and a direct link between each of the communicating nodes is considered. In this scheme, either the transmitter or the relay sends useful information, while the other sends AN such that this noise will only be eliminated when passing through the intended channel. The selection of transmitting nodes depends on the CSI, more specifically, it depends on the rate between any two nodes and the buffer state information of the relay.

The authors of [217] study the security of physical layer transmission of multi-antenna beamforming with imperfect channel estimation by comparing the system transmission performance under perfect and imperfect CSI. The MIMO precoding technique used in this study is based on the channel between the legitimate users, in which the channel matrix is decomposed using the SVD technique. This technique is similar to previously discussed techniques and its weaknesses have already been highlighted.

An original symbol phase rotated secure transmission strategy to defend against eavesdroppers, is presented in [218]. The basic idea is to change the phase of original symbols, at the MIMO BS before transmitting the symbols, according to the estimated channel matrix. As such, the massive MIMO eavesdropper can only intercept the phase rotated symbols, while legitimate single-antenna users are able to recover the original symbols by reversing the phase rotation operation.

In [219], data confidentiality is achieved through random symbol rotation. The procedure works as follows: the BS transmits a reference signal to each of the single antenna users through a uniformly and randomly chosen antenna (antennas randomly assigned to each user). In turn, each user estimates the channel between itself and the corresponding antenna, which is randomly assigned to it, and sends the encrypted data after performing phase rotation based on the estimated CSI.

Differently, the concept of chaos modulation is used in [220] to achieve PLS in MIMO systems. In this technique, the modulated signals, which follow a Gaussian distribution, are generated from chaos signals correlated to transmit bits. In addition, the initial chaos signal is shared between the two communicating parties and it serves as a secret key.

Jamming and Beamforming:

A virtual MIMO-based cooperative beamforming and jamming scheme for the clustered wireless sensor networks is presented in [221]. The authors consider a wireless sensor network where the sensor nodes are divided into several clusters

each having a cluster head and some normal nodes. Moreover, each node is assumed to be equipped with a single antenna and the inter-cluster distance is assumed to be much smaller than the intra-cluster distance. The security scheme is based on cooperative beamforming and jamming and consists of two phases: in the first phase, the transmitter sends the pre-processed data to a number of cooperative nodes in its cluster, and at the same time, some of the nodes in the receiver's cluster transmit jamming signals to the receiver. In the second phase, the transmitter and the nodes send data to the receiver, which results in a virtual MISO system. Also, the jamming nodes in the receiver's cluster send again the same jamming signals to the receiver. The receiver can recover the real signal by subtracting the received signals in phases one and two. The authors assume that the eavesdropper is in a different cluster, which might not be always the case. Moreover, cooperative jamming requires a high level of synchronization between all nodes, which cannot be always attained.

The PLS scheme presented in [222] highly depends on the concept of beamforming. This scheme splits the cylindrical array, consisting of several circular arrays stacked on top of each other, into two arrays: the first is used to transmit useful information to the intended user with little leakage from the second array, which is used to transmit jamming signals to the eavesdropper. This can be achieved through beamforming in which the main beam used for sending useful information is directed towards the intended user and the array transmitting the jamming signal is directed towards the eavesdropper. It should also be noted that some of the useful information will be leaked to the eavesdropper; however, the effect of the jamming signal will completely mask the real data. The authors assume that the antennas operate in multiple bands, since it is impossible to transmit real and jamming signals simultaneously and on the same band of frequencies.

Many works in the literature are interested in the MIMO information theoretic secrecy and analysis of the achievable secrecy capacity, to provide practical security algorithms against eavesdroppers. These techniques focus on improving the received signals at the legitimate user while degrading the reception of the eavesdropper. However, to do so, the transmitter should be aware of the eavesdropper's CSI, which is impossible in many passive attack scenarios. One way to achieve secrecy is through beamforming; however, when both receiver and eavesdropper share the same conditions, the eavesdropper is able to decode and recover the data, correctly. Therefore, the authors in [223] and [224] combine the concept of beamforming with the concept of security codes to achieve a Bit Error Rate (BER) close to 0.5 at the eavesdropper (desired secrecy performance).

On another note, the PLS protocol presented in [225] utilizes the concept of jamming to secure communications between two legitimate parties. The transmitter selects two antennas that maximize and minimize the instantaneous SNR at the receiver. Then, it uses the strongest and the weakest transmit antennas to transmit valid data and jamming signals, simultaneously. Since the strongest

antenna is transmitting useful information, the useful channel capacity will be maximized while the other one will minimize the effect of jamming. At the eavesdropper's side, the strongest and weakest antennas at the receiver, correspond to random antennas for the eavesdropper, hence, the authors claim that the effect of jamming is larger at the eavesdropper's side, which in turn enhances system security. This mainly depends on the location of the eavesdropper; whenever the eavesdropper is closer to the legitimate receiver, its channel conditions will be similar to that of the receiver, hence, this technique will not be effective.

Cooperative Jamming based on Simultaneous Transmission:

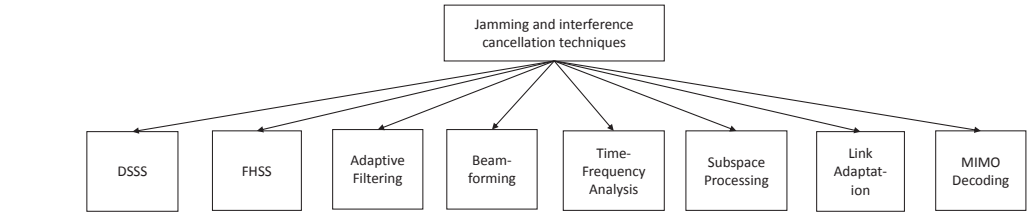
In [226, 227, 228], authors consider a relay system and rely on the idea that all users in a specific cell send traffic simultaneously, to create interference at the eavesdropper's side in the first time slot, while in the second time slot the relay broadcasts all the received signals to the users in the cell. Hence, the users will be able to cancel the inter-user interference and recover their data.

A PLS scheme for MIMO-D2D communications is presented in [229]. The physical layer network coding is divided into two stages over two-time slots. In the multiple access stage (first-time slot), two D2D devices transmit symbols simultaneously to the relay, which will prevent the eavesdropper from intercepting any of the two symbols due to the interference from the other device. During the broadcast stage (second-time slot), the relay XORs the received symbols after decoding, and sends the encrypted message to the D2D devices. Here, the eavesdropper is unable to intercept either symbols since both symbols are unknown to the eavesdropper. Moreover, in this technique, the notion of network coding is utilized to jointly increase the throughput and decrease traffic overhead.

The authors in [230], also exploit channel state information to enhance the PLS of MIMO cognitive radio networks. In this scheme, the concepts of both, energy harvesting and transmit antennas selection, are used. First, the cognitive transmitter harvests the energy from the Radio Frequency (RF) signal of the primary transmitter using all its antennas. Then, the secondary transmitter (cognitive) selects the optimal transmit antennas based on the channel between itself and the secondary receiver.

In [231] and [232], directional modulation is used to achieve data secrecy. Unlike traditional beamforming, directional modulation distorts the same signal in all directions other than the desired one. This technique assumes that the eavesdropper is not aligned with the receiver, however, in some cases the eavesdropper can be at almost the same location of the legitimate receiver. Therefore, this method is not fully secure in such cases.

In [233], the authors present a cooperative jamming scheme to improve the security of MIMO cooperative cognitive radio networks. The model includes primary and secondary transmitter-receiver pairs, a relay node and an eavesdropper. Each node is half-duplex, hence, transmission is divided into two phases. The eavesdropper can wiretap in both phases. In the first phase, primary and



Advantages	**Users share the whole bandwidth simultaneously**	**Attacker can not detect the pattern of the frequency hops since they are pseudo-random**	**Minimizes error between received signal and the real one**	**Beam steering/ no added hardware or additional resources (no added costs)**	**Provide a detailed analysis of the interfering signal**	**Using orthogonality, interference is effectively and entirely removed**	**Flexibility in various parameters such as transmit power, modulation, coding**	**Multiple copies of the same message are sent**
Limitations	**Effective against narrowband interference**	**Synchronization between users is needed**	**Ineffective when interference can not be predicated**	**Computationally exhaustive to estimate the direction of the interfering signal**	**Interfering signal should have high power**	**Information about the interfering signal is needed**	**Feedback channel is needed**	**CSI of all steams is needed**

Figure 3.7: The comparison of different anti-jamming and interference techniques

secondary transmitters transmit their signals to the relay, while the secondary receiver sends jamming signals to disrupt the received signals at the eavesdropper's side. In the second phase, the relay forwards the two signals received from the primary and secondary transmitters to the primary and secondary receivers, respectively. At the same time, the secondary transmitter transmits a jamming signal to the eavesdropper. This scheme depends on multiple antennas in which users can adjust their jamming beamformers to jam the eavesdropper and simultaneously null out the interference at legitimate receivers. However, in this case, the location of the eavesdropper should be known. In most cases, eavesdroppers remain passive, therefore, their locations and their CSI cannot be known.

The MIMO-PLS presented in [234] depends on cooperative transmission and is described as follows: two source-destination pairs, one exchanging secret information while the other is sharing a public message, transmit simultaneously. The main concept is that the message and the interference lie in different subspaces at the intended destination (interference is eliminated), however, are aligned along the same subspace at the eavesdropper, which will not be able to detect the confidential message. This scheme also depends on the location of the eavesdropper.

The MIMO PLS scheme presented in [235] depends on the concept of both cooperative jamming and AN, to secure data transmission. The legitimate user and the cooperative jammer concurrently transmit pilot signals and AN, respectively, to prevent the eavesdropper from detecting the transmitted data. The cooperative jammer introduces the AN in a spatially selective manner without affecting the data transmission of the legitimate user.

Table 3.8 summarizes and evaluates the presented data confidentiality schemes for MIMO systems.

Table 3.8: A summary of the MIMO PLS data confidentiality schemes presented in the literature

Data confidentiality for MIMO systems	**Scrambling matrix based on channel properties**	**Artificial noise (AN) and artificial fast fading (AFF)**	**Channel state information (CSI)**	**Jamming**	**Beamforming and optimal transmit antenna selection**	**Interference based on simultaneous transmission**
Advantage	Low complexity and energy consumption	Automatic cancellation of AN and AFF. No further actions are done at the receiver	Ciphertext is resilient to cryptanalysis	Data hiding	Simple and low complexity	Data hiding and traffic reduction
Limitation	Weak level of security since it is based on channel properties only	Eavesdropper can recover transmitted data if it is aligned with the legitimate receiver	CSI can be easily obtained if the eavesdropper is synchronized with users	Data distortion and data rate reduction	Alignment	Required assumption: synchronization
Resource and communication cost	No additional cost and overhead	Generation of AN and AFF matrices from the decomposition of the channel matrix	No additional resources and communication cost	Additional resources are needed to send jamming signals	No additional resources and communication cost	Additional communication cost for synchronization
Complexity	Not computationally complex: performing shuffling and permutation	1) Extract information from channel. 2) Generate AN and AFF, and 3) include them in transmitted data	Data multiplication with extracted channel information	Generate and send jamming signals	Does not require additional computations	Does not require additional computations

3.3.4 Data Availability

Traditionally, there are two popular ways to mitigate jamming attacks, which are: Direct Sequence Spread Spectrum (DSSS) and Frequency Hopping Spread Spectrum (FHSS). Both techniques fall under the same type of modulation techniques that is spread spectrum modulation. DSSS, which is similar to CDMA, multiplies a bit sequence by a (faster) chip sequence. As a result, transmitted signals spread over a wide range of bandwidth leading to the suppression of interference and jamming signals. FHSS, on the other hand, uses narrow-band signals and allows users to randomly hop from one frequency to another during transmission [236]. Beamforming also overcomes jamming and intentional interference since antenna weights are adjusted in a way that a specific beam is directed towards the intended user. However, for this technique to be successful, one should know the adversary's location. Similarly, subspace processing requires information about the interfering signal in order to design a signal subspace orthogonal to the interference. Link adaptation and adaptive filtering, on the other hand, adapt transmission parameters to decrease the error between the received signal and the original signal. However, both are not effective in the presence of unpredictable and strong interference. Time-Frequency analysis is a different technique that only reveals the power localization of the jamming signal in the time and frequency domains. Finally, MIMO decoding simultaneously transmits multiple streams which are recognized separately at the receiver. This requires knowing the CSI corresponding to all streams [237]. Figure 3.7 compares the techniques presented in [237].

Authors in [238] and [239], prove that massive MIMO is naturally resilient to standard jamming and eavesdropping attacks unless jamming is done during the training phase when pilot signals are transmitted by mobile users. To this end, the authors present a PLS scheme to secure pilot signals in the initial training phase. More specifically, the index of the pilot signals, transmitted to different users, is encrypted with a unique shared key between the base station and each of the corresponding users. This way, the eavesdropper will not be able to distinguish which pilot signal corresponds to which user and consequently, will not be able to conduct jamming attacks later on in the communication phase.

Similar to existing anti-jamming techniques, which depend on the concept of jamming the jammer, a virtual MIMO mechanism is introduced in [240]. Several single-antenna relays are grouped to form a virtual MIMO system in order to defend against jamming attacks in cognitive radio networks by nulling out the jamming signals through jamming.

On the other hand, the authors in [241] present the multi-channel ratio decoding scheme to overcome jamming attacks and recover jammed signals. This technique doesn't require shared keys or the transmitter's channel state information. Here, a multi-antenna receiver (2×2) and a single-antenna jammer are assumed. Basically, the receiver computes the multi-channel ratio by dividing

the jamming signals received by both antennas of the receiver. This ratio depends on a strong assumption that the receiver and the jammer are static, which means that the ratio channel remains constant since in this case, the CSI will not change during a short period of time. When the transmitter starts transmission, the receiver will be able to subtract the jamming signal and recover the intended messages having the multi-channel ratio.

The different types of jamming and mitigation techniques are listed in [242] and shown below:

- **Physical jamming**: Physical jamming is one kind of DoS attacks, which is realized by either continuous transmission or random bit transmission.
- **Virtual jamming**: This attack is launched at the media access control layer through attacks on the request to send and clear to send frames or the data frames.
- **Synchronization signal jamming**: Here, a user is denied access to both the primary synchronization signal and the secondary synchronization signal, which are used by a UE at the beginning of the connection establishment phase with a specific node in its cell.
- **Primary synchronization signal jamming**: This attack forges a primary synchronization signal (bogus primary synchronization signal) and tricks a UE into using this primary synchronization signal instead of the real signal.
- **Physical uplink control channel jamming**: Physical uplink control channel jamming will result in assigning additional uplink resources to every active UE that are probably not needed, which will cause degradation of service.

To prevent jamming attacks, the authors present two schemes: the rate adaptation scheme and the mapping to commitment scheme. The former allows optimal transmission mode selection based on the maximum expected throughput, while the latter depends on the notion of cryptography in which only the intended receiver is able to recover the real information conveyed by the transmitter.

Table 3.9 summarizes and compares the existing anti-jamming solutions present in the literature.

Table 3.9: A summary of the MIMO anti-jamming schemes presented in the literature

Anti-jamming scheme	**Pilot-index encryption**	**Jamming signal estimation**	**Pilot insertion within payload**	**Jamming the jammer**	**Estimating the multi-channel ratio and then subtract the jamming from the real signal**
Advantage	Low latency and complexity	No additional overhead in terms of resources and latency	Accurate estimation of the jamming signal within the jammed pilot	Prevents the jammer from sending jamming signals in real time unlike waiting for the authority's response	No additional hardware or communication are needed
Limitation	If the attacker has enough power the whole bandwidth would be affected. In such a case the attacker doesn't need to know the pilots sent to each user	Jamming signals cannot be estimated or known since these signals are random	The jammed payload cannot be recovered since the jamming signal can be fully known from the jammed pilots within the payload. Decrease in data rate	The jammer should be able to differentiate jammers from non-jammers and an extra node monitoring the network is required	Strict assumption: Receiver should be equipped with multiple antennas while the jammer should be equipped with a single antenna. The jammer is assumed to be a constant jammer
Resource and communication cost	Encryption of pilot indices and exchanging them which requires additional communication cost	No additional costs in terms of resources and communication	Using additional resources for pilot insertion in payload since pilots replace part of the intended data	Additional devices to constantly monitor the network are needed in addition to power resources to be able to send jamming signals	Receiver needs to perform additional computational operations (more power/battery)
Complexity	Not computationally complex: encrypting pilot index not the pilot signal itself	Computationally complex to estimate a jamming signal or even a part of it	Computationally complex: 1) pilot insertion upon transmission. 2) Jammed pilot extraction upon reception and comparison with original pilots	1) The friendly jammer should first detect whether unauthorized users are sending. 2) Generate jamming signals	1) The receiver should estimate the multi-channel ratio. 2) The receiver should recover the real signal

Chapter 4

Device Authentication

Device authentication is crucial for any digital system as it represents the first step towards accessing data and resources. Currently, most authentication mechanisms in the literature, are based on single-factored cryptographic solutions. These techniques are often not sufficient in the context of 5G systems due to the limited computational power of employed devices and the severity of security concerns, especially that these devices are physically not well protected. Consequently, any weakness in the identification or authentication process will allow a compromised entity to establish communication, inject false data and launch dangerous attacks leading to system malfunction.

To overcome the above-mentioned limitations and achieve high authentication accuracy, two lightweight and secure multi-factor device authentication protocols are presented. The schemes are based on two concepts, configurable Physical Unclonable Functions (PUF) within 5G devices (such as IoT devices), and channel-based parameters. Both protocols use few and simple cryptographic operations such as the bit-wise Exclusive-OR (XOR) operation and a one-way hash function. The unique PUF value serves as the mutual secret between a pair of users, which frequently changes for every session. Moreover, the proposed protocols exploit the random channel characteristics to provide high robustness against different kinds of attacks, while maintaining low complexity. Security and performance analysis prove the superiority of the proposed protocols, which are designed with minimum overhead in terms of computations, communication costs and execution time.

4.1 System Models

Before presenting the authentication protocols, the network model, the threat model, the utilized fuzzy system and the basic properties of PUFs, are briefly discussed.

4.1.1 Network Model

A 5G IoT communication system is considered, where different IoT devices (such as smartphones) are able to communicate directly with each other [243]. The system consists of several mobile devices and a fixed number of communication units (gateways). Here, it should be noted that the presence of aggregation nodes is not mandatory since in most cases, IoT devices are able to communicate with the gateway directly (for example: the LoRa (Long Range radio) technology [244]). The presented authentication protocols are not specific to 5G IoT devices; any 5G device communicating with a gateway or another 5G device can perform any of the presented mutual authentication protocols (generic protocols that can be applied by any 5G device using any multiple access technique).

The following assumptions are considered:

- 5G IoT devices communicate with the gateway directly over public wireless links (star topology).
- The gateway resides in the network server. If not, users can perform authentication with the network server, using the proposed protocols.
- The proposed solutions target single-hop networks (no aggregation nodes).
- Channel between two entities is non-reciprocal.
- The gateway has high computational power and large memory. This unit is responsible for storing unique parameters (unique identification values: X_{ID}), each corresponding to a different IoT user in the cell.
- Unique user-identification values, X_{ID}, are kept secret by the corresponding parties.
- The unique user-identification values, X_{ID}, are first generated by the gateway/network server and then relayed to each IoT device (physically). The user-identification values are saved in the memory of the gateway/network server (initial configuration). Consequently, the gateway will be able to have the X_{ID} values of all IoT devices, including new devices joining the network.

The main advantage of introducing PUFs is to eliminate the need of secret symmetric keys.

For the case of M2M communication systems (direct communication/ no gateways), devices relay the user-identification values (X_{ID}) by physically connecting to the other device or by using a secure exchange scheme (for example, using public/private keys). Afterwards, the same authentication steps are followed.

4.1.2 Threat Model

End-points are assumed to be untrustworthy nodes and the adversary is able to read, forge, manipulate, reply, delay and delete messages. Moreover, end-points communicate with the gateway (gateway residing in the network server) directly, as is the case in the LoRaWAN system [244].

In addition, it is assumed that the only way to compromise the authentication session, is by obtaining the long-term and short-term secrets. However, this is not possible since the attacker will have to guess the values of these secrets. Based on the work presented in [245], it is assumed that the underlying cryptography is perfect: each cryptographic primitive is modeled as an abstract symbolic function with strong properties [246]. For example, hash functions are irreversible (one-way) [246].

4.1.3 Fuzzy System

In the proposed protocols, the advantages of fuzzy systems are exploited to overcome the non-reciprocity of wireless channels. Generally, a fuzzy system, which is based on a collision resistant extractor, takes as input a binary string, $Seq_{(binary)}$, of some metric space $Sp \in \{0,1\}^n$ (n is a positive number) and outputs a random string $\sigma \in \{0,1\}^l$ (l is a positive number) and an auxiliary string, $\tau \in \{0,1\}^r$, where r is a positive number that can be equal to l or n [247]. This mapping procedure is denoted by:

$$Gen(Seq_{(binary)}) = (\tau, \sigma). \tag{4.1}$$

Typically, τ is a public reproduction parameter, that is known by all users. However, in the proposed schemes, this is not the case since the input to the $Gen(.)$ function is a parameter only known by communicating entities. Therefore, the resulting outputs, τ and σ, will not be public parameters.

Another procedure that is also used in fuzzy systems is the recovery function, in which a different string, $Seq'_{(binary)}$, of the same metric space $Sp \in \{0,1\}^n$ is fed, along with $\tau \in \{0,1\}^r$, to produce $\sigma \in \{0,1\}^l$ [247]. This mapping procedure is denoted by:

$$Rep(Seq'_{(binary)}, \tau) = \sigma. \tag{4.2}$$

Both of these functions are used in the proposed mutual authentication processes, which combine the secrecy of the PUF output values, and the randomness and dynamicity of wireless channels (PLS).

4.1.4 PUFs: Basic Properties and Characteristics

A PUF circuit has a specific architecture, which is typically added to a chip to extract its unique fingerprint. The input to a PUF circuit is a sequence of bits,

which is referred to as the challenge, and the output is another sequence of bits, which is referred to as the response. Each chip (5G IoT device) has its own fingerprint related to the unique pairs of challenges/responses; that is no two chips (5G IoT devices) are able to produce the same response for the same challenge [248, 249], which is mainly due to the variability within the manufacturing process.

In general, PUFs are widely used for the authentication of resource-limited devices since no cryptographic operations are required in this case. The use of PUF circuits is a very popular technique for authenticating IoT devices. Specifically, PUF-based authentication is divided into two phases: enrollment and authentication [248, 249].

In the first phase, the chip, which contains the PUF circuit, is physically linked to the server (connected). The server generates challenges, and the PUF circuit returns back the corresponding responses which are stored in the server. Next, the chip is attached to the IoT device [248, 249].

During the second phase, the server sends a dynamic random PUF challenge to the device. If the device produces and transmits the correct corresponding response, the device is authenticated [248, 249].

The enrollment and authentication steps are slightly modified in the proposed protocols. Here, it is assumed that the gateway (or network server in case the gateway does not reside in the network server) already has the PUF inputs (challenges) of each IoT device in the network.

During the enrollment phase, the chip of each IoT device is connected to the gateway (in case it resides in the network server). The gateway generates the challenge, which is a unique identification value for each device (X_{ID}), and the IoT device returns the corresponding response (ID_S). Finally, both of the device and the gateway store the challenge/response pair (X_{ID}, ID_S).

4.2 First Device Authentication Protocol

The first authentication protocol is based on two main factors, the secret session identifier (ID_S), and the secret channel-based parameter, σ [250]. The secret session identifier ID_S is derived from a PUF output value that is only known by the communicating entities (ID_S is the response of the challenge X_{ID}). More specifically, the gateway keeps a list of input PUF values (initial challenge), which are the unique identification values of each IoT user (X_{ID}). This list is private and is only accessible by the gateway. The parameter, ID_S, serves as the common shared secret. This allows the gateway to distinguish the different IoT devices.

The proposed scheme depends only on two lightweight operations, a cryptographic one-way hash function ($h(.)$) and the XOR operation. Moreover, the widely used fuzzy extractor technique is used to overcome the issue of channel non-reciprocity. In general, most PLS techniques in the literature assume that the

channel between a pair of users is reciprocal, consequently, both communicating entities extract the same channel parameters and use them for device authentication and data encryption [8]. However, this is not always true. In fact, channel characteristics and features of the same channel may differ slightly between the transmitter and receiver, and they may change from time to time. As result, a channel-based nonce extracted by User A is not always equal to a channel-based nonce extracted by User B ($N_{0,A} \neq N_{0,B}$). Common sources of channel-based nonces are the Channel State Information (CSI), the Received Signal Strength (RSS) and Angle of Arrival (AoA) [94]. For this purpose, the fuzzy extractor, which depends on two functions, $Gen(.)$ and $Rep(.)$, is used in this scheme.

- $Gen(.)$: This function is a probabilistic function that generates a uniform string of random bits given a specific input. In the proposed scheme, the input is the channel nonce extracted by User A (transmitter of authentication request) and is represented by $N_{0,A}$. The produced outputs are the l-bit channel-based key σ and the reproduction parameter τ. Hence, $Gen(N_{0,A}) = (\sigma, \tau)$.

- $Rep(.)$: This function recovers the uniform string of random bits from an input that is slightly different from the original input (Hamming Distance less or equal to a predefined threshold value th): $HD(N_{0,B}, N_{0,A}) \leq th$. In particular, $N_{0,B}$ and τ are given as inputs to produce the channel-based key, σ. Hence, $\sigma = Rep(N_{0,B}, \tau)$.

Cryptographic hash functions are employed in the proposed protocol to ensure the one-way property of exchanged messages, in addition to high input sensitivity. This prevents attackers from recovering any secret information (irreversibility) from the collected traffic. Only legitimate entities, sharing similar features and unique parameters (secret), are able to calculate the same hash digest. Hence, eavesdroppers will not be able to acquire any useful information from the transmitted messages, unless they have all of the correct parameters, which is very unlikely. This step is crucial for ensuring proper and secure authentication.

Finally, in order to guard against replay attacks, both a Time Stamp (TS) and a Random number (R) are used.

4.2.1 First Authentication Protocol and Key Agreement

This phase is executed by the IoT device (User A) and gateway (User B).

1. **Step 1:** User A first extracts a channel nonce, $N_{0,A}$, and generates (σ, τ) from $N_{0,A}$ using $Gen(N_{0,A}) = (\sigma, \tau)$. User A also generates a random number R_A. The secret session identifier, ID_S, derived from the PUF challenge, X_{ID}, is concatenated with the current time-stamp TS_A. The resultant is hashed, then XORed with R_A to generate $M_1 = h(ID_S||TS_A) \oplus$

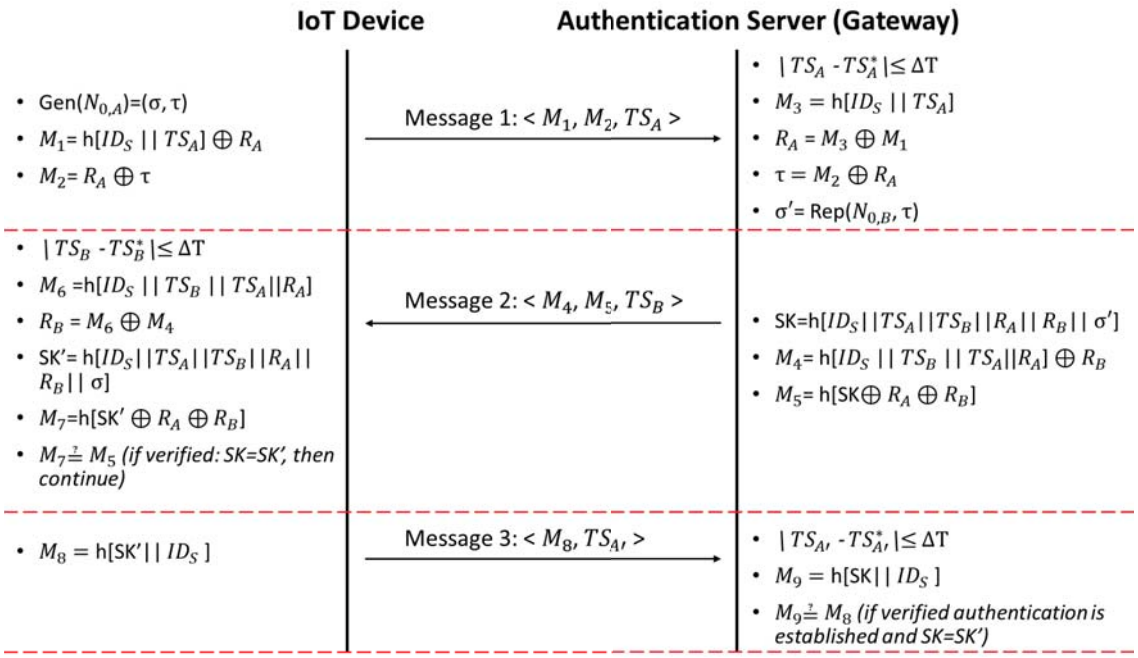


Figure 4.1: First PLS authentication protocol

R_A. A second message M_2 is calculated by XORing the random number R_A and τ, $M_2 = R_A \oplus \tau$. Finally, User A transmits the authentication request composed of $< M_1, M_2, TS_A >$, to User B over a public channel. Here, it should be noted that the reproduction parameter, τ, is used to protect and securely transmit R_A to User B. In this protocol, τ is derived from the shared physical channel between legitimate users, hence, this parameter is not publicly available and cannot be acquired by adversaries. On the other hand, σ is kept as a secret by the transmitter.

2. **Step 2:** Once the authentication request is received, User B validates the currency of TS_A using $|TS_A - TS_A^*| \leq \Delta T$, where ΔT is the maximum transmission delay, and TS_A^* is the received time of the message. If this condition fails, User B terminates the connection.

3. **Step 3:** User B calculates $M_3 = h(ID_S||TS_A)$ using the stored information ID_S and the received information TS_A. Next, User B calculates $R_A = M_1 \oplus M_3$ and $\tau = M_2 \oplus R_A$. Using τ, $\sigma' = Rep(N_{0,B}, \tau)$ is generated.

4. **Step 4:** Afterwards, User B generates $M_4 = h(ID_S||TS_A||TS_B||R_A) \oplus R_B$ using a random number R_B and the current time stamp TS_B. In addition, a secret session key SK is derived using the stored and received information,

such that $SK = h(ID_S||\sigma'||R_A||R_B||TS_A||TS_B)$. User B replies to User A using the message $< M_4, M_5, TS_B >$, where $M_5 = h(SK \oplus R_A \oplus R_B)$.

5. **Step 5:** User A receives the message $< M_4, M_5, TS_B >$ and checks the currency of the messages based on $|TS_B - TS_B^*| \leq \Delta T$. If this condition fails, User A terminates the connection.

6. **Step 6:** Then, User A extracts TS_B and generates $M_6 = h(ID_S||TS_A||R_A||TS_B)$. R_B, which is obtained by XORing M_6 with M_4, is used to derive $SK' = h(ID_S||\sigma||R_A||R_B||TS_A||TS_B)$. Using SK', R_A and R_B, User A calculates $M_7 = h(SK'||R_A||R_B)$. If $M_7 = M_5$, User A authenticates User B and verifies that both users were able to derive the same secret session key $SK = SK'$.

7. **Step 7:** User A sends a message $< M_8, TS_{A'} >$ to User B as an acknowledgment, where $M_8 = h(SK'||ID_S)$ and $TS_{A'}$ is the new time stamp.

8. **Setp 8:** Finally, User B checks $|TS_{A'}^* - TS_{A'} \leq \Delta T|$ and generates $M_9 = h(SK||ID_S)$. If $M_8 = M_9$, then User B verifies that User A has produced the same secret session key, hence, User A is authenticated. If any of the above steps fails, the connection will be immediately terminated.

At the end of this phase, both users A and B reserve the same secret session key SK, which will be used for secure communication, after performing the mutual authentication phase (Fig. 4.1).

In order to increase the robustness of the proposed protocol, an additional factor is used to enhance the mutual authentication of users. This factor employs non-cryptographic parameters such as physical channel parameters, traffic, and energy consumption. Different features (parameters) are chosen for each entity to generate a unique user profile (fingerprints). This means that device fingerprinting is generated from a set of features that can be obtained from different layers (application layer, network layer, data link layer, physical layer, ...). For example, the network traffic of each device is frequently monitored and compared to its history log (or any feature in the device's profile). Upon any change in network traffic (for example sudden increase), both devices will have to re-authenticate each other. Moreover, new user-specific credentials are used in every new authentication session. The proposed protocol mainly depends on the shared channel parameters between a pair of devices and a secret session identifier. Since wireless channels are random and dynamic, channel-based parameters will differ greatly from one time period to another. On the other hand, the secret identifier, ID_S, which is derived from the device's PUF using X_{ID}, is constant. To enhance security even further, variable challenges and responses are employed (using configurable PUFs). More specifically, input values (X_{ID}) are updated

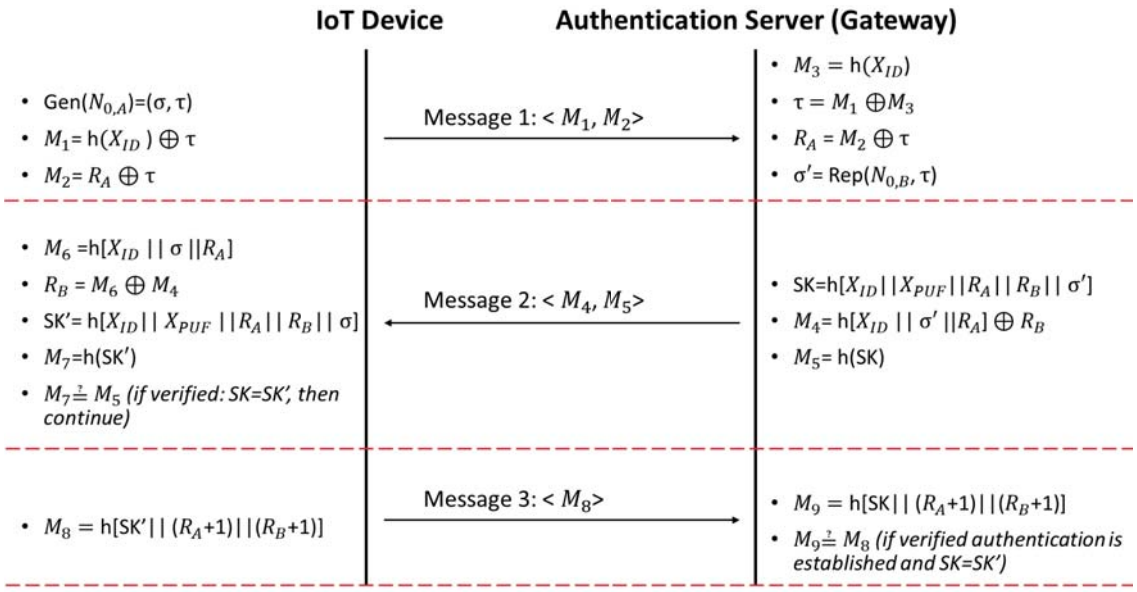


Figure 4.2: Second PLS authentication protocol

based on the previous output (recursive function). As such, the secret identifier will be dynamic and able to resist different kinds of attacks.

4.3 Second Device Authentication Protocol

The second device authentication protocol also aims at ensuring a lightweight mutual authentication process (cryptographic approach), in order to build a trusted relationship between IoT objects for access control [251]. Existing cryptographic protocols, rely on symmetric or asymmetric cryptography (public key infrastructure), which are not suitable for resource-limited 5G devices. Moreover, existing cryptographic protocols are prone to physical attacks, such as side channel and fault attacks, which are possible in the context of 5G devices since they are not physically protected [252].

Similar to the first authentication protocol, the protocol presented in this section also depends on two main factors, which are PUFs and channel-based parameters. It assumes that a gateway/server stores the unique identification value (X_{ID}) of each device. Initially, the device is physically connected to the gateway and X_{ID} is relayed. Asymmetric cryptography can also be used for relaying X_{ID}. As for the initial challenge/response of the PUF, it depends on the random channel characteristics and extracted properties. More specifically, a channel-based parameter (τ) is fed as an input to the PUF (unlike the first protocol which uses X_{ID} as input/challenge to the PUF). The challenge/response pair (τ, X_{PUF}) is stored at the IoT device and gateway. After each authentication cycle

between the IoT device and the gateway, a fresh (variable) challenge/response (new channel-derived parameters) is generated and updated in a secure and synchronized manner. The new challenge/response will be used in the next authentication cycle with the gateway. It will also be used to form the next session key and, consequently, derive the required authentication and encryption secret keys, which solves the key management issue in IoT systems, and improves the system scalability. The dynamicity of the challenge/response values is attributed to the fact that wireless channels are dynamic and random.

The main advantage of this approach is that the used structure and its parameters are dynamic in nature, and are independent of the previous and next states for every new communication session. Once mutual authentication is established, both the transmitter and the receiver can communicate securely using a shared session key (SK). The main authentication factors used in the proposed protocol are the unique identification value (X_{ID}), the secret channel-based parameter (σ) and the output PUF value, X_{PUF}. It should be noted that X_{ID} is pre-shared (using physical connection of the device and gateway as mentioned previously) and is only known by the communicating entities.

The proposed scheme depends on two lightweight operations only, the cryptographic one-way hash function and the XOR operation. Additionally, it uses the fuzzy extractor technique in order to overcome channel non-reciprocity in wireless systems.

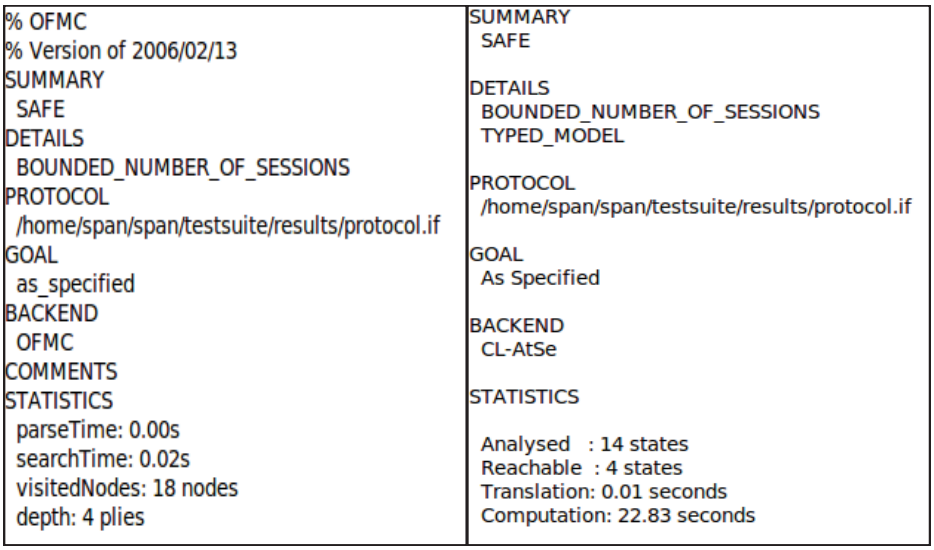
4.3.1 Second Authentication Protocol and Key Agreement

The proposed scheme consists of the following steps:

1. **Step 1:** User A first extracts a channel nonce, $N_{0,A}$, and generates (σ, τ) from $N_{0,A}$ using $Gen(N_{0,A}) = (\sigma, \tau)$. τ is fed to the PUF to produce X_{PUF}. This value is later used in the authentication process to generate the session key SK. User A also generates a random number R_A. The stored unique identification value X_{ID} is hashed, then XORed with τ to generate $M_1 = h(X_{ID}) \oplus \tau$. A second message M_2 is calculated by XORing the random number R_A and τ, $M_2 = R_A \oplus \tau$. Finally, User A transmits the authentication request, composed of $< M_1, M_2 >$, to user B over a public channel.

2. **Step 2:** Once the authentication request is received, user B calculates $M_3 = h(X_{ID})$ using the stored information X_{ID}. Next, user B calculates $\tau = M_1 \oplus M_3$ and $R_A = M_2 \oplus \tau$. Using τ, $\sigma' = Rep(N_{0,B}, \tau)$ is generated (σ=σ'). Moreover, τ is used to derive the output value, X_{PUF}.

3. **Step 3:** User B generates $M_4 = h(X_{ID}||\sigma'||R_A) \oplus R_B$ using a random number R_B, the obtained R_A, and the generated σ'. In addition, a secret session key SK is derived using the stored and received information such



(a) OFMC (b) CL-AtSe

Figure 4.3: Simulation results of the first authentication protocol using the AVISPA tool under (a) OFMC and (b) CL-AtSe backends

that $SK = h(X_{ID}||\sigma'||R_A||R_B||X_{PUF})$. User B replies to user A using the message $< M_4, M_5 >$, where $M_5 = h(SK)$.

4. **Step 4:** User A receives the message $< M_4, M_5 >$ and generates $M_6 = h(X_{ID}||R_A||\sigma)$. R_B, which is obtained by XORing M_6 with M_4, is used to derive $SK' = h(X_{ID}||\sigma||R_A||R_B||X_{PUF})$. Using SK', user A calculates $M_7 = h(SK')$. If $M_7 = M_5$, user A authenticates user B and verifies that both users were able to derive the same secret session key $SK = SK'$.

5. **Step 5:** User A sends a message $< M_8 >$ to User B as an acknowledgment, where $M_8 = h(SK'||(R_A + 1)||(R_B + 1))$.

6. **Setp 6:** Finally, User B generates $M_9 = h(SK||(R_A + 1)||(R_B + 1))$. If $M_8 = M_9$, then user B verifies that user A has produced the same secret session key and hence, user A is authenticated. If any of the above steps fails, the connection will be immediately terminated.

At the end of this phase, both users A and B reserve the same session key SK, which will be used to secure communication after performing the mutual authentication phase (Fig. 4.2). More specifically, SK is used to derive the keys that will be used for data confidentiality, source authentication and message integrity.

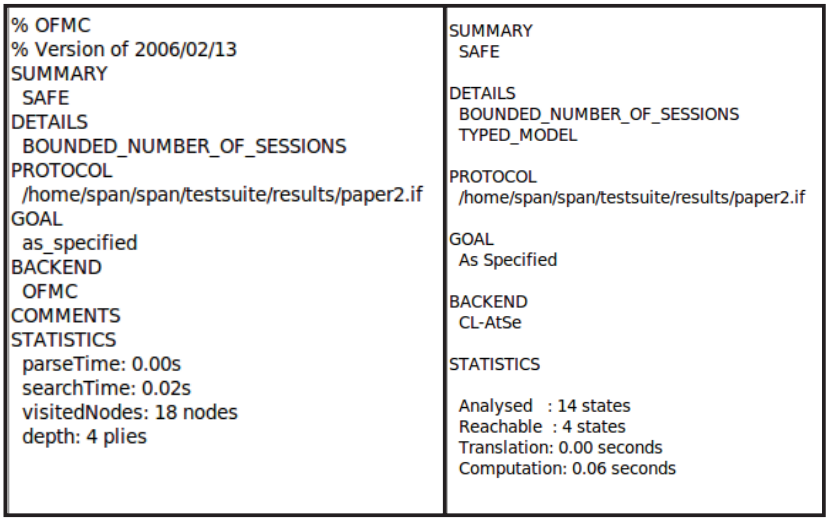


Figure 4.4: Simulation results of the second authentication protocol using the AVISPA tool under (a) OFMC and (b) CL-AtSe backends

4.4 Security Evaluation of the Presented Authentication Protocols using AVISPA

In this section, the security of the proposed authentication schemes is verfied using the widely used AVISPA tool [116, 253, 254, 255, 256, 257, 247]. The AVISPA tool includes four backends which are: 1) On-the-Fly Model Checker (OFMC), 2) Constraint-Logic-based Attack Searcher (CL-AtSe), 3) SAT-based Model Checker (SATMC) and 4) Tree Automata based on Automatic Approximation for the Analysis of Security Protocols (TA4SP). Here, the results of the proposed schemes under the SATMC and TA4SP backends are omitted since these backends don't support the XOR operation (result: "inconclusive"). The proposed schemes are implemented using the High-Level Protocol Specification Language (HLPSL), and are simulated in Security Protocol Animator for AVISPA (SPAN). It should be noted that this tool captures the replay and the Man-In-the-Middle (MIM) attacks, only.

Figures 4.3 and 4.4 prove that the proposed protocols are "safe" against replay attacks and MIM attacks, hence, both satisfy the design properties of secure authentication protocols.

4.5 Security Analysis of the Proposed Authentication Schemes

In this section, the proposed schemes are analyzed in the context of different authentication attacks.

4.5.1 Resistance Against Privacy Threats

A mutual authentication protocol is considered immune against privacy threats, if the following metrics are satisfied: indistinguishability, anonymity and identity privacy.

Indistinguishability:

In both authentication protocols, the identity-related information (X_{ID} and the PUF-derived ID_S) of users is guaranteed by the one-way property of the hash function, the randomly generated numbers and the freshness of time stamps (R_A, R_B, TS_A and TS_B). Here, the attacker cannot obtain plaintext information related to the users' identity or the transmitted data, hence, it is indistinguishable.

Anonymity:

In the presented schemes, communicating entities do not show their true identity in either the authentication stage or later in the data communication stage. In other words, data is anonymous during transmission across the public channel. Even if data is stolen, it is difficult to identify the data owner. Assuming that the adversary intercepts the exchanged messages of the first authentication protocol: $Message_1 = < M_1, M_2, TS_A >$, $Message_2 = < M_4, M_5, TS_B >$ and $Message_3 = < M_8, TS_{A'} >$. Since, TS_A, TS_B, R_A and R_B are unique and dynamic, all of the three messages are distinct where M_1, M_2, M_4, M_5 and M_8 vary greatly with any slight change in the above parameters. Similarly, assuming that the adversary intercepts the exchanged messages of the second authentication protocol: $Message_1 = < M_1, M_2 >$, $Message_2 = < M_4, M_5 >$ and $Message_3 = < M_8 >$. Since, $N_{0,A}$, $N_{0,B}$, R_A and R_B are unique and dynamic, all of the relayed messages, M_1, M_2, M_4, M_5 and M_8, are random and secured.

Identity Privacy:

One of the main security requirements for exchanging information is privacy. Using a set of secret parameters hides the real identity of communicating entities. The real identity of the IoT users is, thus, preserved and threats related to the user location tracking attacks are not possible in this case.

4.5.2 Man-In-the-Middle (MIM) Attack

The MIM attack is a form of active eavesdropping, where the attacker initiates independent connections with victims and relays messages between them. For the first authentication protocol: assume that the adversary intercepts the first message issued by the sender, $< M_1, M_2, TS_A >$. Afterwards, he creates another message using the current time-stamp TS_E and a randomly generated number R_E. The resulting message will be $< [(h(ID_S^E || TS_E)) \oplus R_E], (R_E \oplus \tau_E), TS_E >$. ID_S and τ are unknown to the adversary, consequently, another secret identifier ID_S^E and a different channel parameter $Gen(N_{0,E}) = (\sigma_E, \tau_E)$ will be used to generate M_1^E and M_2^E. In this case, connection will be terminated since both users will not be able to authenticate each other and derive the same secret session key SK. Therefore, the proposed approach is immune against MIM attacks since it relies on a secret and on channel parameters which are unknown to adversaries. The second authentication protocol is also immune against MIM attack since it is based on a pre-shared secret X_{ID}, a random number and a channel-based nonce (which is fed to the PUF) to ensure the authentication of users.

4.5.3 Resistance Against Replay Attacks

Even if the attacker was able to intercept authentication credentials and resend these credentials back to the legal entity, it is difficult to pass legal authentication due to the validity of the random numbers and time stamps. Consequently, replay attacks are easily prevented.

4.5.4 Camouflage Attack and Tracking Prevention

At the authentication stage, adversaries shouldn't acquire information related to the real user's identity or their secret credentials. For this purpose, random numbers, fresh time stamps and a one-way hash function are used, in which every new challenge is updated with a fresh time-stamp and new random number.

Moreover, the authentication mechanisms use a new channel-derived parameter ($Gen(N_0) = (\tau, \sigma)$), which makes it impossible for attackers to get the content of previous authentication sessions. This is attributed to the fact that the wireless channels are random and dynamic, hence, extracted channel parameters vary greatly from one session to another. Accordingly, the proposed protocols, effectively, resist camouflage attacks and prevent tracking.

4.5.5 Masquerading, Forgery & Impersonation Attacks

In the impersonation/masquerading attack, adversaries try to deceive users by pretending to be a legitimate sender/receiver. In the proposed schemes, all of the exchanged messages require a valid secret identifier X_{ID}, which only known

to the legitimate users. Consequently, the impersonation attack is only feasible if the adversary acquires X_{ID} (very unlikely).

4.5.6 Forward Secrecy

Forward secrecy is achieved by the one-way property of the hash function. Even if the adversary acquires the used channel-based parameters at the authentication stage, he will not be able to derive the same secret session key SK since a secret and unique identifier is utilized.

4.6 PUF-Based Threats

Since both of the presented authentication protocols utilize PUFs, it is important to assess these schemes in the context of different PUF-based attacks. In [248], authors consider two main attack models. The first model assumes that the adversary is able to intercept the communicated messages between devices (Man-In-the-Middle attack), and the second one assumes that the adversary has physical access to the device (side channel attack).

4.6.1 Man-In-the-Middle Attack

The MIM attack allows adversaries to capture messages that are communicated between two devices such as the exchanged challenges and responses. However, it has been proven earlier that the proposed protocols are immune against this type of attacks.

4.6.2 Side Channel Attack

In such an attack, the adversary has physical access to the device. The attack can be invasive, semi-invasive or non-invasive, and it can be either passive or active, according to [258].

In general, invasive (active) attacks are complicated and costly since adversaries have to move the compromised IoT device to a specialized lab, where expensive laboratory equipment are available. This type of attacks is not convenient for IoT devices, especially when devices are located in public places; hence, bringing them to a laboratory is not possible.

On the other hand, semi-invasive (active) attacks require the emission of photonics and electromagnetic sampling, and they depend on much simpler techniques compared to invasive attacks. In particular, the semi-invasive attacker should have access to the chip surface, which will not be damaged by the attack. However, this technique also requires moving the IoT device to the laboratory and utilizing special equipment.

Unlike invasive and semi-invasive attacks, the equipment needed for conducting non-invasive attacks can be transported and installed near the attacked IoT devices. These equipment are relatively small and inexpensive.

Moreover, non-invasive (passive) attacks do not require direct access to internal components. In this technique, secret information is extracted by exploiting data related to power consumption and time delay. Non-invasive attacks use analysis tools based on machine learning algorithms. Therefore, IoT devices are prone to this type of attacks. To prevent non-invasive attacks, dynamic challenge/response authentication protocols have been presented, based on physical channel parameters. The introduced dynamic physical properties prevent side channel attackers from recovering any useful information (channel parameters have a high level of randomness).

In order to reinforce the resistance against side channel attacks, proper defense strategies (restrictions on physical access) for IoT devices should be taken into account.

4.7 Security of Produced Secret Session Keys

This section provides a brief formal (mathematical) analysis to assess the security of the proposed protocols. Similar to [72, 74, 75], the Real-Or-Random (ROR) model is used to prove the robustness of the produced secret session key SK. This is important since SK will be used to ensure data confidentiality, source authentication and message integrity. Note that not all attacks are captured by mathematical modeling.

In this analysis, it is assumed that the adversary is able to eavesdrop, modify, inject, and fabricate messages using the following queries [72, 74, 75]:

- $Execute(\Lambda^v, \Lambda^w)$: This represents a passive attack, where an adversary is able to read the transmitted messages between legitimate participants at instances v and w (Λ^v and Λ^w).
- $Reveal(\Lambda^v)$: This query reveals SK to the adversary.
- $Send(\Lambda^v, M)$: This models an active attack, where an adversary sends a message M to a participant instance Λ^v, and receives a reply back.
- $Test(\Lambda^v)$: This corresponds to the security of the secret session key SK between the IoT user and gateway following the indistinguishability style in the ROR model [72]. Here, an unbiased coin is flipped before the experiment starts. Λ^v returns SK if $coin = 1$ otherwise, it returns a random number.

The adversary initiates $Test$ queries to either the IoT device or the gateway. If the guessed bit $coin'$ is equal to the random bit $coin$, the adversary wins the game

($Succ$). According to [72, 259], the adversary's advantage in breaking the security of the proposed approach and deriving SK is $Adv_{proposed} = |2.Pr[Succ] - 1|$. Using the ROR model, the proposed scheme is secure if $Adv_{proposed} \leq \epsilon$, where $\epsilon > 0$ is very small.

Theorem 1. *The secret session key SK is secure against adversaries. Using the ROR model, $Adv_{proposed} \leq \frac{q_h^2}{|H_a|}$, where q_h and $|H_a|$ are the number of access times to a collision-resistant hash function $h(.)$ and the range of space of a hash function $h(.)$, respectively.*

Proof. The approaches in [72, 74, 75] are modified, where three games, $Game_i (i = 0, 1, 2)$ are defined.

- **Game$_0$:** This represents the original attack on the protocol using a random bit test. Since *coin* should be guessed by the adversary before the game starts, by definition:

$$Adv_{proposed} = |2Pr[Succ_0] - 1|. \tag{4.3}$$

- **Game$_1$:** $Game_0$ is transformed to $Game_1$. Here, the adversary intercepts (eavesdropping) the transmitted messages between the sender and receiver ($Execute$ query). The adversary uses the $Test$ and $Reveal$ queries to test whether the $Test$ query gives the real value of SK. Since the secret session key contains short and long-term secrets, the adversary's chance of winning this game is not increased by eavesdropping the exchanged messages. Hence, it is clear that:

$$Pr[Succ_0] = Pr[Succ_1]. \tag{4.4}$$

- **Game$_2$:** $Game_1$ is transformed to $Game_2$, which is an active attack. The adversary performs several $Send$ queries in order to guess the output of the hash functions of the transmitted messages. However, these messages also include long and short-term secrets. As a result, this will lead to no collision which gives the following:

$$Pr[Succ_1] \leq \frac{q_h^2}{2|H_a|} + Pr[Succ_2]. \tag{4.5}$$

  Since the adversary has no choice other than guessing the bit *coin* in order to win the game:

$$Pr[Succ_2] = \frac{1}{2}. \tag{4.6}$$

From Equations (4.5) and (4.6), it follows that:

$$Pr[Succ_1] \leq \frac{q_h^2}{2|H_a|} + \frac{1}{2}, \tag{4.7}$$

$$Pr[Succ_0] \leq \frac{q_h^2}{2|H_a|} + \frac{1}{2}. \tag{4.8}$$

Using Equation (4.3):

$$Adv_{proposed} \leq |2[\frac{q_h^2}{2|H_a|} + \frac{1}{2}] - 1|, \tag{4.9}$$

$$Adv_{proposed} \leq \frac{q_h^2}{|H_a|}. \tag{4.10}$$

Since the range of space of a hash function $|H_a|$ is much greater than the number of $Test$ queries, $\frac{q_h^2}{|H_a|}$ is negligible. Consequently, $Adv_{proposed} \leq \epsilon$, which proves that SK and data transmitted using the proposed scheme are secure. For a detailed discussion, refer to [72, 74, 75]. □

4.8 Performance Analysis

In this section, the performance of the presented protocols is evaluated in comparison to similar protocols presented in the literature [72], [73], [74], and [75]. The tested parameters include communication cost, computational cost and execution time. Although, the protocols in [72], [73], [74], and [75] do not utilize the notion of PLS, they use PUFs in the authentication process. In contrast, the proposed schemes benefit from both PUFs and PLS, to increase the robustness and efficiency of 5G IoT systems.

4.8.1 Communication Costs

For comparative purposes, the unique identification value and the secret session identifier are both set to 160 bits, the random number is also 160 bits, the time stamp is 32 bits, and the hash digest is 160 bits (using the SHA-1 hash function as in [72]).

The protocol presented in [72] requires the exchange of three messages, which consist of 512, 512, 192 bits, respectively. Consequently, the total communication cost is $1,216$ bits. On the other hand, the total communication cost of the scheme presented in [73] is $1,856$ bits, where four messages are needed to achieve mutual authentication. Similarly, the authentication protocols in [74] and [75] require four messages as a communication overhead, and a total of $2,752$ bits and $2,080$ bits, respectively.

The first authentication protocol involves the exchange of three messages: 1) $< M_1, M_2, TS_A >$, 2) $< M_4, M_5, TS_B >$ and 3) $< M_8, TS_{A'} >$, however, it requires a fewer number of bits. The first message consists of $(160 + 160 + 32) = 352$ bits. The second message is also composed of $(160 + 160 + 32) = 352$ bits whereas the final message requires $(160 + 32) = 192$ bits only. Hence, the total number of required bits is $(352 + 352 + 192) = 896$ bits, which less than $1,216$ bits [72]. In other words, the proposed scheme is more efficient than the scheme presented in [72] in terms of communication cost (Table 4.1).

Three messages are also required to achieve mutual authentication using the second authentication protocol, 1) $< M_1, M_2 >$, 2) $< M_4, M_5 >$ and 3) $< M_8 >$. The first message requires $(160 + 160) = 320$ bits. The second message is also composed of $(160 + 160) = 320$ bits, whereas the final message requires 160 bits only. The total number of required bits is $(320 + 320 + 160) = 800$ bits, which is less than the number of bits required for the schemes in [72, 74, 75, 73].

Table 4.1: Communication cost

Scheme	**Required messages**	**Required bits**
Protocol in [72]	3	$1,216$
Protocol in [73]	4	$1,856$
Protocol in [74]	4	$2,752$
Protocol in [75]	4	$2,080$
First protocol	3	896
Second protocol	3	800

4.8.2 Computational Cost

In order to asses the computational costs of the proposed schemes, the following parameters are identified: T_h, T_{xor}, T_f and T_E, which denote the time of the hash function, the time of the XOR operation, the time of the fuzzy extractor, and the time of the elliptic curve cryptosystem point multiplication, respectively. The

total computational delay of the schemes presented in [72], [73], [74] and [75] are:

$$Delay_{[72]} = 17T_h + 8T_{xor} + 1T_f, \tag{4.11}$$
$$Delay_{[73]} = 21T_h + 3T_E, \tag{4.12}$$
$$Delay_{[74]} = 31T_h + 4T_E + 1T_f, \tag{4.13}$$
$$Delay_{[75]} = 19T_h + 7T_{xor}. \tag{4.14}$$

On the other hand, the total computational delay of the first and second protocols is:

$$Delay_{first} = 10T_h + 10T_{xor} + 1T_f. \tag{4.15}$$
$$Delay_{second} = 10T_h + 6T_{xor} + 1T_f. \tag{4.16}$$

Since the time required by the XOR operation is much less than that of the hash operation (negligible), one can conclude that the proposed schemes outperform the schemes in [72], [73], [74] and [75]. In particular, the first and second authentication protocols perform 10 hash operations, only, which is less than the number of hash operations in the previously listed schemes (17, 21, 31 and 19 hash operations). Table 4.2 summaries the computational cost in terms of delay.

Table 4.2: Computational cost

Scheme	**Computational delay**
Protocol in [72]	$17T_h + 8T_{xor} + 1T_f$
Protocol in [73]	$21T_h + 3T_E$
Protocol in [74]	$31T_h + 4T_E + 1T_f$
Protocol in [75]	$19T_h + 7T_{xor}$
First protocol	$10T_h + 10T_{xor} + 1T_f$
Second protocol	$10T_h + 6T_{xor} + 1T_f$

4.8.3 Execution Time

In order to evaluate the execution time of the proposed authentication protocols, "OpenSSL" is used. It is a very popular tool and is widely used since it is considered as one of the most important and efficient cryptographic libraries that

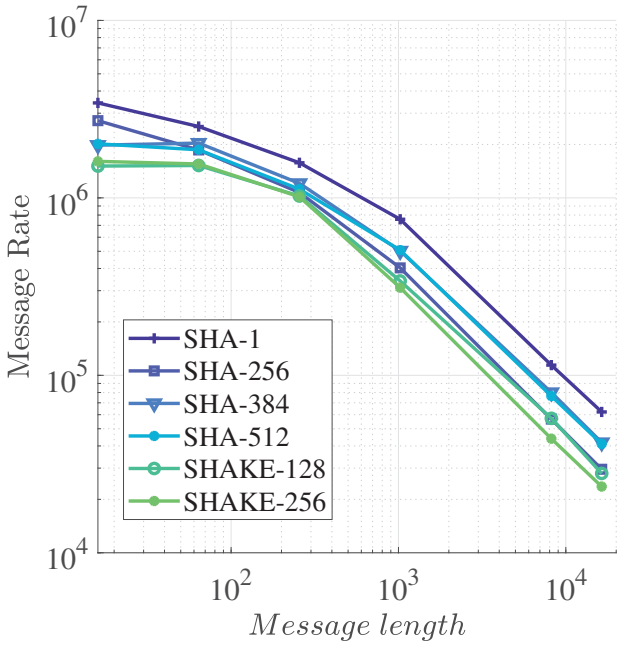


Figure 4.5: The number of hashed messages in one second using different hash functions, versus message length

provide robust, commercial-grade, and full-featured toolkit for Transport Layer Security (TLS) and Secure Sockets Layer (SSL) protocols. Moreover, OpenSSL is implemented on a common IoT hardware which is Raspberry Pi 2, which has a Broadcom BCM2836 SoC with a 900 MHz 32-bit quad-core ARM Cortex-A7 processor. To show the efficiency of the proposed authentication protocols, the time required by different hash functions is computed, mainly the SHA-256 and SHA-512 functions (Table 4.3 and Fig. 4.5). It is important to assess the delay introduced by hash functions since, among all of the listed operations, hash functions require the largest execution time (the execution time of other operations is considered negligible in comparison to hash functions).

For different message sizes (block size in bytes), it is evident that the SHA-512 requires less time, hence, it is more efficient. In contrast, the SHA-256 is efficient for small-size messages only (16 bytes).

Considering the SHA-512 scheme and a 256-byte message, the total execution time that is required by the tested authentication schemes is shown in Table 4.4. The obtained results prove that both of the presented mutual authentication schemes are more efficient than similar schemes in the literature. Since the first and second schemes require the same number of hash functions, their total execution time values are equal. Moreover, both protocols reduce the introduced delay, significantly, where a reduction of 41.17%, 52.38%, 67.74% and 47.36% are achieved with respect to the protocols presented in [72], [73], [74] and [75],

Table 4.3: Execution time (sec) of the SHA-256 and SHA-512 hash functions

Type	**16 bytes**	**64 bytes**	**256 bytes**	**1024 bytes**
SHA-256	2.1209e-07	3.7359e-07	7.7219e-07	2.3952e-06
SHA-512	2.9697e-07	2.7289e-07	6.3527e-07	1.7382e-06

respectively.

Table 4.4: The total execution time of the tested schemes using SHA-512 and a 256-byte message

Scheme	**Total execution time**	**Percentage reduction using the first protocol**	**Percentage reduction using the second protocol**
Protocol in [72]	1.0799e-05	41.17%	41.17%
Protocol in [73]	1.3340e-05	52.38%	52.38%
Protocol in [74]	1.969e-05	67.74%	67.74%
Protocol in [75]	1.2070e-05	47.36%	47.36%
First protocol	6.3527e-06	-	-
Second protocol	6.3527e-06	-	-

Chapter 5

Key Generation

The presented dynamic key generation scheme is considered as a cryptographic primitive and a basis for realizing any of the PLS data confidentiality techniques in the literature [260, 182, 261]. In order to achieve robust data secrecy, encryption and confidentiality schemes should depend on a unique, pseudo-random and dynamic secret key between two legitimate users, and not only on the physical channel characteristics. This key can also be utilized to achieve other security services such as source authentication and message integrity, and data availability.

Figure 5.1 illustrates the key derivation function, which takes as input a secret session key SK and a nonce N_0 (or σ in case the channel between users is not reciprocal). These parameters are updated every new session [260, 182, 261].

- **Secret session key** SK: This secret session key is exchanged between communicating entities during the mutual (device) authentication step (discussed in the previous chapter).

- **Nonce** N_0: This nonce is extracted from the shared channel parameters between the legitimate users. For each new session, a new nonce is generated (wireless channels are dynamic; they change frequently). When the channel is reciprocal, both the transmitter and receiver are able to extract the same nonce, separately ($N_0 = N_{0,A} = N_{0,B}$). In case of channel non-reciprocity, users can utilize σ instead of N_0, which is derived from the fuzzy extractor functions ($Gen(\cdot)$ and $Rep(\cdot)$) using $N_{0,A}$, $N_{0,B}$ and τ. In particular, the transmitter (user A) generates τ and σ using its extracted nonce: $Gen(N_{0,A}) = (\tau, \sigma)$. On the other hand, the receiver (user B) generates σ' using its own nonce and τ: $Rep(N_{0,B}, \tau) = \sigma'$. Whenever the hamming distance between $N_{0,A}$ and $N_{0,B}$ is less than a pre-defined threshold (slightly different), σ' will be equal to σ. Here, it should be noted that τ is relayed securely from the transmitter to the receiver as discussed in the previous chapter.

The obtained SK and N_0 (or σ) are XOR-ed and then hashed (using SHA-512), to produce the dynamic key DK, which is composed of 512 bits. Hashing is

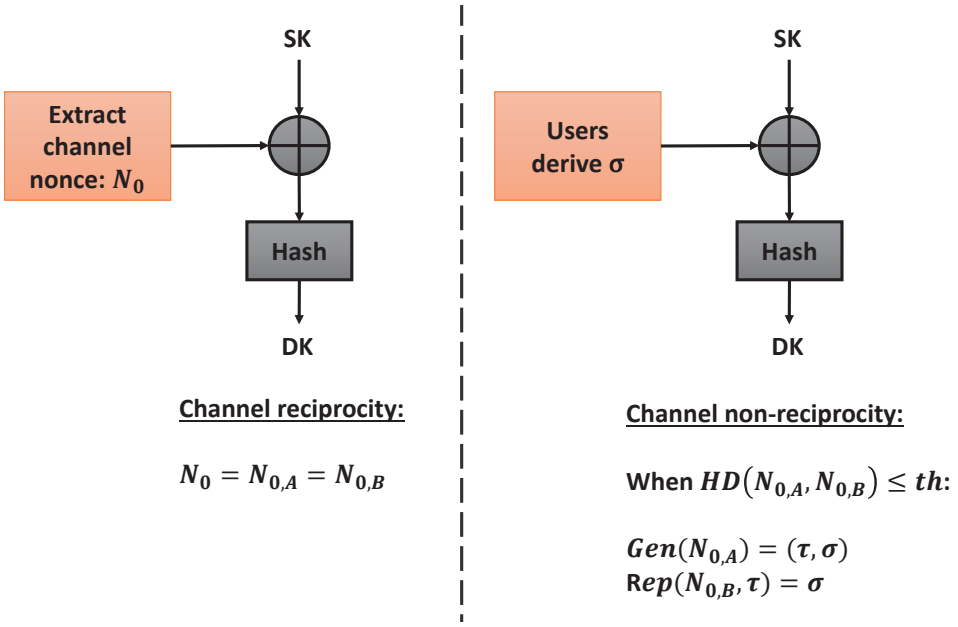


Figure 5.1: The key generation function

utilized to ensure the irreversibility property of DK. In case the two parameters don't have the same lengths, zero padding is applied to the shorter sequence in order to match the length of the other parameter. The dynamic key is sensitive to any change that occurs in the channel, and hence, its dynamic property guarantees a high level of security since it is directly related to a secret key between two users. The proposed key approach is generic and can be used to generate any cipher primitive.

In order to realize several security services, such as data confidentiality, source authentication and message integrity and data availability, the obtained dynamic key is divided into smaller sub-keys, each used to generate a specific cipher primitive (for example: permutation table, substitution table, ...). DK, being 512 bits long (64 bytes), is split into multiple different sub-keys, each having a different length. The generation of the sub-keys will be further discussed in the following chapters, since for each security service a different number of sub-keys is derived from DK.

Chapter 6

Data Confidentiality for OFDM-based IoT Systems

After generating the channel-based dynamic key (DK), data confidentiality can be achieved. For consistency purposes, the privacy of data in IoT systems is addressed, however, the presented schemes can be applied for any 5G technology, utilizing any multiple access scheme. In particular, four data confidentiality schemes are presented, each targeting a different type of IoT systems: an OFDM-based IoT system [183], a general IoT system utilizing any multiple access scheme (general scheme) [261, 182], a NOMA-based IoT system and a MIMO-based IoT system. Each case is discussed in a separate chapter.

OFDM is the basic building block for multi-carrier modulation in most contemporary networks such as Vehicular Ad Hoc Networks (VANETs), Internet of things (IoTs), as well as 4G/5G systems. Most existing OFDM-based security solutions lack the notion of secrecy and dynamicity when combining a secret key with random information extracted from the physical channel. Yet, some solutions perform encryption Pre-IFFT (Inverse Fast Fourier Transform) and some Post-IFFT, without clear guidelines concerning the impact on performance and security. In this chapter, OFDM-based encryption schemes at the physical layer are investigated, analyzed, and weaknesses are identified. It is shown that encryption in the frequency-domain slightly mitigates the effects of channel fading and improves the bit error-rate performance. On the other hand, time-domain encryption is shown to be more secure. Furthermore, a dynamic secret key approach that enhances the security level of OFDM-based encryption schemes, in addition to a new technique for updating cipher primitives for input OFDM symbols or frames, are proposed. These schemes are shown to strike a good balance between performance and security robustness as demonstrated through experimental simulations.

6.1 Proposed 2-D Permutation Scheme

Algorithm 1 describes the proposed 2-D permutation scheme, which is a variant of the traditional permutation scheme. First, the real and imaginary parts of each complex symbol in one OFDM frame symbol, (before or after the IFFT transformation: time-domain or frequency-domain complex symbols) are split, and saved in a temporary vector. The odd indices of this vector contain the real components of the complex symbols (CS), and the even indices contain the imaginary components. Then, pseudo-random permutation is performed based on the permutation table $Pbox$ ($Pbox$ is twice the size of one OFDM frame symbol, that is $2 \times NB_{FS}$, since each complex symbol in one frame symbol is split into real and imaginary). As such, the real and imaginary parts of the complex symbols are completely shuffled. Finally, a new complex-valued vector is constructed from the permuted vector such that the odd and even indices of the permuted vector represent the new real and imaginary components of the complex symbols in the new vector, respectively. To improve the security level even further, $Pbox$ is randomly shuffled/updated using the channel-based dynamic key for each new frame symbol.

By dynamically changing the permutation table using the obtained dynamic key, the proposed method becomes very effective and robust against several attacks. This approach, on the other hand, introduces additional overhead in terms of latency and resources since the size of the permutation table is doubled.

Algorithm 1 Proposed 2D-permutation cipher scheme

1: **procedure** 2D_PERM_ENCR(CS, $Pbox$)
2: **for** $i = 1$ to NB_{FS} **do**
3: $vec[2i-1] = \mathrm{Re}(CS[i])$
4: $vec[2i] = \mathrm{Im}(CS[i])$
5: **end for**
6: **for** $i = 1$ to $2 \times NB_{FS}$ **do**
7: $temp[i] = vec[Pbox[i]]$
8: **end for**
9: **for** $i = 1$ to NB_{FS} **do**
10: $vec_{new}[i] = temp[2i-1] + j \times temp[2i]$
11: **end for**
12: **return** vec_{new}
13: **end procedure**

6.2 Enhanced Phase Encryption Scheme

In order to enhance the security of the scheme presented in [110], a simple swapping operation is introduced, based on the sequence Seq_3 (size equal to NB_{FS}, which is the number of elements in one frame symbol). Whenever Seq_3 is equal to -1, the real and imaginary components of the complex time-domain or frequency-domain symbols are swapped and then multiplied by sequences Seq_1 and Seq_2 (Algorithm 2). Both sequences, Seq_1 and Seq_2, have random values of 1 and -1 (each has a length equal to the number of complex elements in one OFDM frame symbol, NB_{FS}). Such an enhancement has proven to greatly improve the security level of the scheme presented in [110]. The decryption process is also a two-step process similar to encryption but with a reversed order.

This solution reduces the required key-stream length in comparison to the traditional cipher stream technique for symbols having a length more than 3 bits.

Algorithm 2 Proposed enhanced phase encryption algorithm

procedure ENHANCED_PHASE(CS, Seq_1, Seq_2, Seq_3, NB_{FS})
  for $i = 1$ to NB_{FS} **do**
    $R[i] = \mathrm{Re}(CS[i])$
    $I[i] = \mathrm{Im}(CS[i])$
    if $Seq_3[i] == -1$ **then**
      $temp = R[i]$
      $R[i] = I[i]$
      $I[i] = temp$
    end if
    $R[i] = R[i] \times Seq_1[i]$
    $I[i] = I[i] \times Seq_2[i]$
    $vec_{new}[i] = R[i] + j \times I[i]$
  end for
  return vec_{new}
end procedure

6.3 Sub-key Generation and Encryption Model

The dynamic key, DK (512 bits), can be divided into two main sub-keys DSK_{cipher} and DSK_{Pbox_U}, each having a length of 256 bits (Fig. 6.1).

- **Sub-key** DSK_{cipher}: It consists of the most significant 32 bytes of DK and it is used to construct the cipher primitives of any cipher scheme such as the permutation scheme and the enhanced phase encryption scheme ($Pbox$,

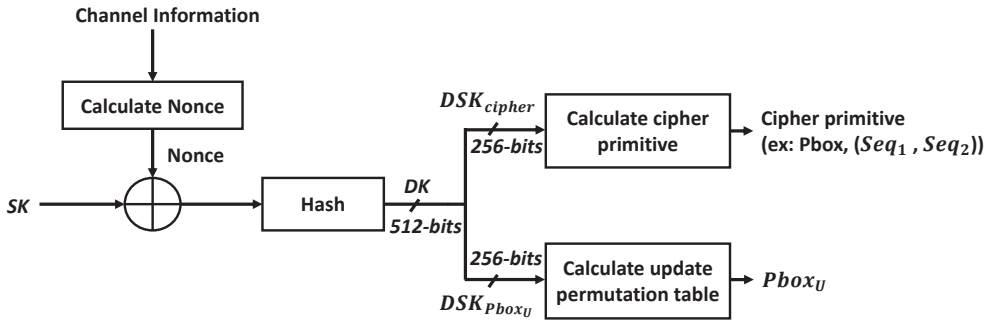


Figure 6.1: Proposed dynamic sub-key generation scheme for OFDM systems

Seq_1, Seq_2 and Seq_3). The size of the cipher primitive depends on the size of one OFDM frame symbol, that is NB_{FS}. Here, encryption is done at the frame symbol level, where one frame symbol consists of several complex modulation symbols.

- **Sub-key** DSK_{Pbox_U}: It consists of the least significant 32 bytes of DK and it is used to produce a dynamic key-dependent update permutation table ($Pbox_U$), which allows the dynamic permutation of cipher primitives for each new input OFDM frame or frame symbol. The length of the $Pbox_U$ depends on the used cipher primitive (usually the size of $Pbox_U$ is equal to the size of cipher primitive). For example, the $Pbox_U$ has a length of NB_{FS}, $2\times NB_{FS}$, $2\times NB_{FS}$, and $3\times NB_{FS}$ for the traditional permutation scheme, the 2-D permutation scheme, the phase encryption scheme [110], and the enhanced phase encryption scheme, respectively (encryption is realized at the frame symbol level).

For any change in the secret key or the nonce, a new dynamic key and a new set of sub-keys will be generated. This will result in a completely different set of encrypted OFDM symbols.

The proposed scheme is based on a dynamic key approach, which can be used by two entities sharing the same channel in wireless link-to-link or end-to-end communication. The motivation behind this approach is to benefit from channel characteristics and extract a nonce, which is combined with a secret key to produce a dynamic secret key. Accordingly, cipher primitives and dynamic permutation tables are generated (according to the selected cipher scheme) leading to enhanced security levels. The update permutation tables are used to permute the cipher primitives after each new input OFDM frame or symbol, making the cryptanalysis task unfeasible (Fig. 6.2). In other words, the permutation operation is introduced to pseudo-randomize the cipher primitive. This operation is robust against possible future attacks and consequently, ensures a higher security level compared to the existing cipher approaches that use static cipher

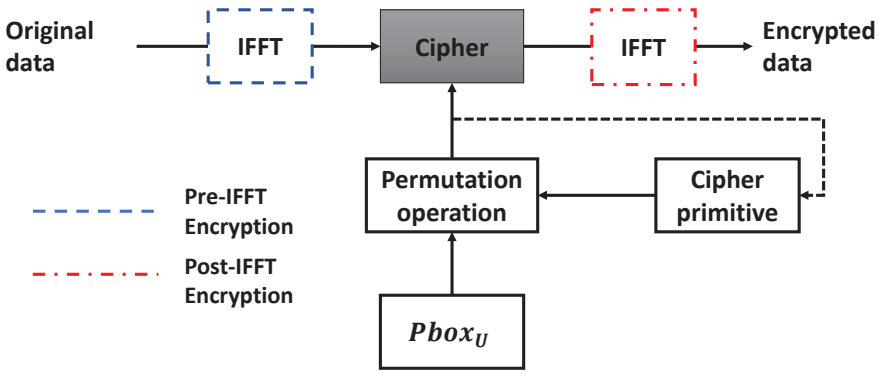
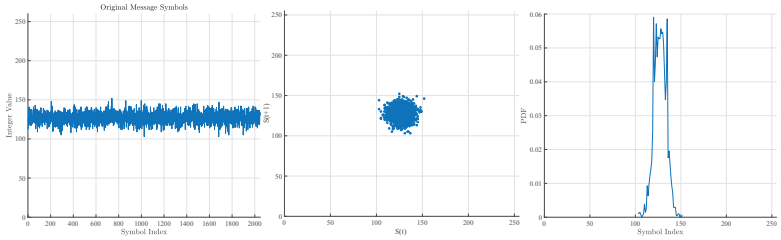


Figure 6.2: Proposed dynamic key-dependent OFDM cipher scheme

layers. The proposed update process introduces a negligible latency overhead (only permutation process), and does not degrade performance.

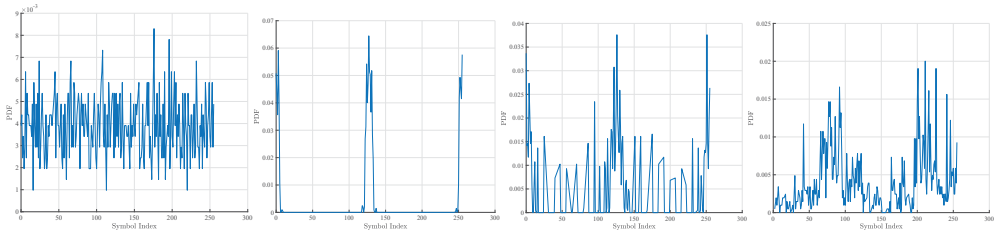


(a) Symbol Amplitude (b) Recurrence (c) PDF

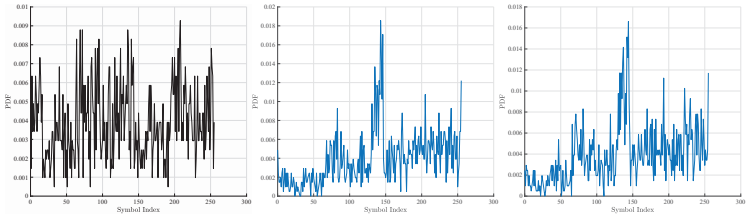
Figure 6.3: Signal values, recurrence, and PDF of the chosen original message having a normal distribution

6.4 A Comparative Security Study of Different Cipher Schemes in OFDM Systems: Pre-IFFT versus Post-IFFT

In this section, the security level of the proposed cipher schemes is studied and compared to some of the well-known encryption schemes in the literature. These schemes are tested under two scenarios: Pre-IFFT and Post-IFFT. It is assumed that the adversary knows the characteristics of the channel and the protocols used for transmission, and is able to intercept the encrypted messages exchanged between the transmitter and receiver. Various techniques and attacks are used to recover key streams. Hence, the security level of the enhanced phase encryption

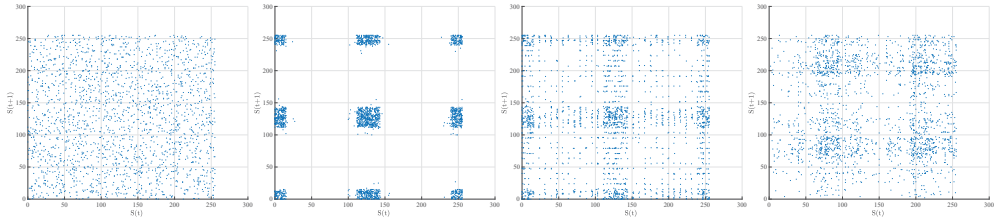


(a) XOR, Pre-IFFT (b) "AB", Pre-IFFT (c) "ABC", Pre-IFFT (d) PAPR, Pre-IFFT

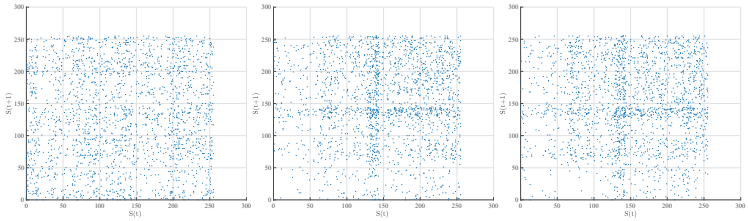


(e) Permutation, Post-IFFT (f) "AB", Post-IFFT (g) "ABC", Post-IFFT

Figure 6.4: Probability density functions of the various tested encryption schemes



(a) XOR, Pre-IFFT (b) "AB", Pre-IFFT (c) "ABC", Pre-IFFT (d) PAPR, Pre-IFFT



(e) Permutation, Post-IFFT (f) "AB", Post-IFFT (g) "ABC", Post-IFFT

Figure 6.5: Recurrence plots of the various tested encryption schemes

scheme is analyzed to demonstrate its robustness against several attacks. For simplicity, the phase encryption scheme will be referred to as the "AB" scheme in the following figures and legends, and the proposed enhanced version of this scheme will be referred to as the "ABC" scheme. It should be noted that the security level of the proposed 2-D permutation scheme is similar to that of the

traditional permutation scheme, consequently, the test results of the former are omitted.

6.4.1 Uniformity and Independence of OFDM Symbols

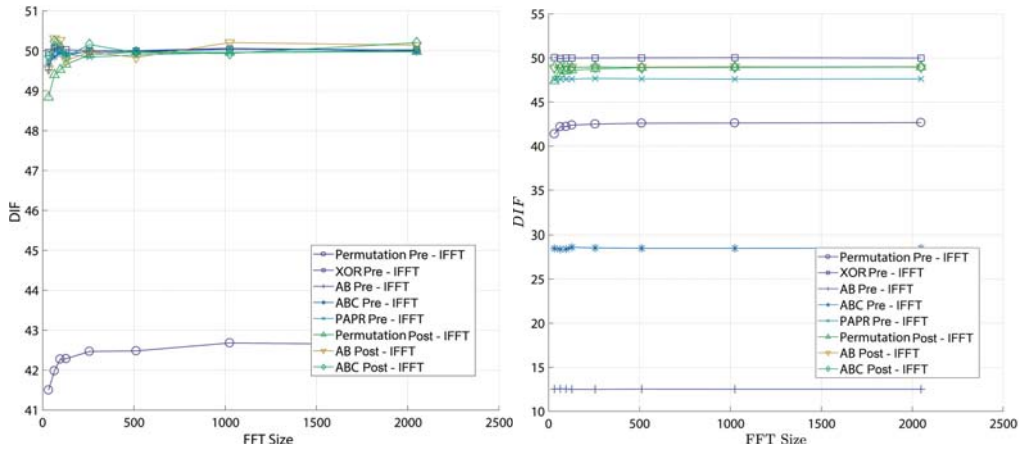
A secure OFDM encryption algorithm should ensure high levels of 1) uniformity, 2) recurrence, and 3) independence [17].

Uniformity: To evaluate the uniformity of a specific cipher scheme, the Probability Density Function (PDF) is plotted. In Figures 6.3c and 6.4, the PDF of 1,000 original OFDM symbols and their encrypted versions (using different encryption schemes) are shown, respectively. In principle, having a PDF close to a uniform distribution testifies that the data has a good level of mixing and that the encrypted OFDM symbols are spread over the entire space.

As shown in Fig. 6.3c, the original data has a normal distribution. Figures 6.4a and 6.4e show that the stream cipher XOR scheme (Pre-IFFT) and the permutation scheme (Post-IFFT) have PDFs very close to a uniform distribution. The "AB" scheme, however, performs poorly in the frequency-domain (Fig. 6.4b) but has a better distribution for time-domain encryption (Fig. 6.4f). The proposed cipher ("ABC") scheme, on the other hand, has a better uniform distribution in both domains compared to the "AB" scheme (Figures 6.4c and 6.4g). Additionally, the proposed "ABC" scheme in the frequency-domain (Fig 6.4c), has a PDF closer to a uniform distribution than the PAPR reduction scheme, which is also applied before the IFFT transformation (Fig. 6.4d).

Recurrence Test: A good cipher scheme should reach a high level of randomness within the OFDM cipher symbol space. In other words, the recurrence plot for a certain encryption algorithm should be scattered as much as possible to be considered as a 'good' encryption candidate. In order to evaluate the level of randomness of the proposed encryption algorithm ("ABC"), the recurrence test is used to measure the randomness and estimate the correlation among the data by considering the variation between the received demodulated stream of bytes $Seq_{byte}(t)$ (encrypted), and a delayed version of it for $t \geq 1$ given by $Seq_{byte}(t+1)$. For comparison purposes, the recurrence plot of the original message is presented in Fig. 6.3b, in which all points are grouped within one region (not randomized).

Figure 6.5 shows the correlation (ρ) between $Seq_{byte}(t)$ and $Seq_{byte}(t+1)$ for different encryption schemes. The ciphertext space of the XOR scheme (Fig. 6.5a) as well as the Pre-IFFT permutation scheme (Fig. 6.5e) exhibit the highest levels of randomness compared to other schemes, and no clear pattern can be clearly discerned after the encryption process. In contrast, recurrence plots corresponding to the "AB" scheme and the proposed "ABC" cipher variant Pre-IFFT block are less scattered, as shown in Figures 6.5b and 6.5c. These schemes exhibit more random recurrence plots when performing encryption Post-IFFT, which



(a) QPSK (b) 256-QAM

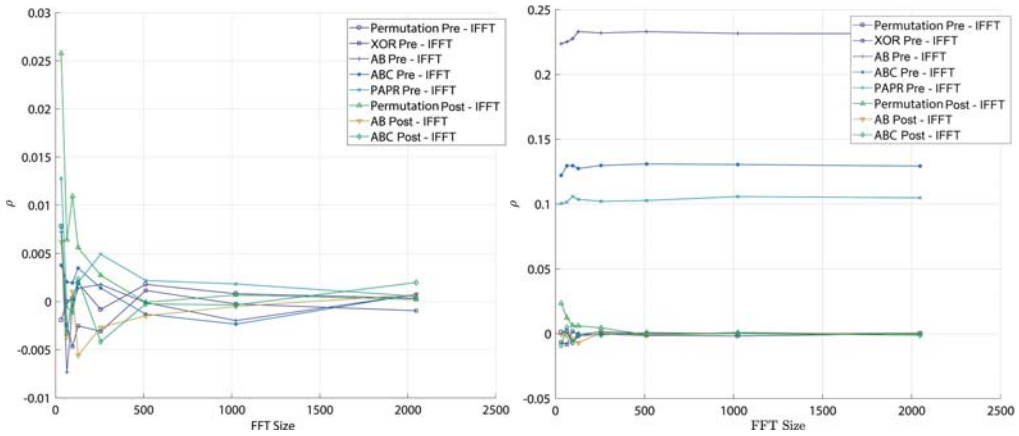
Figure 6.6: Percentage of bits changed between original and encrypted OFDM symbols for (a) QPSK, and (b) 256-QAM

validates the significance of time-domain encryption in enhancing the security level (Figures 6.5f and 6.5g) and the importance of the IFFT diffusion property in increasing symbol randomness. Finally, the PAPR reduction scheme has a randomness level similar to that of the "ABC" scheme when applied before the IFFT transformation (Fig. 6.5d).

Independence: In order to ensure a secure OFDM encryption technique, the difference in bits between the encrypted and the original OFDM symbols should be close to 50%, and the cross-correlation of the data itself should be as low as possible (close to 0).

Figure 6.6 plots the difference in terms of percentage of bits changed between the original and encrypted OFDM symbols, assuming QPSK and 256-QAM symbol modulation. Figure 6.7 plots the corresponding correlation coefficients for 1,000 OFDM symbols. Here, the correlation between the received demodulated ciphertext (in bytes) and the original data (bytes) is considered, where the correlation coefficient ranges between -1 and 1.

As depicted in Fig. 6.6a, all encryption schemes expect for the Pre-IFFT permutation scheme, have a difference value close to the desired value of 50%. This indicates that these schemes are considered good encryption candidates when using QPSK modulation. On the contrary, Fig. 6.6b shows that only the encryption schemes performed Post-IFFT reach a difference value equal to 50% using 256-QAM, expect for the PAPR reduction scheme (Pre-IFFT) which has a difference value equal to 48%. On the other hand, the permutation, "ABC", and "AB" schemes all have difference values below 50% (43.5%, 28% and 13% respectively), when applied in the frequency-domain. Moreover, it should be



(a) QPSK (b) 256-QAM

Figure 6.7: Correlation coefficients between original and encrypted OFDM symbols using (a) QPSK, and (b) 256-QAM

noted that for higher order modulation schemes, the proposed variant "ABC" shows a better difference value (28%) compared to the "AB" scheme (13%) in the frequency domain.

Figure 6.7a shows that the correlation coefficient between the original and encrypted OFDM symbols for all encryption schemes is close to the desired value of 0, using QPSK. However, when using 256-QAM (Fig. 6.7b), slight variations in the correlation coefficient occur whereby schemes performed Post-IFFT remain around 0, while those applied Pre-IFFT take values between 0.1 and 0.24. Again, the proposed variant "ABC" exhibits a better performance compared to the "AB" scheme in the frequency domain, achieving values around 0.13 and 0.24, respectively.

It is also evident from Figures 6.6 and 6.7 that the performance of some encryption schemes, especially those performed Post-IFFT block, degrades when using higher-order modulation schemes. For example, the difference value of the "AB" scheme decreases from 50% when using QPSK to 13% when using 256-QAM.

These results demonstrate that performing encryption Post-IFFT in the time-domain ensures a better cryptographic performance in terms of difference and correlation properties. The reason is that the IFFT block acts as a diffusion layer, which in turn increases the level of independence.

6.4.2 Key Sensitivity

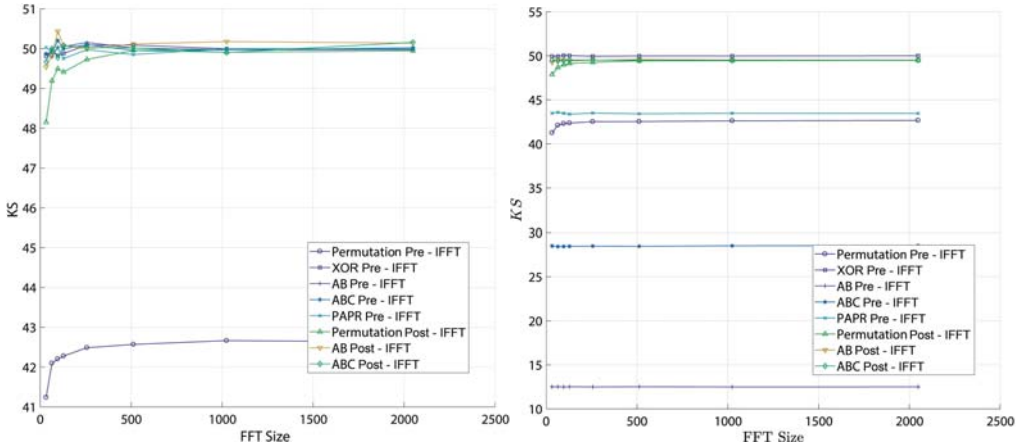
The key sensitivity test quantifies a specific scheme's sensitivity against any slight change in the key. Key sensitivity also refers to one of the most important prop-

erties in a robust cryptosystem, which is **confusion**. Confusion is achieved when half of the bits in the ciphertext change upon a one bit-change in the key stream. For all tested encryption schemes, the original OFDM symbols are encrypted separately with two dynamic keys (DK_1 and DK_2) and the Hamming distance between the two corresponding encrypted OFDM symbols is computed. In Fig. 6.8a, the results of the corresponding tests are presented for 1,000 OFDM symbols. The majority of values is close to 50%, which indicates that even the slightest change in the dynamic key leads to a different set of OFDM encrypted symbols (at least 50% difference) in the case of QPSK modulation. However, when using 256-QAM (Fig. 6.8b), the performance of Post-IFFT encryption schemes is better than their Pre-IFFT counterparts since, as mentioned previously, time-domain symbols are more random and scattered after the IFFT transformation (diffusion), therefore, encrypting Post-IFFT plaintext results in more secure ciphertext. The "AB" scheme of [110] has a major advantage, which is simplicity; however, when using 256-QAM, its performance is the worst since in this scheme only the phases of the symbols are changing, which results in low key sensitivity value close to 13%. In contrast, the proposed cipher "ABC" scheme adds a swapping operation, which enhances the key sensitivity (29%). The same test was applied for nonce sensitivity and similar results were obtained. The results demonstrate the robustness of the proposed scheme against attacks, especially key-related ones, compared to the "AB" scheme.

It should be noted that the stream cipher (XOR) achieves the best cryptograpic performance among all encryption schemes in all simulated tests since this scheme performs bit-level encryption where each bit in the transmitted scheme is randomly changed. However, this is computationally intensive since long pseudo-random key streams need to be generated for each OFDM frame or frame symbol. The permutation scheme is also a good cipher candidate, having a performance (security) close to the stream cipher scheme. However, this scheme has a longer processing delay since encryption is done on fixed block sizes and requires large key streams (permutation tables).

6.5 Cryptanalysis in OFDM Systems: Pre-IFFT versus Post-IFFT

An attacker has to overcome two obstacles: data dispersion and data encoding. An efficient data protection scheme should resist most known types of attacks such as statistical, differential, brute-force, chosen/known plaintext and ciphertext attacks [262]. This section discusses the proposed "ABC" scheme in the context of these attacks and compares its performance with other schemes in the literature. Note that the "ABC" scheme is considered to be public, and a cryptanalyst is assumed to have complete knowledge of all the steps, but none regarding the



(a) QPSK (b) 256-QAM

Figure 6.8: Key sensitivity test results. Measuring the bit difference between two encrypted OFDM symbols obtained from the same OFDM symbol but with two slightly different dynamic keys

secret key.

6.5.1 Statistical Attacks

To resist statistical attacks, the encrypted OFDM symbols should meet the requirements related to uniformity and random nonlinear recurrence. The results in the previous section prove the random nonlinear recurrence of OFDM encrypted symbols, especially those Post-IFFT. Furthermore, it has been shown that the encrypted OFDM symbols have a uniform distribution, and that no correlation exists between the encrypted and original symbols. Finally, the proposed dynamic key approach safe-guards against conducting statistical attacks. Therefore, no useful information can be inferred from the encrypted OFDM symbols, which validates the robustness of the proposed cipher "ABC" and its high resistance to statistical attacks.

6.5.2 Linear and Differential Attacks

Since the proposed scheme is based on a dynamic key approach, it is very difficult for an attacker to determine the rate at which the dynamic key changes, and to know the dynamic key itself in each session. Hence, the issues related to single data failure and accidental key disclosure are avoided using this approach. A differential attacker exploits the relation between the results of two encrypted OFDM symbols. However, in the proposed scheme, different dynamic keys and different cipher primitives are used for each OFDM symbol, making the relation

between two consecutive symbols highly uncorrelated. The key sensitivity test has demonstrated that two encrypted OFDM symbols originally derived from the same OFDM symbol using different keys, are significantly different. Consequently, differential and linear attacks become ineffective.

6.5.3 Brute-Force and Key-Related Attacks

The secret key space can be 2^{128}, 2^{196}, 2^{256}, which is sufficiently large to render any brute-force attack unfeasible. In addition, the tests conducted on the 512-bit dynamic key as well as the Nonce sensitivity tests indicate that any bit change in the secret key or the nonce causes a significant difference in the encrypted OFDM symbol (Fig. 6.8). This demonstrates the efficiency of the proposed key derivation function against key-related attacks.

It should be noted that in order to overcome replay attacks, a time stamp can be appended to each frame or frame symbol.

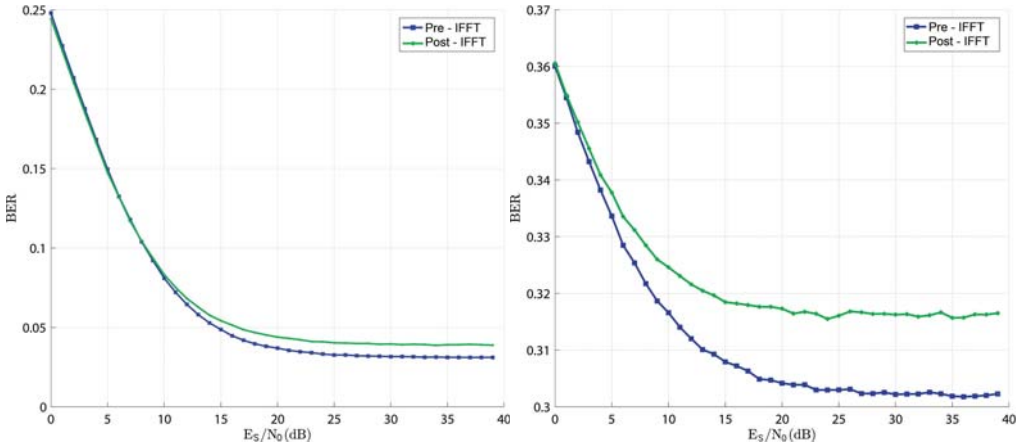
6.6 Performance Analysis in OFDM Systems: Pre-IFFT versus Post-IFFT

In this section, simulations are conducted using MATLAB to study the performance of different cipher schemes. The average number of symbols used in each simulation run is equal to 10^3. Several simulation runs are performed, and the Pre-IFFT encryption schemes are compared with Post-IFFT encryption schemes, in terms of BER, PAPR, as well as security level in the presence of Additive White Gaussian Noise (AWGN) and different frequency selective fading levels. QPSK and 256-QAM symbol modulation are used. Doppler frequency is equal to 200 Hz. Moreover, the performance of encryption schemes is studied under different FFT sizes and different values of Signal-to-Noise Power ratio (E_b/N_0).

6.6.1 BER Performance of Pre- and Post-IFFT Encryption Schemes

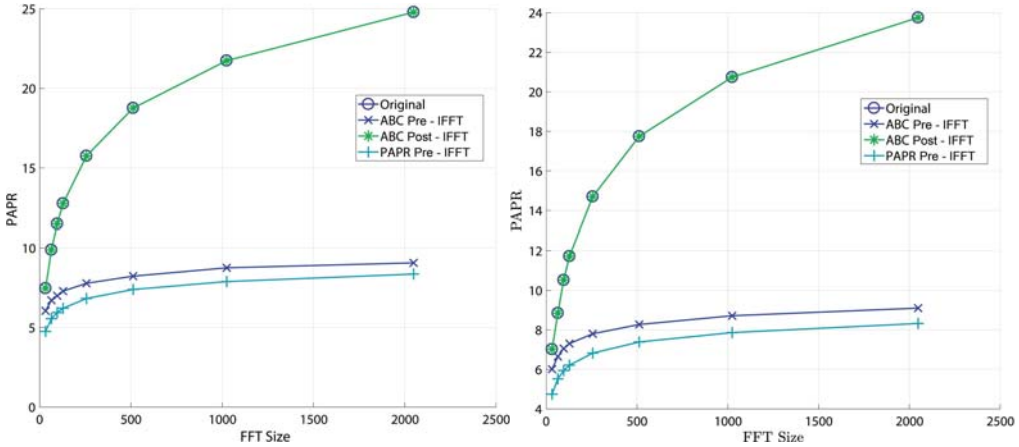
Here, the performance of existing OFDM encryption schemes is studied before and after the IFFT block for various levels of E_b/N_0, assuming an FFT size of 128, and a frequency selective channel with 6 paths. Figure 6.9 plots the BER versus E_b/N_0 using QPSK and 256-QAM, respectively. Note that the two BER curves in each figure correspond to the average BER value of the encryption schemes before and after IFFT, respectively.

As depicted in both figures, the gap between the two curves representing encryption before and after IFFT, increases for higher order modulation schemes



(a) QPSK (b) 256-QAM

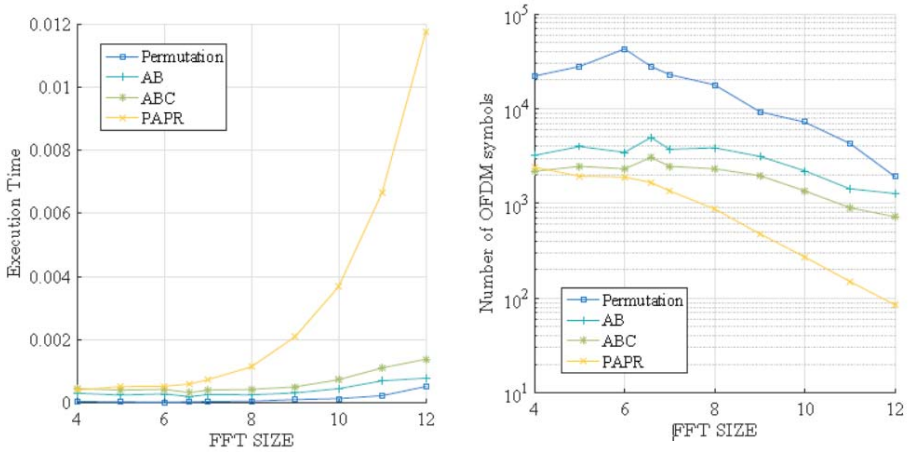
Figure 6.9: BER performance of Pre- and Post-IFFT encryption schemes versus E_b/N_0 using (a) QPSK, and (b) 256-QAM modulation



(a) QPSK (b) 256-QAM

Figure 6.10: Variation of PAPR as a function of FFT size for different encryption schemes

due to higher error propagation within symbols after the IFFT block, which results from diffusion. Additionally, the BER in the case of Post-IFFT encryption is higher than that of the Pre-IFFT case, since any bit error in the time-domain encrypted symbols will propagate further after passing through the FFT block due to the diffusion property of FFT. On the other hand, the BER for QPSK modulation is much lower than that for 256-QAM. For example, the BER in both cases (Pre- and Post-IFFT) is less than 5% in the case of QPSK, while it



(a) Execution Time (sec)

(b) Number of encrypted symbols per second

Figure 6.11: (a) Execution time (sec), and (b) number of encrypted symbols as a function of FFT size ($\log_2$) for different encryption schemes

is equal to 30.2% and 31.7% in the case of 256-QAM modulation, respectively. This difference is attributed to the modulation scheme itself since in QPSK, the distance between the 4 possible symbols is larger than that in 256-QAM, where any error (deviation in phase or amplitude) in the received symbol demodulates into a completely different set of data bits.

The difference in performance of each of the two classes of schemes is rather small. However, this difference becomes relevant when higher order modulation schemes are used. Consequently, there exists a trade-off between BER performance and the security level. More specifically, it has been shown that frequency-domain encryption schemes reduce the effect of channel fading and improve the BER compared to time-domain encryption schemes. However, time-domain encryption is more secure.

6.6.2 PAPR Simulations

High PAPR is a major drawback in OFDM. It is the maximum power of a sample in a specific OFDM symbol divided by the average power of that symbol. Figures 6.10a and 6.10b plot the PAPR values for the tested encryption schemes with QPSK and 256-QAM. The PAPR-reduction scheme, which chooses the sequence with minimum PAPR to be transmitted, performs better than the other schemes. The minimum attained PAPR is equals to 7 dB and 8 dB corresponding to QPSK and 256-QAM, respectively, for an FFT size of 2,048.

On the other hand, the proposed "ABC" scheme in the frequency domain

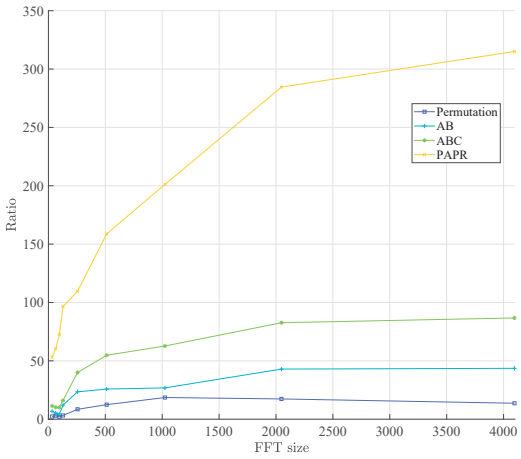


Figure 6.12: The ratio (%) of overheard in terms of execution time for different FFT sizes

achieves smaller PAPR values compared to all other schemes, expect the PAPR-reduction scheme, with values close to 9 dB for QPSK and 256-QAM modulation schemes when encrypting time-domain symbols. These values increase significantly for frequency-domain encryption schemes when using "ABC" encryption, resulting in very high PAPR values close to 25 dB, for FFT size equal to 2,048. Hence, these results validate the effectiveness and the need for time-domain encryption to ensure both low PAPR and better security performance.

Additionally, high PAPR values arise from the coherent addition of time-domain signals which have multiple high peaks. Consequently, increasing the FFT size leads to higher PAPR since a larger number of time-domain signals are being added. For instance, the PAPR reduction scheme has PAPR values equal to 7 dB and 8 dB for FFT sizes equal to $1,024$ and $2,048$, respectively. Moreover, simulation results show that PAPR is not affected by the order of the modulation scheme.

6.6.3 Execution Time

In principle, low execution time leads to reduced latency (fewer calculations), which is critical for devices with limited battery power. Therefore, evaluating the execution time is necessary when comparing cipher schemes.

The average time to encrypt OFDM symbols, having different FFT sizes of $32, 64, 128, 256, 512, 1024, 2048$, is calculated assuming the following software and hardware environment: MATLAB R2017b simulator, Intel Core i5 3 GHz CPU, 2 GB RAM, and Microsoft Windows 7 operating system.

In this subsection, IoT devices that utilize OFDM at the physical layer are considered. Simulation experiments are emulated in software, however, the ob-

tained results reflect the performance of encryption schemes in hardware implementation.

According to Fig. 6.11a, the "ABC" scheme ensures an acceptable execution time of 1.5 ms for FFT size of 2,048. The encryption with the longest execution time is the PAPR-reduction scheme (around 6.7 ms for FFT size $= 2,048$), which is logical since this scheme generates several random sequences and then takes the sequence with minimum PAPR. On the other hand, the stream cipher attains the lowest execution time due to its simplicity (below 0.2 ms).

In order to assess the overheard associated with encryption, the ratio (percentage) of encryption execution time over the total OFDM system execution time is plotted in Fig. 6.12. Logically, the execution time of an encryption scheme increases with the increase of FFT size and the number of operations required by a cipher scheme, as depicted in the Fig. 6.12. Being the simplest cipher scheme, the permutation technique has the lowest encryption overhead compared to the PAPR reduction scheme, the "AB" and the "ABC" schemes (below 50% for all FFT sizes). On the other hand, the PAPR reduction scheme has the highest overhead in terms of execution time, where this ratio increases beyond 100% for FFT sizes greater than 128. The "AB" and "ABC" schemes are less complex than the previously mentioned cipher scheme, having an overhead below 100% for all FFT sizes.

The plots shown in Fig. 6.11b corroborate the previous results. They show that the PAPR-reduction scheme encrypts fewer OFDM symbols per second than other schemes, while the stream cipher scheme attains a higher throughput owing to its small processing latency.

6.7 Security of OFDM Cipher Variants: The FBMC System

The OFDM-based scheme, presented in this chapter, can also be adapted and applied to any OFDM variant system. As such, the security of the filter bank is discussed in the following.

The encryption process at the FBMC transmitter side is depicted in Fig. 6.13. After serial-to-parallel conversion, the OQAM symbols (frequency-domain) are transformed into time-domain symbols via the IFFT block.

The IFFT output is then encrypted before entering the PPN filter bank. The encryption process is simply based on shuffling the post-IFFT symbols using the permutation table, $Pbox$, which changes the order of data symbols. Here, $Pbox$ depends on two main parameters, which are the pre-shared secret key and the common physical channel characteristics between users.

Afterwards, the encrypted symbols are processed with the prototype filter before being transmitted. This procedure is accomplished in two steps: first, the

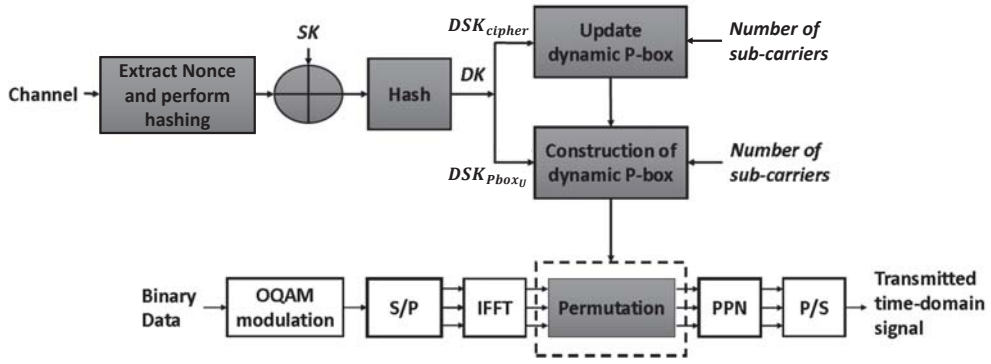


Figure 6.13: Block diagram of proposed FBMC-based cipher scheme

encrypted symbols are duplicated and then multiplied by the impulse response of the filter (in the time-domain). The resulting filtered frames are shifted by half a symbol period and are then added all together to form the final transmitted signal.

At the receiver side, the decryption process depends on the inverse permutation table, $Pbox^{-1}$, where the received signal is first processed by the PPN and then decrypted using $Pbox^{-1}$. The resulting time-domain symbols are transformed into frequency-domain symbols using the FFT transformation and then demodulated.

Note that having a static permutation table ($Pbox$) makes a cipher scheme vulnerable to chosen/known plaintext/ciphertext attacks. Therefore, in the proposed scheme, the permutation table is shuffled for every input frame where a new permutation box is generated to ensure robust security for FBMC systems. By dynamically changing the permutation table using the proposed channel-based key, this method becomes very effective and robust against several attacks while maintaining a low complexity (one round and one operation).

In order to generate the cipher primitives, the dynamic key is divided into two sub-keys (256 bits each). One sub-key is used for data encryption and the other is used to shuffle the permutation box. Similar to the OFDM case, DSK_{cipher} is used to produce a dynamic key-dependent permutation table ($Pbox$), which allows the dynamic permutation of unfiltered time-domain symbols (before PPN). The length of $Pbox$ depends on the size of the IFFT output block. On the other hand, DSK_{Pbox_U} is used to shuffle $Pbox$ for every new input frame (updating the permutation box).

The dynamic key is sensitive to any change that occurs either to the channel or to the secret key, and thus, its dynamic property guarantees a high level of security.

For the security assessment of this solution, several security tests have been performed (uniformity, recurrence, independence, entropy and sensitivity tests).

The obtained results confirm the robustness of the FBMC physical layer security solution. Moreover, performance has been evaluated using the bit error rate and execution time. Both validate the efficiency of the proposed scheme.

Chapter 7

Generic Data Confidentiality for IoT Systems

The data confidentiality scheme, proposed in this chapter, also exploits the random and dynamic features of the physical layer to secure data transmitted over wireless channels. This scheme is an enhanced version of the previously discussed approach. By introducing the notion of dynamicity to the secret session key and the cipher primitives, one guards against confidentiality attacks and other attempts to acquire any useful information related to the utilized keys and the relayed data. The proposed scheme is generic in the sense that it applies to all post-modulation data frames in any 5G system (such as the IoT system), independent of the multiple access scheme (not necessarily OFDM systems).

Existing schemes based on PLS rely on the channel randomness as a key feature for securing data, where both the transmitter and receiver estimate the channel state information (CSI), and then encrypt data using an encryption key that is derived from the common channel information [8, 263]. However, such techniques are considered weak; channel parameters can be acquired easily, if one is able to synchronize to the transmitted frames (detecting the preamble and performing correlation). Accordingly, any eavesdropper is able to estimate the channel between two users and extract the CSI. To overcome this issue, the proposed scheme also encrypts the packet preamble to safeguard against any packet synchronization or channel estimation attempts by illegitimate users.

7.1 Dynamic Sub-key and Cipher Primitive Generation

Here, a general input frame is considered. It is divided into three main parts: preamble, header and data. The data field consists of NB_F frame symbols and each frame symbol contains NB_{FS} complex modulation symbols. To enhance the level of security, it is recommended to encrypt the frame preamble and the

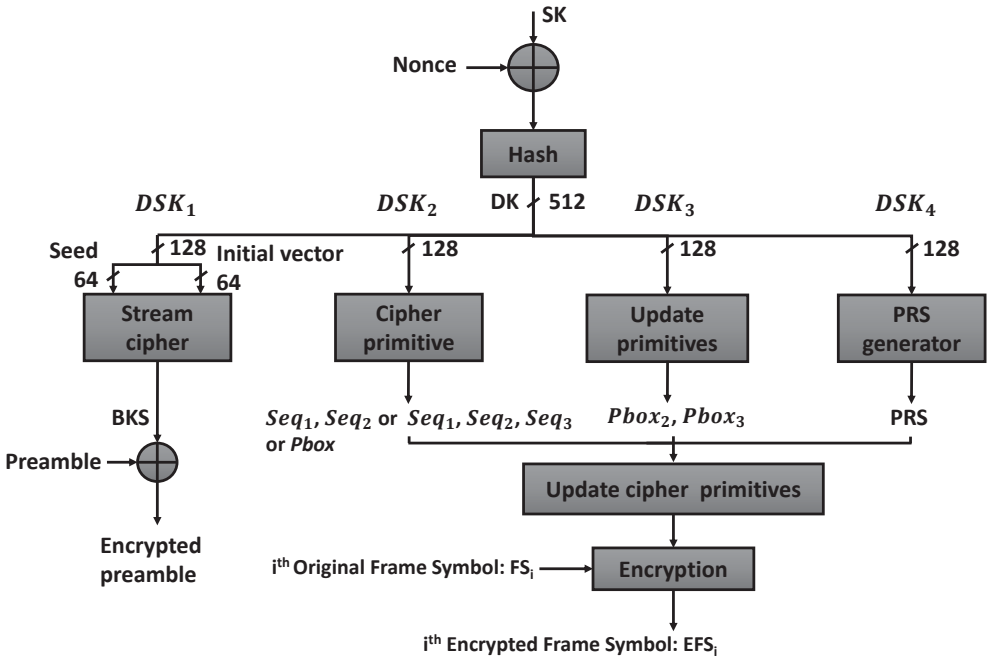


Figure 7.1: Proposed dynamic sub-key generation technique for the generalized scheme

data field with two different dynamic keys. Data confidentiality is achieved by encrypting the transmitted data using physical channel characteristics. However, if the adversary is able to synchronize to the transmitted frames, he will be able to extract channel characteristics, derive the keys used for data encryption and decrypt the transmitted ciphertext. To overcome this vulnerability, the frame preamble should be encrypted using a common dynamic key, only known to the legitimate users. The generation of the dynamic sub-keys is discussed next.

The produced DK, which has a length of 512 bits, is divided into four different sub-keys (128 bits each) $DK = \{DSK_1,\ DSK_2,\ DSK_3,\ DSK_4\}$. While, DSK_1 is needed for preamble encryption, DSK_2 is required for the generation of cipher primitives. DSK_3 and DSK_4 are used for the generation of the update permutation boxes ($Pbox_2$ and $Pbox_3$) and the Pseudo-Random Sequence (PRS), respectively (Fig. 7.1).

- **Preamble encryption:** The first dynamic key, DSK_1, is divided into two equal parts: $DSK1 = \{DSK_{1,1}, DSK_{1,2}\}$, which represent the seed and the initial vector (inputs to any stream cipher) needed to produce the required binary key-stream, BKS. Afterwards, the packet preamble is encrypted by XORing it with BKS.

- **Cipher primitives:** DSK_2 is used to generate the cipher primitives for

any encryption scheme: the permutation table, $Pbox$, for the permutation encryption scheme, the two binary pseudo-random sequences (Seq_1 and Seq_2) for the phase encryption scheme, or the sequences Seq_1, Seq_2, and Seq_3 for the enhanced phase encryption scheme. To produce the dynamic key-dependent permutation table, the technique presented in [264], which is based on the modified key setup algorithm of RC4, can be used. Sequences Seq_1, Seq_2 and Seq_3 can be produced by using any stream cipher.

- **Updating cipher primitives:** In order to enhance the security of encrypted symbols, new cipher primitives can be generated for every frame symbol (FS). This can be very complex and computationally exhaustive. Hence, it is recommended to change the cipher primitives based on the update permutation tables, $Pbox_2$ or $Pbox_3$. Using either one of the permutation tables depends on the generated Pseudo-Random Sequence PRS. If the i^{th} PRS bit, corresponding to the i^{th} frame symbol, is equal to 0, $Pbox_2$ is used to update the cipher primitive, otherwise $Pbox_3$ is used ($i \leq NB_F$, where NB_F is the size of one frame (number of frame symbols in one frame)). In this step, DSK_3 is used to derive both update tables $Pbox_2$ and $Pbox_3$. It should also be noted that the length of these tables is related to the length of the cipher primitives.

- **Pseudo-Random Sequence (PRS) generation:** A sequence equal to the number of frame symbols in a frame is generated using DSK_4. This sequence consists of 0's and 1's. For every frame symbol, $Pbox_2$ is used if the corresponding bit is 0 and $Pbox_3$ is used if the bit is equal to 1.

Using this approach, any bit change in the secret key or in the nonce, will result in a new dynamic key and will lead to a completely different set of sub-keys, cipher primitives and permutation tables, which renders the cryptanalysis task very challenging.

7.2 Data and Preamble Encryption

The proposed two-level encryption process is shown in Fig. 7.2. First, the dynamic key and the corresponding dynamic sub-keys are generated. Then, the packet preamble is XORed with the generated key-stream, BKS, which is obtained using an initial vector, a seed and a stream cipher. Similarly, an encrypted frame symbol (EFS) is derived from the original frame symbol by performing physical layer encryption using the produced cipher primitives (any cipher primitive can be utilized). This cipher primitive is updated after each new frame symbol (FS). Finally, the encrypted preamble and the encrypted data symbols are reconstructed to form the complete transmission frame. Here, encryption is performed at the modulation symbol level, and error propagation is minimized

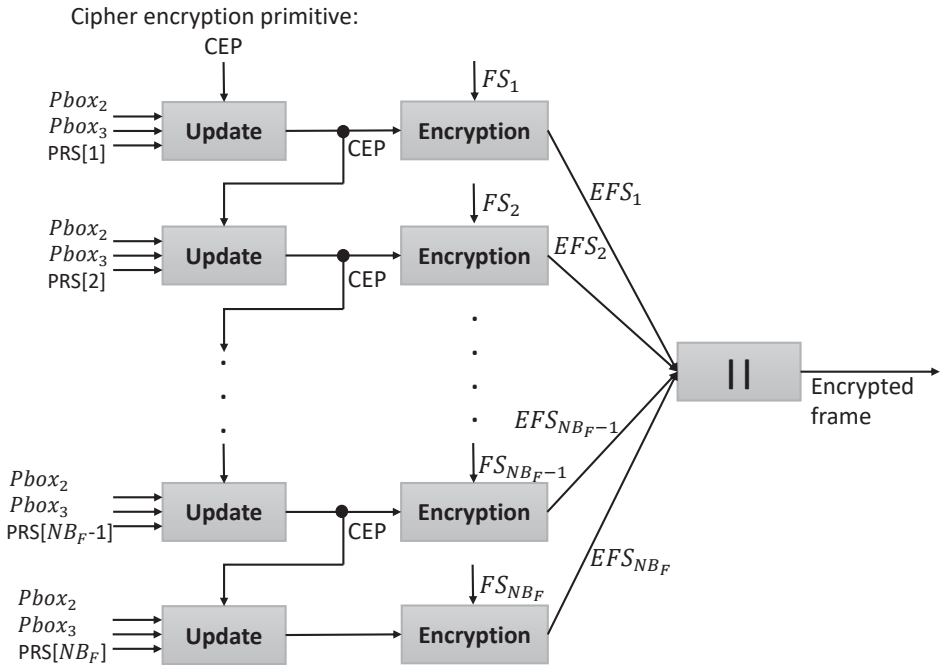


Figure 7.2: Proposed generalized cipher scheme

and mitigated since error in one symbol doesn't propagate or affect other symbols in the frame.

A legitimate receiver will use the same steps and the same dynamic key to produce the same cipher primitives. The only difference at the receiver's side is the use of the different decryption algorithms for the data and the preamble in order to recover the original frame. The decryption algorithm has minor modifications compared to the encryption algorithm; it requires inverse cipher primitives (for example inverse permutation box $Pbox^{-1}$) and reverse operations depending on the used cipher scheme.

7.3 Update Cipher Primitive Process

For the permutation encryption scheme, $Pbox$, $Pbox_2$ and $Pbox_3$ have lengths equal to NB_{FS} (size of each frame symbol in a frame). On the other hand, $Pbox_2$ and $Pbox_3$ have lengths equal to $2 \times NB_{FS}$ for the phase encryption scheme and $3 \times NB_{FS}$ for the enhanced phase encryption scheme.

In the following, the proposed techniques for updating the cipher primitives of the permutation (Algorithm 3) and the phase encryption schemes (Algorithm 4), are described.

Algorithm 3 The proposed update permutation process

Input: A cipher primitive $Pbox$,
two update permutation tables ($Pbox_2$ and $Pbox_3$),
and the i^{th} bit element of PRS.
Output: Updated permutation table $Pbox$

1: **procedure** UP_PERMU_TAB($Pbox,\ Pbox_2,\ Pbox_3, PRS[i]$)
2: **if** $PRS[i] = 0$ **then**
3: $Pbox = \mathsf{Perm}(Pbox,\ Pbox_2)$
4: **else**
5: $Pbox = \mathsf{Perm}(Pbox,\ Pbox_3)$
6: **end if**
7: **return** $Pbox$
8: **end procedure**

Algorithm 4 The proposed update phase encryption process

Input: Two binary sequences Seq_1 and Seq_2,
two update permutation tables($Pbox_2$ and $Pbox_3$),
and the i^{th} bit element of PRS.
Output: Update Seq_1 and Seq_2 cipher primitives

1: **procedure** UP_AB_PRIM($Seq_1,\ Seq_2,\ Pbox_2,\ Pbox_3, PRS[i]$)
2: $vec_{temp} \leftarrow Seq_1||Seq_2$
3: $vec'_{temp} \leftarrow Up\_Permu\_Tab(vec_{temp},\ Pbox_2,\ Pbox_3, PRS[i])$
4: $Seq_1 \leftarrow vec'_{temp}[1 \rightarrow NB_{FS}]$
5: $Seq_2 \leftarrow vec'_{temp}[NB_{FS} + 1 \rightarrow 2 \times NB_{FS}]$
6: **return** $(Seq_1,\ Seq_2)$
7: **end procedure**

The phase encryption and enhanced phase encryption schemes use the permutation table $Pbox_2$ or $Pbox_3$ (based on the corresponding PRS) to permute the temporary vector, vec_{temp}, that contains the concatenated pseudo-random sequences. From the obtained permuted vector, new pseudo-random sequences are derived. In contrast, no additional operations are needed for the permutation scheme, since $Pbox$ is directly used to encrypt a frame symbol ($Pbox$ is directly permuted using either one of the update permutation boxes).

Note that the steps required for updating the enhanced phase encryption cipher primitives are similar to those of the phase encryption scheme (Algorithm 4), except for the fact that vec_{temp} is equal to $Seq_1||Seq_2||Seq_3$ instead of $Seq_1||Seq_2$ and Seq_3 will be equal to the $vec'_{temp}[2 \times NB_{FS} + 1 \rightarrow 3 \times NB_{FS}]$.

7.4 Security Analysis of the Generalized Cipher Scheme

The security analysis in this section is divided into three main parts, which are: the security of the dynamic key, the security of the update process and the security of the produced encrypted data.

The utilized metrics are well-known metrics used to quantify the security and performance of traditional cryptographic algorithms. Here, these metrics are adapted to quantify the security of the proposed scheme at the physical layer. In this study, four cipher schemes are employed: the permutation scheme, the 2-D permutation scheme, the phase encryption scheme and the enhanced phase encryption scheme.

7.4.1 The Security of the Dynamic Key

The security level of the proposed cipher scheme depends on the security of the dynamic key. Therefore, testing the randomness degree of the proposed dynamic-key generation function is necessary. For this purpose, the empirical NIST statistical test [265] is applied on 100 sequences of one million bits, produced with 100 different secret keys (SK) and channel-based nonces in order to validate the security of the proposed dynamic-key derivation scheme. In Fig. 7.3, the obtained NIST proportion values and their corresponding P-values are shown. The obtained P-value is greater than 0.01, which indicates that the null hypothesis is not rejected and the produced sequences reach a high level of randomness. As it can be inferred, the plotted proportion values (marked in blue) are above the threshold represented by the red line, which proves that the proposed dynamic key generation scheme passes all of the statistical tests and attains a high randomness level.

7.4.2 The Security of the Update Process

Next, the security level of the proposed "update cipher primitive technique" is studied and quantified in order to prove its secure deployment in the ciphering process. In this context, the recurrence and the correlation coefficient, ρ, are used to examine the randomness of the produced dynamic permutation boxes for different input frames. These tests were applied for $1,000$ random dynamic keys.

In fact, the recurrence plot serves to evaluate the randomness and estimate the correlation among the data as in [266]. Considering a sequence $Seq_i = Seq_{i,1}, Seq_{i,2}, Seq_{i,3}, \ldots$, a vector with delay $t \geq 1$ can be constructed by: $Seq_i(t) = Seq_i, Seq_{i,1+t}, Seq_{i,2+t}, Seq_{i,3+t}, \ldots$. Whereas, the correlation coefficient $\rho_{(vec_1, vec_2)}$ between two vectors vec_1 and vec_2 can be calculated using the following equation:

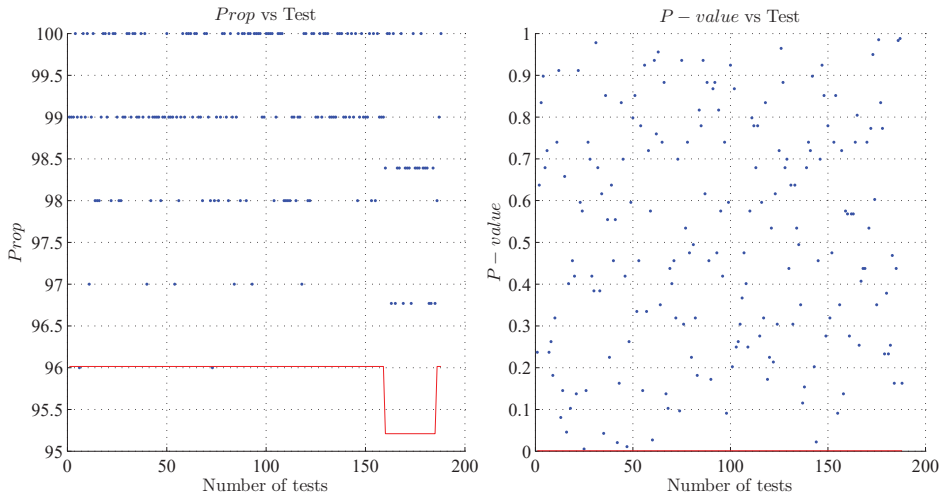


Figure 7.3: NIST test results: (a) Proportion values, and (b) P-value

$$\rho_{(vec_1,vec_2)} = \frac{cov(vec_1, vec_2)}{\sqrt{D(vec_1) \times D(vec_2)}} \tag{7.1}$$

where

$$E_{vec_1} = \frac{1}{N} \times \sum_{i=1}^{N} vec_{1,i}$$

$$D_{vec_1} = \frac{1}{N} \times \sum_{i=1}^{N} (vec_{1,i} - E(vec_1))^2$$

$$cov(vec_1, vec_2) = \frac{1}{N} \times \sum_{i=1}^{N} (vec_{1,i} - E(vec_1))(vec_{2,i} - E(vec_2))$$

Figure 7.4a shows the recurrence of a randomly produced permutation table (highly scattered). Figure 7.4b, on the other hand, shows the cumulative distribution function of the correlation coefficient between the recurrence of the permuted index versus 1,000 random dynamic keys. It is clear that for any dynamic key, the produced permutation table has a high randomness degree since the correlation coefficient of its recurrence is always close to zero (optimal value). More specifically, most values, which are uniformly distributed (close to a straight line), fall within the range $\{-0.1, 0.1\}$ which is close to the desired value (zero).

Additionally, Fig. 7.4c shows the CDF of the correlation coefficient between the original (primary *Pbox*) and the updated permutation table as a function of the number of iterations. The correlation coefficient is always close to zero

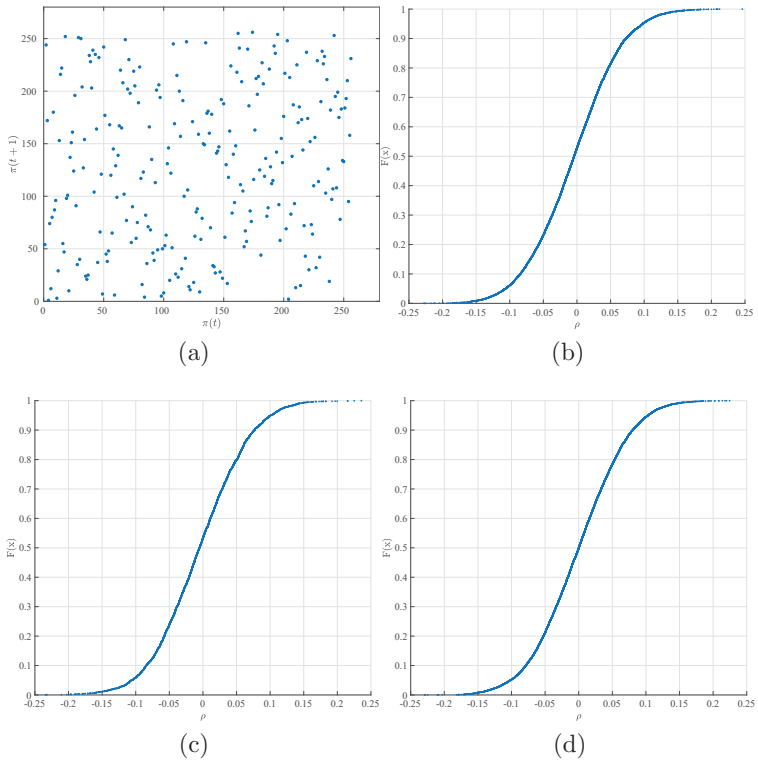


Figure 7.4: (a) Recurrence of a pseudo-random primary permutation table, (b) the Empirical Cumulative Distribution Function (ECDF) (1000 times) of the correlation coefficient of the recurrence of primary permutation tables, (c) the ECDF of the correlation coefficient between a primary permutation table and its updated version (permuted version), and (d) the ECDF of the coefficient correlation between two successive permutation tables

(range $\{-0.1, 0.1\}$). This proves the independence of the original *Pbox* from the updated permutation boxes. Finally, Fig. 7.4d shows the CDF of the correlation coefficient between two successive updated permutation boxes for 1,000 iterations. Similarly, the correlation coefficient is close to zero (range $\{-0.1, 0.1\}$), which also indicates the independence of any two successive permutation boxes.

The obtained results show that no correlation exists between the primary and the updated permutation boxes, as well as between any two successively updated permutation boxes. This clearly indicates the high dynamicity and uniqueness level of the produced permutation tables. Consequently, this provides high immunity against eavesdropping attacks since no useful information can be extracted from the dynamically permuted ciphered data.

7.4.3 The Security of Encrypted Data

In order to improve the immunity of a specific cipher scheme against statistical attacks, encrypted frames should exhibit a high level of randomness. This can be achieved by performing the uniformity and the independence tests.

Uniformity: One way to evaluate uniformity is by plotting the Probability Density Function (PDF) and assessing it visually. Each encrypted symbol should have an equal existence probability close to $\frac{1}{NB_{FS}}$. Figures 7.5i, 7.5j, 7.5k, and 7.5l show the PDF of the original message and the encrypted frames using the generalized cipher approach for different encryption schemes, respectively. As shown in Fig. 7.5i, original data has a normal distribution, as expected, whereas Fig. 7.5j, 7.5k, 7.5l show that the "AB", "ABC" and the 2-D permutation scheme all have uniform distributions. Therefore, the proposed generalized cipher approach produces encrypted frames having the desired uniformity level.

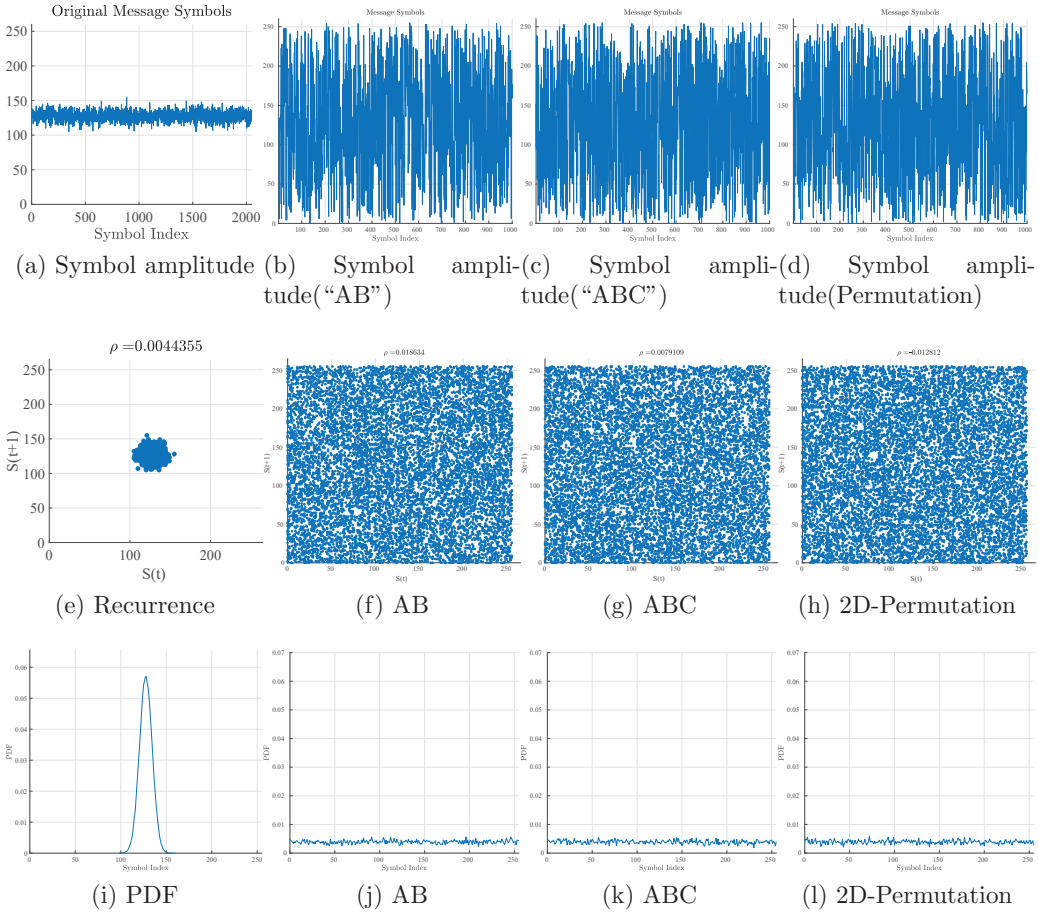
Recurrence Test: Another important property of a good encryption scheme is the high level of randomness of encrypted messages within a message space, which is demonstrated by the recurrence test and the correlation coefficient. In other words, the recurrence plot corresponding to good cipher scheme should be uniformly distributed within the available space for a scheme to be considered as a good encryption candidate.

Figures 7.5f, 7.5g and 7.5h show the recurrence plots of different frames symbols using the generalized cipher approach, for different encryption schemes. All of these plots show a good level of randomness in which encrypted symbols are well scattered and random within the available space, unlike Fig. 7.5e which corresponds to the recurrence plot of the original data having a normal distribution (concentrated in one small area). Additionally, the original message has a fixed range of amplitudes (Fig. 7.5a), while the tested cipher schemes employing the generalized scheme have random and varying amplitudes (Figures 7.5b, 7.5c, and 7.5d).

Independence: An encryption scheme is considered secure against statistical attacks, if it generates encrypted frames that are independent from the original ones. To do so, one can measure the difference in bits between the original frame and the encrypted frame (dif), which in principle should be close to 50%. In this test, plain frame symbol FS_i is encrypted to produce the cipher frame symbol EFS_i, where $i = 1,\ 2,\ \ldots,\ NB_{FS}$. The difference test is computed according to the following equation:

$$dif = \frac{\sum_{i=1}^{NB_{FS}} dec2bin(FS_i) \oplus dec2bin(E_{DK}(EFS_i))}{NB_{FS}} \tag{7.2}$$

Also, the cross-correlation of the symbol itself (original and encrypted) can



(a) Symbol amplitude (b) Symbol amplitude("AB") (c) Symbol amplitude("ABC") (d) Symbol amplitude(Permutation)

(e) Recurrence (f) AB (g) ABC (h) 2D-Permutation

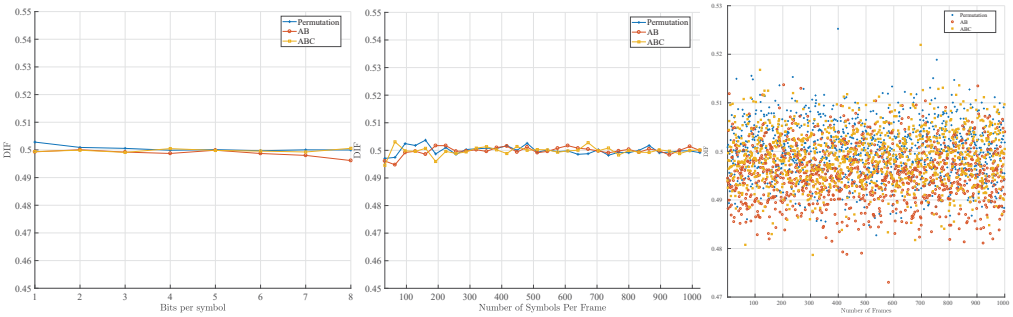
(i) PDF (j) AB (k) ABC (l) 2D-Permutation

Figure 7.5: Symbol amplitude, Recurrence and PDF of chosen original message and the corresponding encrypted frames using the generalized scheme

be used. Independence is reached if the cross-correlation is close to 0.

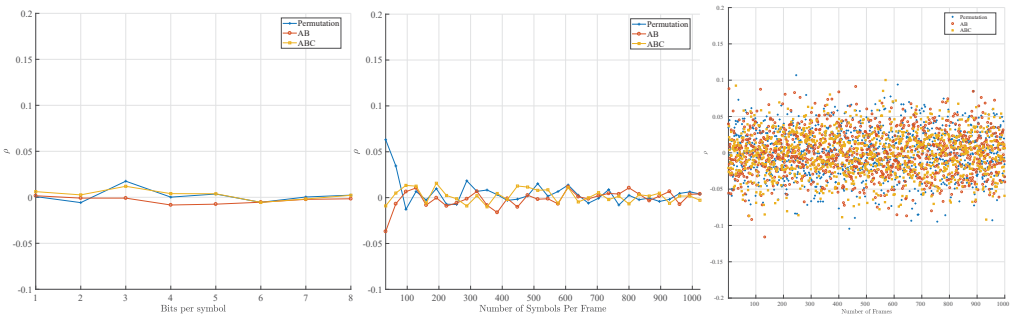
Figures 7.6 and 7.7 correspond to the average difference between the original and the encrypted frames, as well as their corresponding correlation coefficients for 1,000 iterations. As it can be depicted in Figures 7.6a, 7.6b and 7.6c, all encryption schemes have a difference values close to the desired value (50%) when varying the number of bits per symbol, the number of symbols per frame and the number of transmitted frames as a whole. On the other hand, Figures 7.7a, 7.7b and 7.7c show that the correlation coefficients between the original and encrypted frames, for all encryption schemes, is close to the desired value, 0. This result is similar for the three cases mentioned above which are: the variation of 1) bits per symbol, 2) symbols per frame, and 3) number of transmitted frames.

These results clearly indicate that the proposed cipher scheme is immune against statistical attacks.



(a) (b) (c)

Figure 7.6: Independence tests: Difference measurements versus (a) the number of bits per modulation symbol, (b) the number of symbols per frame, and (c) the number of transmitted frames, using the generalized scheme



(a) (b) (c)

Figure 7.7: Correlation coefficients between original and encrypted frames versus (a) the number of bits per symbol, (b) the number of symbols per frame, and (c) the number of transmitted frames, using the generalized scheme

Key Sensitivity: The key sensitivity test evaluates the bit difference in encrypted frames when using different keys (slight change in the keys). This difference between both encrypted frame symbols at the bit level should be close to 50%. Indeed, the sensitivity of the dynamic key DK is calculated as follows:

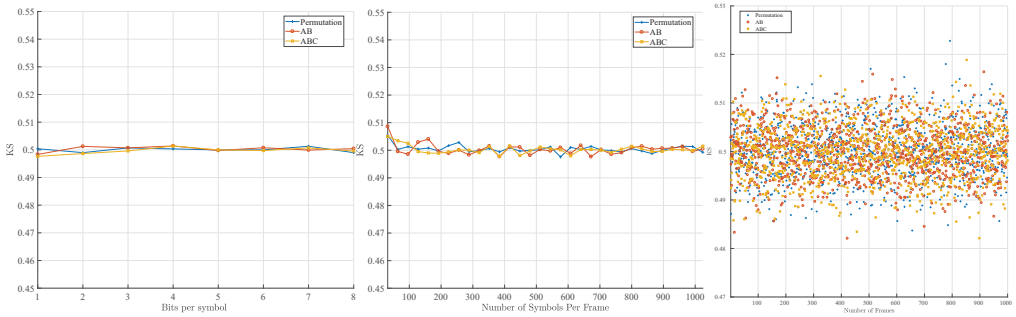
$$KS = \frac{\sum_{i=1}^{NB_{FS}} dec2bin(E_{DK}(FS_i)) \oplus dec2bin(E_{DK'}(FS_i))}{NB_{FS}} \tag{7.3}$$

where all the elements of DK are equal to those of DK', except for the Least Significant Bit (LSB) of a random byte, and NB_{FS} is the length of the original and ciphered frame symbols (in bits).

In this test, and for all encryption schemes, an original data frame is encrypted

with two dynamic keys (DK and DK' of size 512 bits) and the Hamming distance between the two corresponding encrypted frames is computed at the bit level.

The key sensitivity results are shown in Fig. 7.8 for 1,000 iterations. Figures 7.8a, 7.8b and 7.8c prove that for all cipher schemes using the proposed generalized cipher scheme, the key sensitivity has a value very close to 50%. The key sensitivity values are very close to the desired value, whether the number of bits per symbol or the number of symbols per frame or the number of transmitted frames, is varied. Consequently, the generalized cipher approach is immune against linear and differential attacks, weak keys and related-key attacks. Moreover, the dynamic key has a size of 512 bits, which is sufficient to resist brute force attacks.



(a) (b) (c)

Figure 7.8: Key sensitivity measurements versus (a) the number of bits per symbol, (b) the number of symbols per frame, and (c) the number of transmitted frames, using the generalized scheme

Plaintext Sensitivity: In this test, two plain frame symbols FS and FS' that are different by only one bit are encrypted separately to produce two cipher frame symbols EFS and EFS'. The plaintext sensitivity is computed according to the following equation:

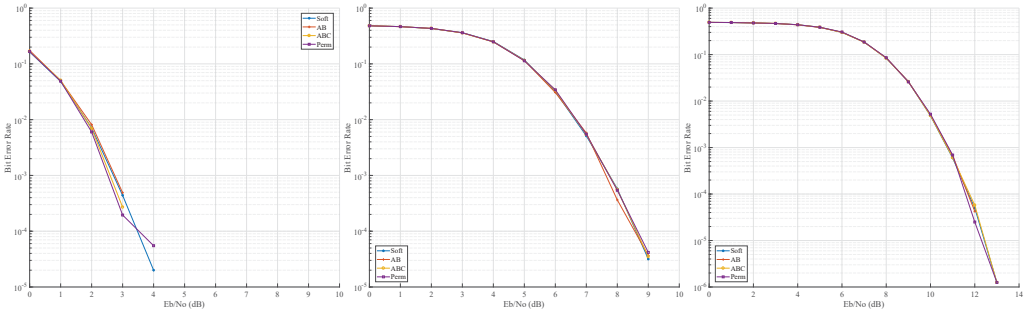
$$PS = \frac{\sum_{i=1}^{NB_{FS}} dec2bin(E_{DK}(FS_i)) \oplus dec2bin(E_{DK}(FS'_i))}{NB_{FS}} \tag{7.4}$$

For the proposed approach, the plaintext sensitivity is neglected since different dynamic keys are generated for every new session, and different cipher primitives are derived for every input frame and frame symbol. Accordingly, the proposed cipher scheme produces completely different encrypted frames, and thus, ensures the avalanche effect.

7.5 Performance Evaluation of the Generalized Cipher Scheme

In this section, several criteria and tests are presented to prove that the proposed solution achieves a good and efficient performance and copes with practical wireless 5G IoT communications. An efficient PLS scheme should, in principle, have low latency, low memory consumption and minimal costs in terms of resources. This will be verified mainly due to using a single round structure that consists of simple operations.

Simulations tests are conducted in MATLAB to show the performance of the proposed key generation scheme and cipher schemes ("ABC" and permutation) in comparison with other encryption methods ("AB"). Throughout this section, the latency and the required memory consumption are introduced for the different encryption schemes, in addition to the effect of error propagation. Moreover, the performance of the encryption scheme is studied under different symbol sizes.



(a) QPSK (b) 64-QAM (c) 256-QAM

Figure 7.9: BER performance of the generalized cipher approach using different encryption schemes versus E_b/N_0 using (a) QPSK, (b) 64-QAM and (c) 256-QAM modulation

7.5.1 Error Propagation

An important criterion that should be considered for any PLS cipher scheme is error tolerance, which means that there should exist no error propagation among encrypted symbols. Interference and noise, which exist in transmission channels, are the main causes of errors. A bit error means that a '0' bit is substituted with a '1' bit or vice-versa. Consequently, this error might propagate and lead to data corruption, which is a big challenge since there exists a strong trade-off between the Avalanche effect and error propagation, as seen in the traditional cryptographic algorithms [267]. Using the proposed approach, if a bit error takes place in any modulation symbol of the encrypted data frame, it does not affect

other modulation symbols since the effect of erroneous bits is restricted to specific bits (corresponding to the erroneous modulation symbol) at the same position in the encrypted and decrypted frame. As such, error propagation is mitigated and minimized.

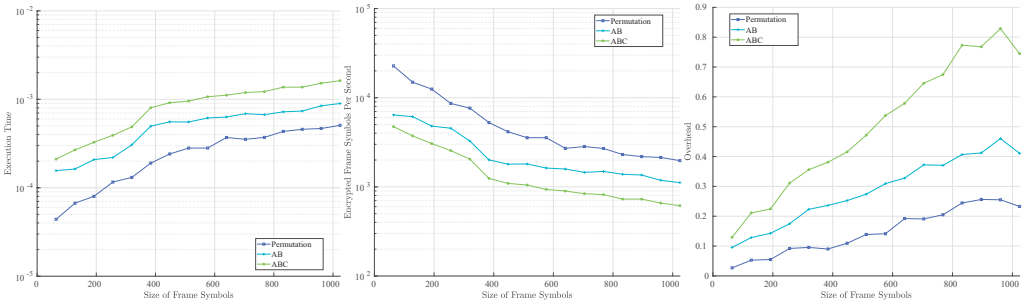
Furthermore, the performance of the generalized cipher approach using different cipher schemes is analyzed in terms of Bit Error Rate (BER). The average number of symbols used in each simulation run is equal to 10^4.

Figure 7.9 shows the average BER curves of the original data sent without encryption in comparison with the encrypted data frames using the generalized cipher approach for different encryption schemes. Different values of signal-to-noise power ratio (E_b/N_0) and modulation schemes (QPSK, 64-QAM and 256-QAM) are taken into account. According to the obtained results, the BER reaches a minimum value of 4 dB for QPSK, 9 dB for 64-QAM and 13 dB for 256-QAM. This difference is attributed to the modulation scheme itself since in QPSK, the distance between constellation symbols (4 symbols) is larger than that in 256-QAM (256 symbols) where any error (deviation in phase or amplitude) in the received symbol demodulates into a completely different set of data bits. Hence, E_b/N_0 should be increased whenever higher order modulation schemes are used to achieve an acceptable BER. In fact, if a bit error appears in any of the modulation symbols in the encrypted data frame, it should not affect other modulation symbols. This is achieved, as shown in the figures above, since the effect of erroneous bits in the modulated symbol is restricted to the same position in the encrypted and decrypted frames where all curves in each of the BER plots have the same trend (matched curves). This means that the error does not propagate to other modulation symbols and it will not affect neighboring modulation symbols. Hence, the proposed generalized cipher scheme is immune to error propagation, and it does not affect the BER.

It should be noted that soft decision decoding using convolutional encoding is used for channel encoding/decoding, which justifies the BER performance. Moreover, several modifications compared to traditional simulations are required in this case.

7.5.2 Execution Time

A a low execution time reflects into low energy consumption, which is essential for resource-limited devices. For this purpose, the average time ($1,000$ iterations) to encrypt a frame symbol, having flexible sizes of $32, 64, 128, 256, 512, 1024$, and 2048, is calculated. The following software and hardware environment is used: **Matlab R2013b simulator, Intel Core i5, 3 GHz CPU, 2 GB RAM Intel and the Microsoft Windows 7 operating system**. The obtained results are illustrated in Fig. 7.10 (generalized scheme). As it can be inferred the permutation cipher scheme has the lowest computation complexity among all other cipher schemes. One the other hand, the "ABC" scheme requires the



(a) Execution Time (sec) (b) Number of encrypted symbols per second (c) Overhead ratio

Figure 7.10: (a) Execution time (sec), (b) number of encrypted frame symbols as a function of frame symbol size ($\log_2$) using the generalized cipher approach for different encryption schemes, and (c) the ratio (%) of overheard in terms of execution time for different frame symbol sizes compared to modulation and channel coding

maximum computational complexity and overhead, which is approximately equal to twice that of the "AB" scheme. Also, the "AB" scheme is twice more complex than the permutation cipher scheme since it requires conversion and re-conversion operations (separating each complex symbol into real and imaginary components, and then, combining them back after encryption). Finally, Fig. 7.10c shows the execution-time overhead of the encryption process with respect to modulation and channel coding.

Chapter 8

Data Confidentiality for NOMA-based IoT Systems

To achieve robust data confidentiality in NOMA-based IoT systems, a lightweight cipher scheme based on DK is proposed. Currently, PD-NOMA-based systems suffer from a major drawback, that is near users (low power coefficients) are able to decode and obtain the signals of farther users (large power coefficients), using SIC. One simple and direct approach is to encrypt the signals of each user before multiplying them with power coefficients and transmitting them.

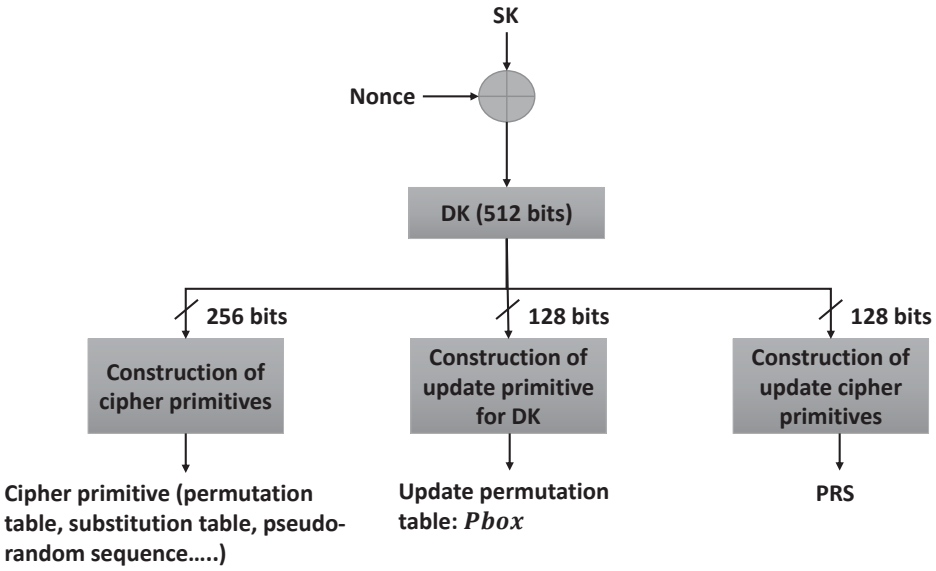


Figure 8.1: The proposed sub-key generation process for achieving data confidentiality in NOMA systems

8.1 Dynamic Sub-Key Generation

Three sub-keys, of lengths 256, 128 and 128 bits, are produced from the dynamic key, as illustrated in Fig. 8.1. The first sub-key is used to generate the cipher primitives needed to encrypt the data of each NOMA user (depends on the selected operation such as permutation, substitution, masking, and phase scrambling). The second sub-key is used to generate an update permutation table, *Pbox*, which is used to update (pseudo-randomly permute or shuffle) the dynamic key for every new input frame (frame consists of a set frame symbols). *Pbox* has the same length of the dynamic key. Finally, the third sub-key is used to derive PRS which is a pseudo-random sequence, having a size equal to one frame. More specifically, the number of elements in PRS is equal to number of frame symbols in one frame ($1 \times NB_F$). For every new input frame symbol, PRS is used to circularly shift the utilized cipher primitives (updating process according to the corresponding PRS value).

For the construction of cipher primitives (permutation or substitution tables), the techniques presented in [264] are adopted. For the masking and phase shuffling cipher primitives, key-stream sequences are required; these can be generated using any stream cipher.

8.2 Encryption Model

In a typical digital communication system, input data is encoded using source and channel encoding, and then modulated. Here, any M-QAM (Quadrature Amplitude Modulation) scheme can be used such as 4-QAM, 16-QAM, 64-QAM or 256-QAM. The main difference between these schemes is the number of bits, that gets mapped to one constellation point (modulation symbol). Following modulation, the proposed dynamic key-dependent cipher scheme is applied to each modulation symbol, to obtain the encrypted frame symbol, and consequently the overall frame, which contains NB_F frame symbols. Any efficient cipher scheme, such as substitution, permutation, phase encryption or masking, can be used in this step.

Particularly, the contents of each frame symbol (which contains multiple modulation symbols) are encrypted using the previously derived cipher primitives. Afterwards, the CP and packet preamble are inserted and the corresponding power coefficient is multiplied. Thus, the superimposed downlink signal becomes:

$$x_D(t) = \sum_{k=1}^{K} \sqrt{\alpha_k P_{BS}} c_{D,k}(t), \tag{8.1}$$

where, $c_{D,k}(t)$ is the downlink encrypted signal of the k^{th} user.

As such, the superimposed signals of PD-NOMA users are secured against internal (legitimate users sharing the time-frequency resources) and external (illegitimate users) users, since each user is only able to recover his signal using the corresponding physical channel parameters and a shared secret. This is important since NOMA allows users with stronger signals (near the base station) to decode and recover the signals of weaker users (far from the base station).

The proposed scheme can also be applied in the uplink case (against external users only), since NOMA users and the base station are able to extract the same channel-based parameters and they share a common secret, hence the base station will be able to successfully decode the signals of each user. However, the security of data in downlink NOMA is more critical and crucial (vulnerability against external and internal users). The proposed uplink signal becomes:

$$y_U(t) = \sum_{k=1}^{K} c_{U,k}(t) h_k + no_k(t). \tag{8.2}$$

At the receiver's side, the same steps are required for decryption but in a reversed order and using the inverse cipher primitives.

8.3 Cipher Primitive Update Process

For every new frame, the dynamic key is updated using the permutation table $Pbox$, which is a simple and low cost technique. Circular shifting depends on the produced PRS having a length NB_F. For example, if the i^{th} PRS value is equal to 3, then the cipher primitive of the i^{th} frame symbol is shifted 3 times.

Figure 8.2 represents the proposed structure of the update cipher primitive process, which relies on the channel-based dynamic key to create and update the needed cryptographic primitives. The encrypted data frame symbol (EFS) is derived by simply applying the proposed one-round, one-operation cipher scheme. The generated cipher primitive(s) are updated, frequently. Specifically, they are circularly shifted after each new frame symbol (FS), depending on the value of PRS. The error propagation effect is reduced since the encryption scheme is applied at the modulation symbol level, which prevents an erroneous symbol from affecting other modulation symbols.

The main advantage of the proposed update cipher primitive scheme is enabling parallel computing and parallel ciphering, which reduces the required latency and delay. In fact, one can pre-compute all of the updated cipher primitives and the dynamic keys ahead of time, according to the generated PRS and $Pbox$. Moreover, the update cipher primitive operation increases the robustness and security level of the solution against different types of attacks.

For the CD-NOMA case, the scheme presented in [181] can be further enhanced and improved to ensure better security. In particular, instead of using the

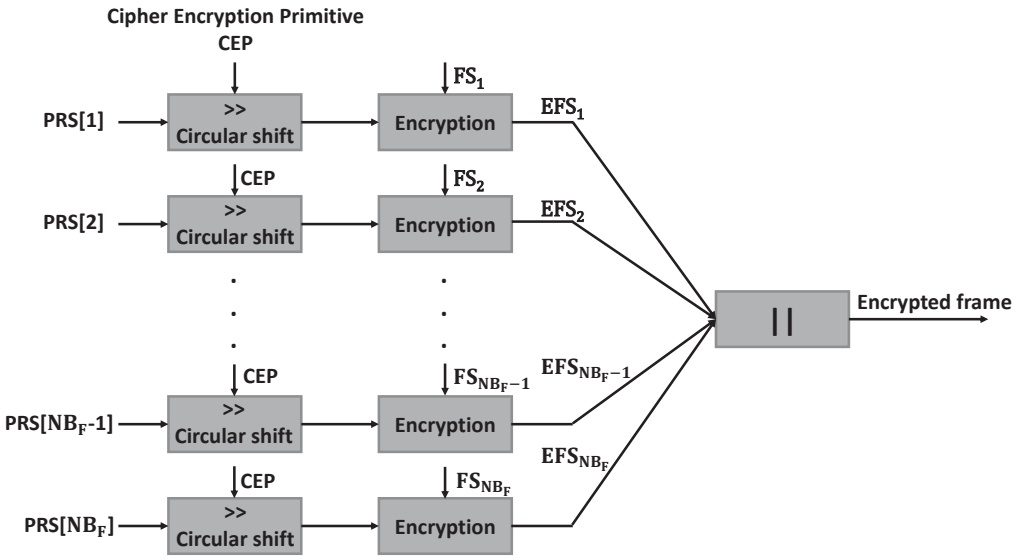


Figure 8.2: The proposed update process of the cipher primitives for NOMA systems

CSI of each user to pseudo-randomly rotate the base constellations (used to construct the SCMA codebooks) (phase rotation), one can utilize the channel-based dynamic key DK, to increase the security of the utilized phases. Consequently, illegitimate users will not be able to know the used phase angles, which enhances the security level of the utilized codebook and the robustness of the overall SCMA process.

8.4 Security Analysis of the NOMA-based Cipher Scheme

In this section, the security level of the proposed NOMA-based scheme is tested and evaluated based on several metrics. Similar to the generalized scheme, four cipher schemes are considered, and which are: the permutation scheme, the 2-D permutation scheme, the phase encryption scheme and the enhanced phase encryption scheme.

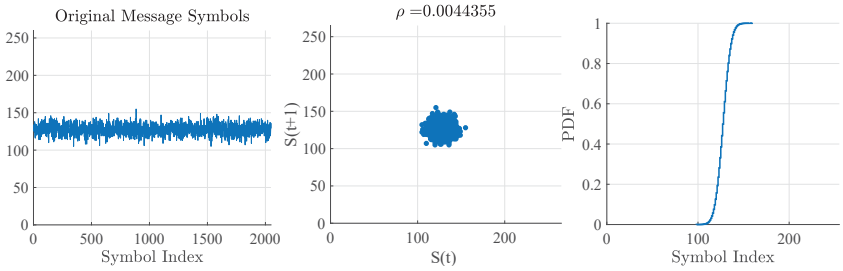
8.4.1 Uniformity

The PDF of the plaintext and ciphertext frames are illustrated in Figures 8.3c and 8.3f, when using the NOMA-based dynamic permutation operation as an encryption scheme. The obtained ciphertext clearly satisfies the required uniform

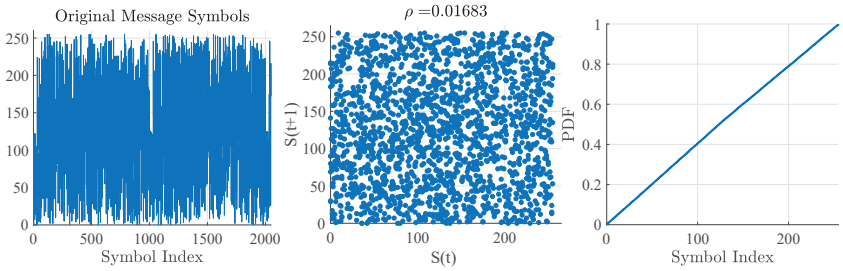
distribution, thus, ensuring the desired randomness level. Similar results are obtained for phase encryption and enhanced phase encryption schemes.

8.4.2 Recurrence Test

Figure 8.3b shows that the original data has a normal distribution. Whereas, Fig. 8.3e shows the recurrence plot of the obtained encrypted message using the NOMA-based permutation scheme (permuted modulation symbols). Results confirm that encrypted symbols are randomly scattered and distributed, and that a good randomness level is attained. Moreover, as seen in Fig. 8.3a, the original frame symbol has a fixed amplitude range, while Fig. 8.3d shows that the encrypted symbols have random and varying amplitudes. Finally, similar results were obtained for both, the phase encryption and phase-amplitude encryption schemes.



(a) Symbol amplitude of original data (b) Recurrence of original data (c) ECDF of original data



(d) Symbol amplitude of permuted data (e) Recurrence of permuted data (f) ECDF of permuted data

Figure 8.3: Symbol amplitude of (a) original and (d) permuted data. Recurrence of (b) original and (e) permuted data. ECDF of (c) original and (f) permuted data, using the NOMA-based cipher scheme

8.4.3 Independence

Figures 8.4a and 8.4b correspond to the average difference between the original and encrypted frames when varying the number of bits per symbol and the number of symbols per frame, for 1,000 iterations, respectively. Figure 8.4c represents the Effective Cumulative Density Function for 1,000 frames symbols. As it can be depicted in Fig. 8.4, all NOMA-based encryption schemes have a difference value close to 0.5 for 1,000 transmitted frames. Moreover, the correlation value is close to the ideal value, 0, for all NOMA-based encryption schemes (permutation, phase encryption ("AB" scheme) and enhanced phase encryption schemes ("ABC" scheme)) based on three cases: 1) bits per symbol (Fig. 8.5a), 2) symbols per frame (Fig. 8.5b), 3) and number of transmitted frames (Fig. 8.5c). This validates that the proposed NOMA-based solution is immune against statistical attacks.

8.4.4 Key Sensitivity

Figure 8.6 shows the results of key sensitivity for 1,000 iterations using the NOMA-based cipher approach. For all simulation cases, the key sensitivity values are very close to the desired value, 0.5 (the number bits per symbol (Fig. 8.6a), the number of symbols per frame (Fig. 8.6b) or the number of transmitted frames (Fig. 8.6c)). Hence, the proposed schemes is able to resist linear and differential attacks.

8.5 Performance Evaluation of the NOMA-based Cipher Scheme

In this section, the performance of the proposed solution is assessed in terms of BER and error propagation.

8.5.1 Error Propagation

The error propagation analysis of the generalized scheme is also true for the NOMA-based cipher approach (Fig. 8.7). Specifically, the proposed scheme does not degrade the BER performance of the NOMA system since the plotted curves overlap (matched curves).

8.5.2 Execution Time

The same analysis of the generalized scheme applies for the NOMA-based approach (Fig. 8.8).

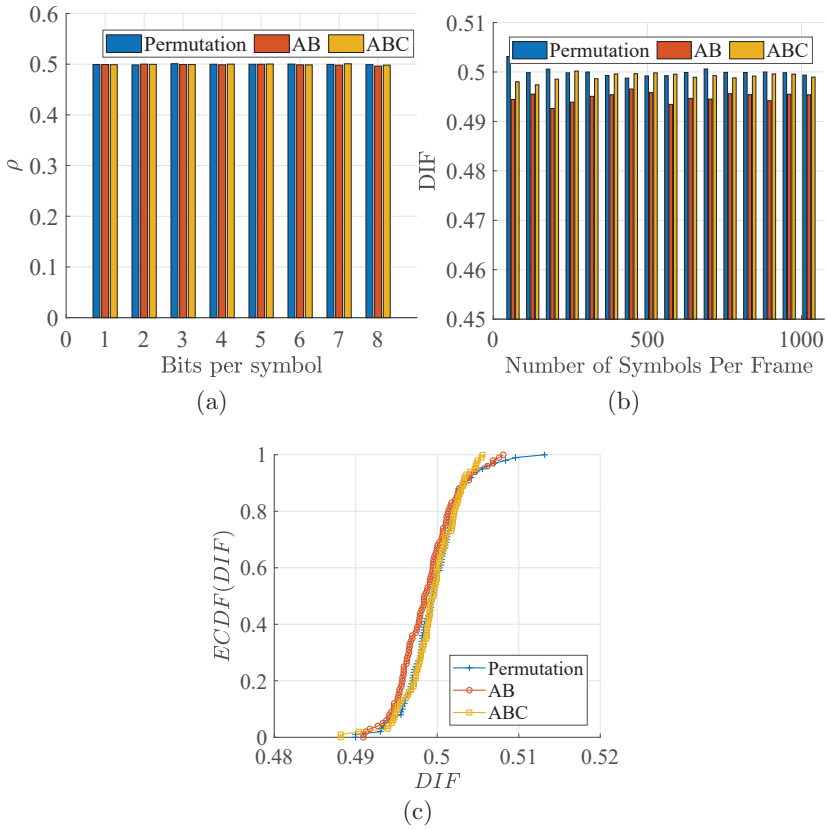


Figure 8.4: Independence test to measure the difference between plaintext and ciphertext using three different NOMA-based cipher schemes, versus (a) the number of bits per modulation symbol, (b) the number of modulation symbols per frame symbol, and (c) the corresponding effective cumulative density function for 1000 frame symbols

Figures 7.10c and 8.8c show the execution-time overhead of the encryption process with respect to modulation and channel coding. The overhead introduced by the generalized scheme is larger than that of the NOMA-based cipher approach for all encryption schemes (permutation, "AB" and "ABC"), since the update process of the NOMA-based cipher approach (circular shifting based on PRS) is simpler than that of the generalized scheme (a PRS is used to choose between two update permutation tables). For example, for a frame symbol size of $1,024$, the encryption overhead of the permutation scheme, the "AB" scheme and the "ABC" scheme is 23%, 40% and 75% when applying the generalized cipher approach, and 11%, 30% and 45% when utilizing the NOMA-based approach.

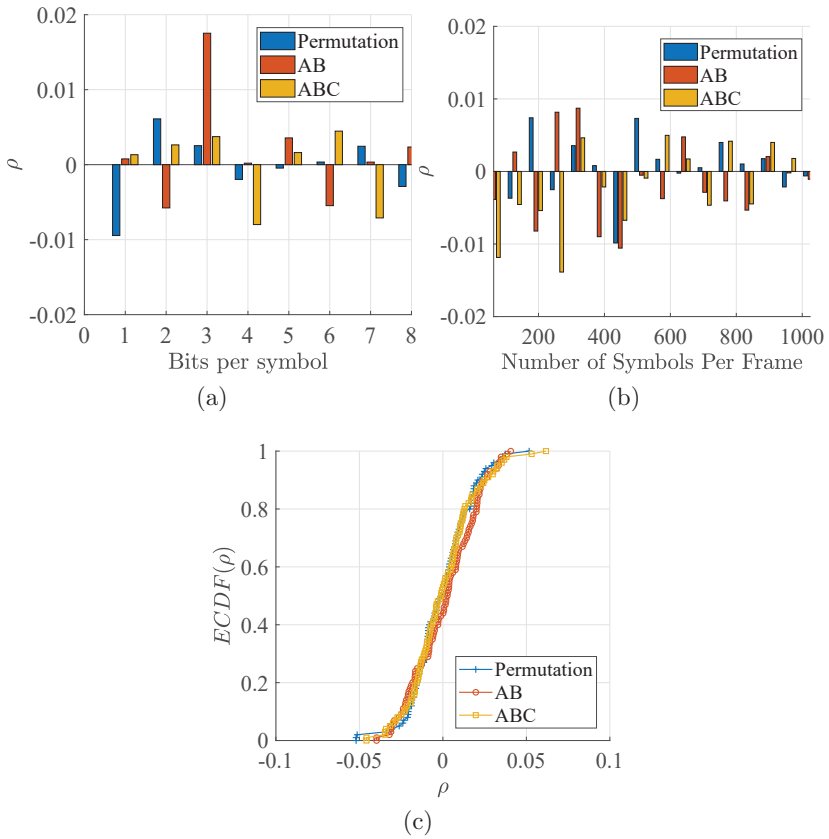


Figure 8.5: Results of the correlation coefficients between the plaintext and ciphertext using three different NOMA-based cipher techniques versus three cases: (a) the number of bits per modulation symbol, (b) the number of modulation symbols per frame, and (c) the number of frame symbols transmitted

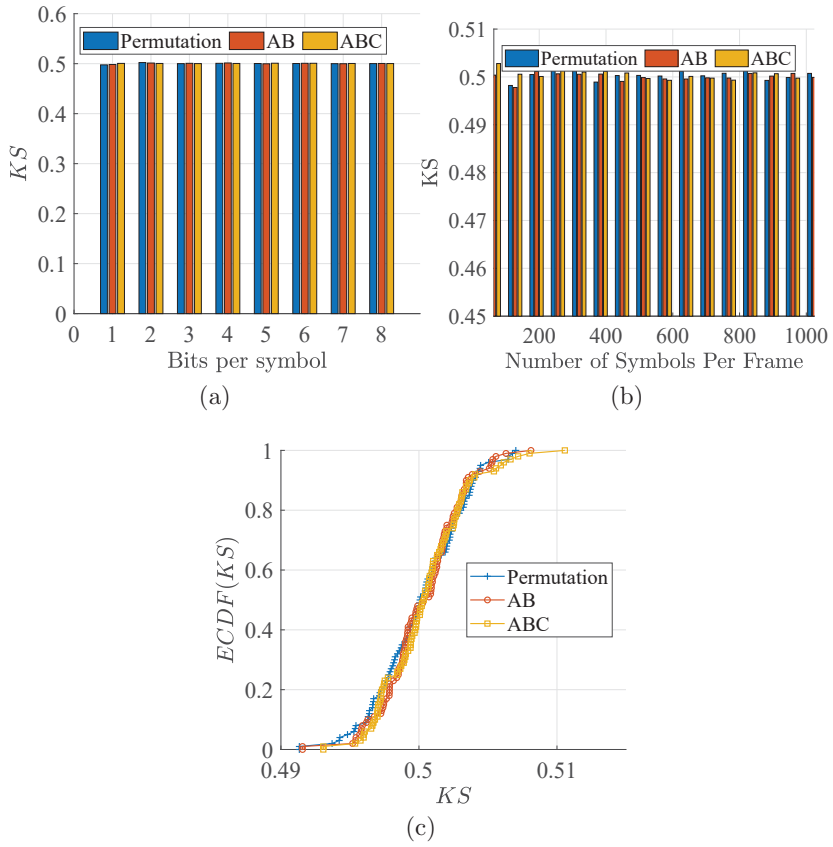


Figure 8.6: Results of key sensitivity versus (a) number of bits per symbol, (b) number of modulation symbols per frame, and (c) number of frames symbols transmitted, using NOMA-based cipher approach

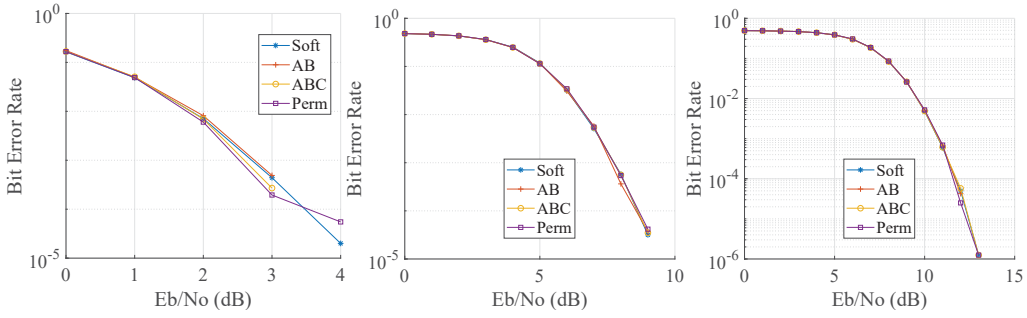
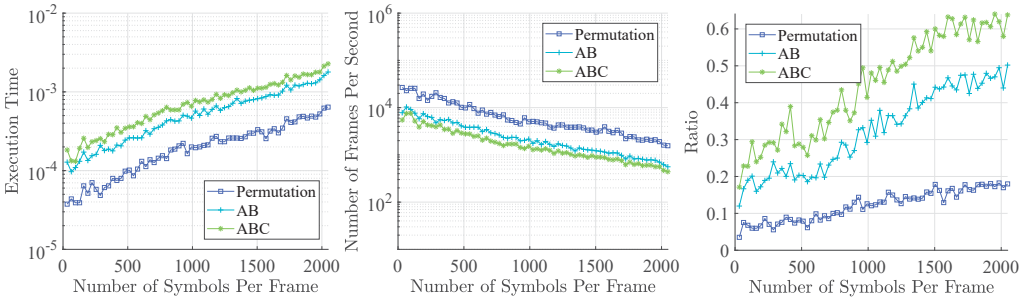


Figure 8.7: The BER performance of the NOMA-based cipher approach using different encryption schemes versus E_b/N_0 using three modulations schemes: (a) QPSK, (b) 64-QAM and (c) 256-QAM modulation



(a) Execution Time (sec) (b) Number of encrypted symbols per second (c) Overhead ratio

Figure 8.8: (a) The required execution time (sec) for the NOMA-based encryption schemes. (b) The number of encrypted frame symbols versus frame symbol size ($\log 2$), and (c) the ratio of execution time overhead (%) for different frame symbol lengths and using different NOMA-based cipher schemes

Chapter 9

Data Confidentiality for MIMO-based IoT Systems

In this chapter, two simple and lightweight PLS solutions that target data confidentiality in MIMO-based IoT systems, are proposed and assessed. The first solution leverages the concept of encoding, while the second solution takes into account the OFDM frame symbol length, which can be fixed or varied from one antenna to another.

9.1 MIMO System Model: Spatial Multiplexing

For transmitting data across a given channel, the MIMO encoder uses one of two available space time processing techniques, which are: space-time coding and spatial multiplexing [268]. The first technique ensures a maximum diversity gain, where multiple copies of the same information are sent across independent fading channels to combat fading and enhance system reliability. In contrast, spatial multiplexing provides a high multiplexing gain (Degrees of Freedom (DoF)), since each spatial channel carries independent (different) information, thereby increasing the data rate [269]. The main advantage of this method is leveraging the multiplexing gain, which is equal to $DoF = min(nt, nr)$. Here, no explicit orthogonality is needed, in contrast to space-time block coding [270]. Spatial multiplexing requires powerful decoding techniques at the receiver [270]. According to [271, 272], the Maximum-Likelihood (ML) optimum receiver results in a good performance, but suffers from receiver complexity, which grows exponentially.

In both of the proposed schemes, spatial multiplexing is considered where the data stream is divided into multiple independent sub-streams. Each sub-stream is separately transmitted using one of the available transmit antennas. The transmitter and receiver are assumed to have nt and nr antennas, respectively. For maximum performance and throughput, it is assumed that $nt = nr$.

9.2 The First Proposed Cipher Solution: Generic MIMO Systems

The generated dynamic key is divided into 5 sub-keys, each utilized for a specific objective.

1. **The first sub-key, DSK_1:** it has a size of 128 bits, and it is utilized for the generation of a group of invertible matrices, G, which includes γ encryption matrices: $G = G_1, G_2, \ldots, G_\gamma$, each having a size of $nt \times nt$ and complex values between $]0, 1]$. First, DSK_1 is used to generate a sequence (key-stream) of bytes including $2 \times \gamma \times nt \times nt$ elements. Next, the values of these elements are mapped to the interval: $]0, 1]$. For example, if the produced sequence (key-stream) is in byte representation, it can be converted by adding 1 and then, dividing each byte by 256. Afterwards, the produced vector is divided into two sub-vectors: the first one includes the real values while the second one includes the imaginary values. Both sub-vectors have $\gamma \times nt \times nt$ elements. In order to obtain the γ complex-valued matrices $(G_1, G_2, \ldots, G_\gamma)$, one element from the first sub-vector and one element from the second sub-vector are combined to produce one complex element in the diffusion matrix. This process is repeated until all nt^2 complex elements in the diffusion matrices are formed (complex representation). The produced complex values are reshaped to form a diffusion matrix of size $nt \times nt$. Each matrix will be multiplied by a different matrix of modulations symbols, which has a size of $nt \times \phi$ in order to attain data confidentiality.

2. **The second sub-key, DSK_2:** it also has a length of 128 bits, and it is used to derive a matrix selection table (SG). This table contains a randomly permuted sequence of repeated numbers ranging between 1 and γ and it has a length of $1 \times NB_{BL}$, where NB_{BL} is the total number of data sub-frames/blocks in each frame ($\gamma \leq NB_{BL}$).

3. **The third sub-key, DSK_3:** with a size of 128 bits, it is used to generate a link selection table (SL) of size $1 \times nt$, which has values between 1 and nt (randomly shuffled values). Every row of the ciphered data matrix is randomly transmitted on a specific antenna using the corresponding value in SL.

4. **The final two sub-keys, DSK_4 and DSK_5:** these sub-keys have a size of 64 bits each, and they are used to generate $Pbox_{SG}$ and $Pbox_{SL}$ (permutation tables), which are used to permute/update SG and SL, respectively, for every input frame. The Modified key-Scheduling Algorithm (M-KSA) of RC4 [262] is used to derive SG, SL, $Pbox_{SG}$ and $Pbox_{SL}$ (Fig. 9.1).

Each frame on every antenna has Φ modulation symbols and it is further divided into NB_{BL} sub-frames, each with ϕ modulation symbols such that $NB_{BL} =$

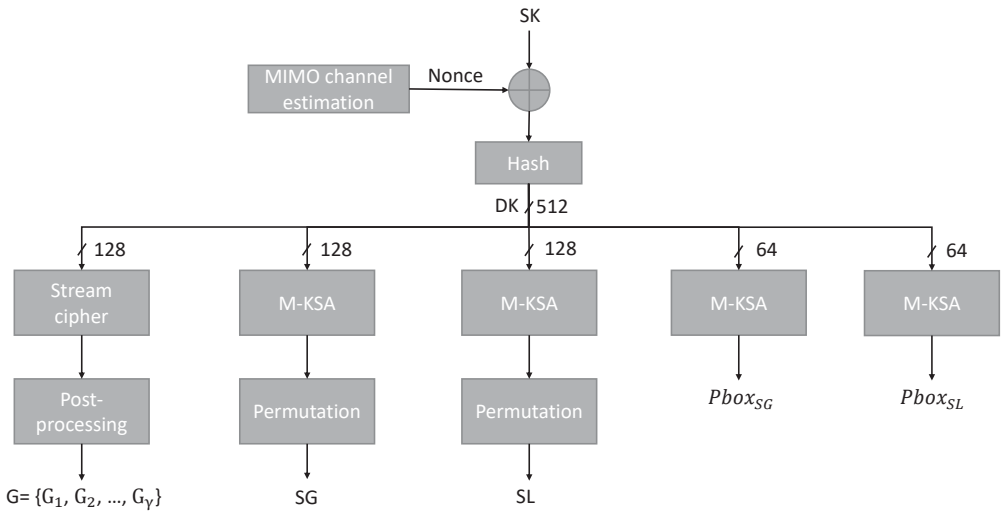


Figure 9.1: Proposed dynamic key generation procedure and ciphers primitives construction technique for MIMO systems

$\frac{\Phi}{\phi}$. The first sub-frame $(1 \times \phi)$ on every antenna are combined/concatenated to form an $nt \times \phi$ plaintext matrix, PM_i. The sequential process is repeated for all of the NB_{BL} sub-frames in each frame.

Next, the square diffusion matrix ($G_i = G[SG[i]]$) with a size of $nt \times nt$, is multiplied by its corresponding input plaintext matrix PM_i $(nt \times \phi)$. The matrix G_i is selected according to the corresponding value of the selection table, SG. The encrypted and diffused matrix CM_i $(nt \times \phi)$ represents the i^{th} encrypted matrix. Specifically, the encryption/diffusion process of the i^{th} plaintext matrix, PM_i, is described by Equation 9.1:

$$\begin{aligned} CM_i &= G_i \cdot PM_i \\ &= \begin{bmatrix} CM_{1,i} \\ CM_{2,i} \\ \vdots \\ CM_{nt,i} \end{bmatrix} = \begin{bmatrix} G^i_{1,1} & \cdots & G^i_{1,nt} \\ G^i_{2,1} & \cdots & G^i_{2,nt} \\ \vdots & \ddots & \vdots \\ G^i_{nt,1} & \cdots & G^i_{nt,nt} \end{bmatrix} \cdot \begin{bmatrix} PM_{1,i} \\ PM_{2,i} \\ \vdots \\ PM_{nt,i} \end{bmatrix}, \end{aligned} \tag{9.1}$$

where $PM_{j,i}$ is the i^{th} plaintext data sub-frame/block on the j^{th} antenna $(i = 1, \ldots, NB_{BL}$ and $j = 1, \ldots, nt)$. In addition, $CM_{j,i}$ is the i^{th} encrypted block on the j^{th} antenna, and G^i is an invertible encryption matrix, which is selected according to the i^{th} entry in SG.

Afterwards, all encrypted blocks are concatenated at the end of encryption process. The obtained ciphertext is a matrix that has a size of $nt \times \Phi$. Each

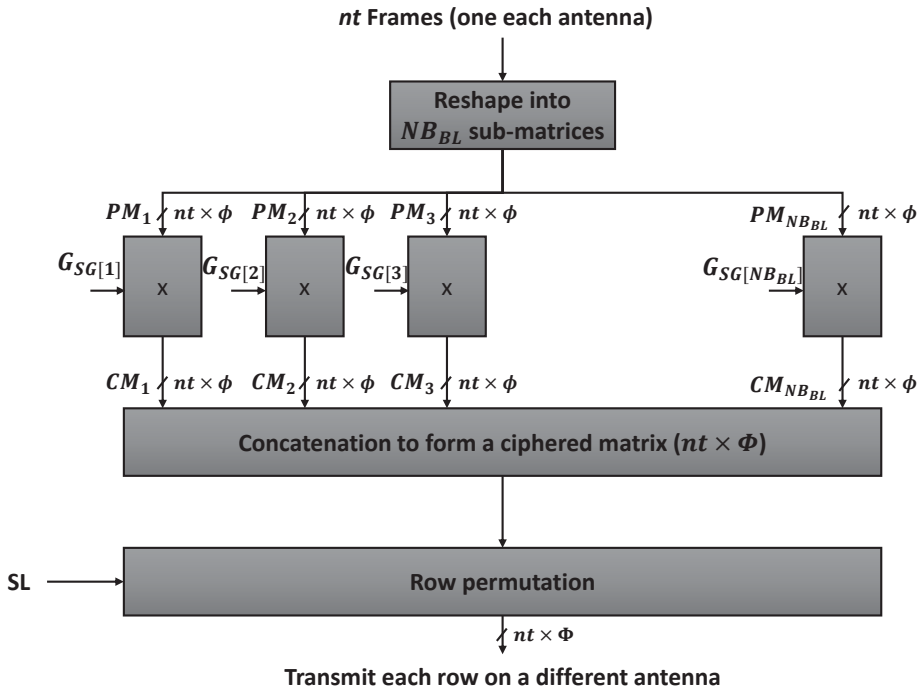


Figure 9.2: Proposed PLS MIMO cipher scheme

row is transmitted using one of the nt antennas. The selection of the antenna is based on SL (each encrypted frame is transmitted on a pseudo-randomly selected unique antenna). The proposed ciphering process is shown in Fig. 9.2.

For every input frame, the selection tables (SG and SL) are updated using the permutation tables, $Pbox_{SG}$ and $Pbox_{SL}$. This increases the dynamicity and randomness of the proposed solution, leading to a higher security level.

At the receiver's side, the same steps are followed to decrypt the received ciphertext, but in a reversed order and using the inverse diffusion matrices and selections tables, SL^{-1} and SG^{-1}. Since the generated encryption matrices are invertible, and the selection tables are bijective, the receiver will successfully construct the inverse of these cipher primitives and consequently, recover the received data.

9.3 The Second Proposed Cipher Solution: MIMO-OFDM Systems

The proposed solution, in this section, consists of two variants and it focuses on the security of data in OFDM-based MIMO systems. The first cipher variant is used when OFDM frame symbols have different lengths among different antennas,

whereas the second variant is used when all OFDM frame symbols have a common and fixed length.

9.3.1 Proposed Key Derivation Scheme

First, the transmitter and receiver estimate the wireless channel between them. Then, they extract unique features and properties that are specific to their physical channel. Here, it should be noted that there exists several wireless channels between two MIMO users, when employing spatial multiplexing. In this case, users can estimate all of the available wireless channels, extract the channel properties of multiple channels, and then combine them to enhance security even further. From these properties, both users are able to derive the same channel-based nonce, independently. In case of channel non-reciprocity, reconciliation is applied. Any user outside the proximity of the legitimate users, is not able to derive the same channel-based nonce since the channel is viewed differently (different properties). However, this can not be generalized. In particular, several works in the literature have proven that adversaries can derive the same channel properties as the authorized users, when having sufficient resources and power [260, 27]. Consequently, one should not rely on channel information, only, to secure a communication session.

In order to overcome this limitation, the pre-shared secret key is combined (Exclusive-OR (XOR)) with the channel nonce (pseudo-random and dynamic), to obtain a dynamic key. The resulting key will be used to secure transmitted data in MIMO-OFDM systems. The SHA-512 hash algorithm is used, following the XOR operation, in order to make sure that the resulting key is irreversible (one-way property). From the 512-bit dynamic key, several sub-keys can be produced. Since the proposed solution considers two cases: 1) frame symbols having different lengths on different antennas ($NB_{FS_1}, NB_{FS_2}, NB_{FS_3}, \ldots, NB_{FS_{nt}}$) and 2) frame symbols having the same length (NB_{FS}), two sub-key derivation functions are proposed (cipher primitive derivation functions).

Case 1: Frame Symbols of Different Lengths

In the first case, the frame symbols on each antenna branch are assumed to have different lengths, whereas the second case assumes that all frame symbols have the same length, NB_{FS}. Both of these cases are real-case scenarios in existing communication systems. The first case is more challenging than the second case, hence, a simple cipher scheme with simple cipher primitives is proposed. Specifically, the 512-bit dynamic key is split into two sub-keys (256 bits each).

- The first sub-key is used to generate a permutation vector (table), $Pbox_{nt}$, of length equal to $nt \times 1$ and random values between 1 and nt (number of antennas). This vector is used to permute the frame symbols on each antenna in a pseudo-random manner.

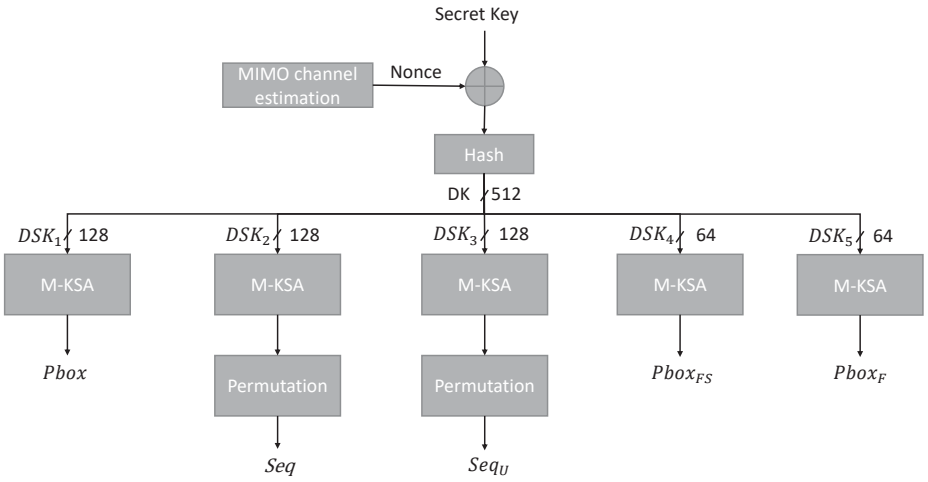


Figure 9.3: The derivation procedure of the dynamic key and the construction technique of the cipher primitives for the proposed second variant (case 2)

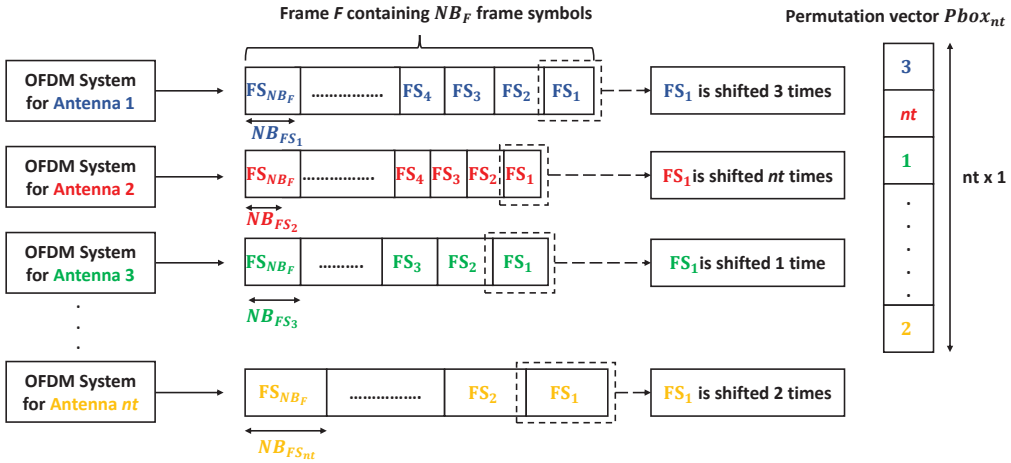


Figure 9.4: Proposed data confidentiality scheme: Case 1

- The second sub-key is used to produce another permutation table, $Pbox_{update}$. However, this table is used to permute/update the first permutation table for every new set of frame symbols (both permutation tables having the same size).

Case 2: Frame Symbols of the Same Length

As shown in Figure 9.3, the obtained dynamic key in the second case is divided into five sub-keys:

- **First sub-key:** DSK_1 (128 bits) is used to generate a permutation table

Pbox. It contains $nt \times NB_{FS}$ elements that have randomized values between 1 and $nt \times NB_{FS}$. This table is used to permute/shuffle the 2-D data matrices, having dimensions equal to $nt \times NB_{FS}$ (nt columns and NB_{FS} rows), at the modulation symbol level. Here, nt is the number of antennas at the transmitter's side and NB_{FS} is the number of modulation symbols in one OFDM frame symbol (size of one frame symbol).

- **Second sub-key:** DSK_2 (128 bits) is used to derive a sequence, Seq, having a length equal to $2 \times NB_{FS}$ and pseudo-random values of -1 and 1. Seq is used to perform masking (phase shuffling) on each frame symbol (NB_{FS} modulation symbols), independently. Moreover, Seq is divided into two vectors, Seq_1 and Seq_2, each having a length of NB_{FS}. The first sequence Seq_1 modifies the phases of the real components in complex symbols. Whereas Seq_2 modifies the phases of the imaginary components. In other words, the phase of the real component and the phase of the imaginary component of every modulation symbol are modified based on one element in Seq_1 and one element in Seq_2, respectively. Hence, $2 \times NB_{FS}$ values are needed so that each component in FS (NB_{FS} complex modulation symbols) is modified based on a unique value in Seq_1 and Seq_2. Note that phase modification refers to modifying the $+$ and $-$ signs of the complex symbols. If the value in Seq_1 or Seq_2 is 1, the phase of the component remains the same. In contrast, when the value is negative, the phase of the corresponding component is flipped.

- **Third sub-key:** DSK_3 (128 bits) is used produce a sequence Seq_U, that includes $2 \times NB_{FS}$ elements having random values between 1 and $2 \times NB_{FS}$. This sequence is used to randomly shift Seq for every set of nt frame symbols (circular shift).

- **Forth sub-key:** DSK_4 (64 bits) is used to generate a permutation table, $Pbox_{FS}$, to permute *Pbox* for every set of nt frame symbols.

- **Fifth sub-key:** DSK_5 (64 bits) is used to generate a permutation table, $Pbox_F$, to permute *Pbox* for every frame.

From the channel-based dynamic key, simple and lightweight cipher primitives are obtained. These primitives are used to ensure data confidentiality for MIMO-OFDM systems, which will be thoroughly discussed in the following.

9.3.2 Proposed Data Confidentiality Scheme

The proposed data confidentiality scheme consists of two cipher variants. The first cipher variant assumes that frame symbols on different antennas, vary in

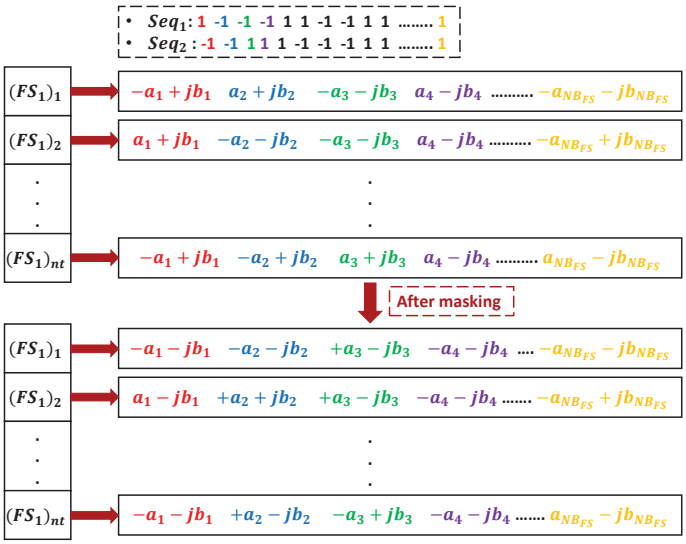


Figure 9.5: Proposed masking operation

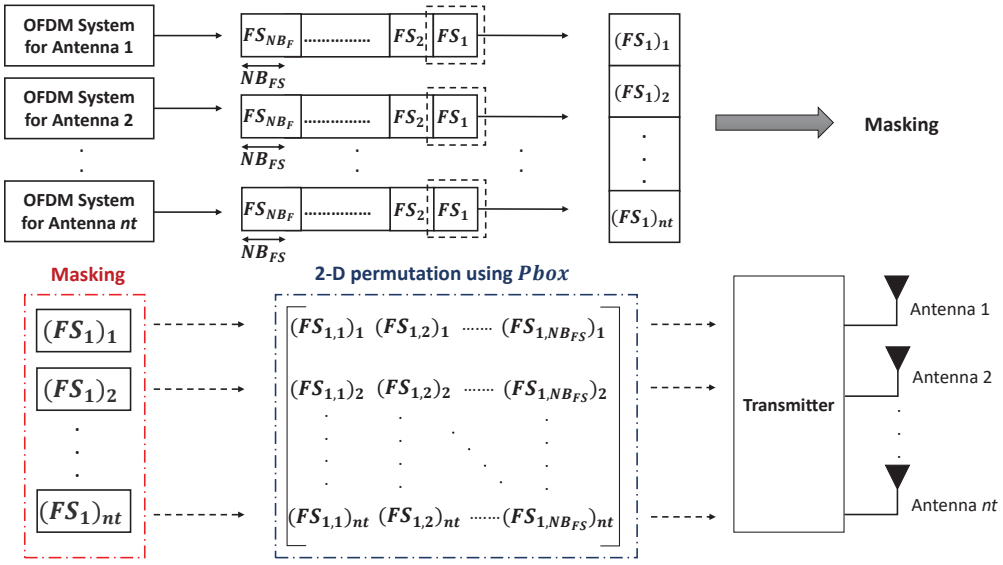


Figure 9.6: Proposed data confidentiality scheme: Case 2

terms of length. On the other hand, the second variant assumes that all frame symbols contain the same number of modulation symbols. Both schemes are applied after the IFFT transformation (time-domain OFDM frame symbols).

Case 1: Frame Symbols of Different Lengths

The first variant of the proposed solution considers a time-domain frame (F) (af-

ter IFFT) that includes NB_F OFDM frame symbols. The length of these frame symbols varies from one antenna to another ($NB_{FS_1}, NB_{FS_2}, NB_{FS_3}, \ldots, NB_{FS_{nt}}$ for antennas $1, 2, \ldots, nt$), as shown in Figure 9.4. The elements of each frame symbol on each antenna is randomly circularly shifted based on its corresponding value in $Pbox_{nt}$. The utilized permutation vector/table has a size of $nt \times 1$ and random values between 1 and nt. For clarification, a simple example is shown in Figure 9.4. For the second set of frame symbols (FS_2) on the available antennas, $Pbox_{nt}$ is first updated/permuted using $Pbox_{update}$, and then employed. This step is repeated NB_F times, that is for each new frame symbol in the frame. Consequently, this complicates the adversary's job and decreases the probability of successful attacks. At the receiver's side, the same steps are performed but using a reversed circular shift operation. It is assumed that $nt \leq NB_{FS_i}, i = 1, 2, \ldots, nt$.

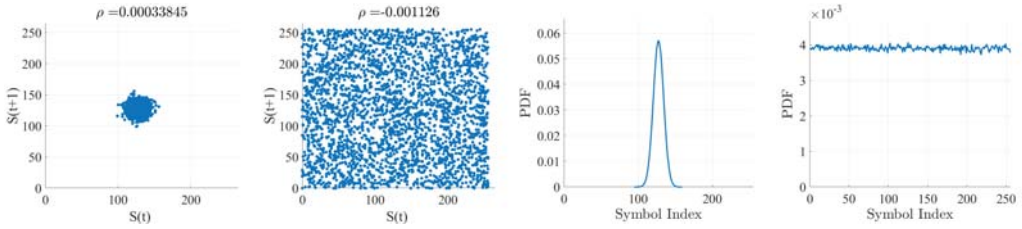
Case 2: Frame Symbols of the Same Length

The second cipher scheme (variant) is divided into two steps: masking and 2-D permutation.

Following the IFFT transformation (time-domain), each OFDM frame (F) will include NB_F OFDM frame symbols (FS), and each OFDM FS will include NB_{FS} complex modulation symbols. Similar to the first variant, this technique also performs ciphering at the modulation symbol level. More specifically, each frame symbol on each antenna branch is grouped with the rest of the frame symbols on other antennas, forming a set of $nt \times 1$ frame symbols ($nt < NB_F$). This scheme is repeated for all NB_F frame symbols in one frame. Here, every set of nt frame symbols is processed, independently.

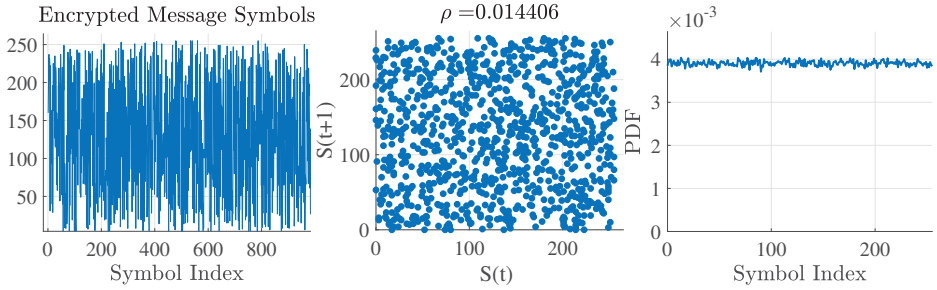
- **Masking:** This step handles each FS on each antenna (within the set of nt frame symbols), separately (Figure 9.5). In particular, the phases of the real and imaginary components of each complex modulation symbol are modified based on two values in Seq_1 and Seq_2, respectively. The sequences Seq_1 and Seq_2 have a length of NB_{FS} and contain pseudo-random values of 1 and -1. For the first modulation symbol in the frame symbol, the real component is multiplied by the first value in Seq_1 and the imaginary component is multiplied by the first value in Seq_2. Assuming that the first values in Seq_1 and Seq_2 are equal to -1 and 1, respectively, then a modulation symbol $a + jb$ will become $-a + jb$. This process is repeated NB_{FS} times, so that all of the modulation symbols within one FS are masked. For every new set of FS (on all antennas), Seq is updated using Seq_U (cyclic shift). Once all nt frame symbols are masked within one set, 2-D permutation is applied.

- **2-D permutation:** The resulting nt frames symbols are concatenated vertically (grouped) to compose a matrix, in such a way that the values of the first FS on the first antenna represent the first row in the data matrix,



(a) Recurrence of original data (b) Recurrence of encrypted data (c) PDF of original data (d) PDF of encrypted data

Figure 9.7: Assessment of the original and encrypted messages in terms of recurrence and PDF using the MIMO-OFDM PLS solution



(a) Symbol amplitude (b) Recurrence (c) PDF

Figure 9.8: Symbol amplitude, recurrence and PDF of the corresponding encrypted frames using the proposed MIMO-based cipher scheme

the values of the first FS on the second antenna represent the second row, and so on. Consequently, the obtained 2-D matrix will have a dimension equal to $nt \times NB_{FS}$ (nt rows and NB_{FS} columns). Next, the permutation table, $Pbox$, is reshaped and transformed to an $nt \times NB_{FS}$ matrix, which is used to randomly permute the 2-D data matrix. The ciphered matrix which results from the permutation operation, also has nt rows and NB_{FS} columns. For each subsequent group of nt frame symbols, $Pbox$ is updated using $Pbox_{FS}$. The permutation table, $Pbox$, is also updated using $Pbox_F$ for every new frame.

After the encryption process, each row in the ciphered matrix will be transmitted on a different antenna (spatial multiplexing), concurrently. The proposed scheme is shown in Figure 9.6.

At the receiver's side, the same steps are performed to decrypt and recover the transmitted data, but, in a reversed order. In addition, inverse cipher primitives are used, which are $Pbox^{-1}$, $Pbox_{FS}^{-1}$ and $Pbox_F^{-1}$.

9.4 Security Analysis

In the following, the security level of the two proposed cipher schemes is analyzed and assessed. For this evaluation, several security metrics and tests are considered, which are the randomness degree, recurrence, uniformity and sensitivity.

9.4.1 Randomness Degree

The security level of a specific cipher scheme is strongly related to the randomness degree of the encrypted frame symbols. Consequently, the randomness of the resulting ciphertext is evaluated using three different tests, which are uniformity, recurrence and independence.

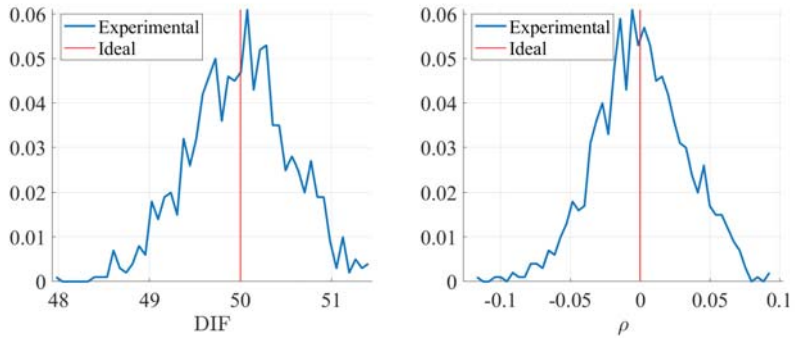
Uniformity: To assess the uniformity property, a simple and straightforward method have been used and which is the PDF. First, normally distributed data have been generated with a very low randomness degree as depicted in Figure 9.7c. In particular, the original data has most of its values centered around a specific value. Next, the original data has been encrypted using the proposed OFDM-based solution and its distribution has been plotted. From Figure 9.7d, it is clear that the obtained ciphertext has a uniform distribution, where all values have an equal probability of occurrence. This is also true for the first proposed scheme, as shown in Fig. 9.8c.

Recurrence: Figures 9.7a, 9.7b and 9.8b represent the recurrence plots of the original and encrypted data (using the first and second schemes), respectively. Unlike the original data which lacks randomness (recurrence points are grouped in one region), the encrypted data has a highly scattered recurrence plot. Therefore, this proves that the proposed schemes achieve the required security level.

Independence: The independence property measures the difference probability between the original frame symbol and its corresponding encrypted version at the bit level. This value should always be close to 0.5 or 50%. Indeed, the obtained results in Figures 9.9a and 9.10a prove that the proposed cipher solutions attain the independence property, in which the majority of the difference values are equal to the ideal value, 50%.

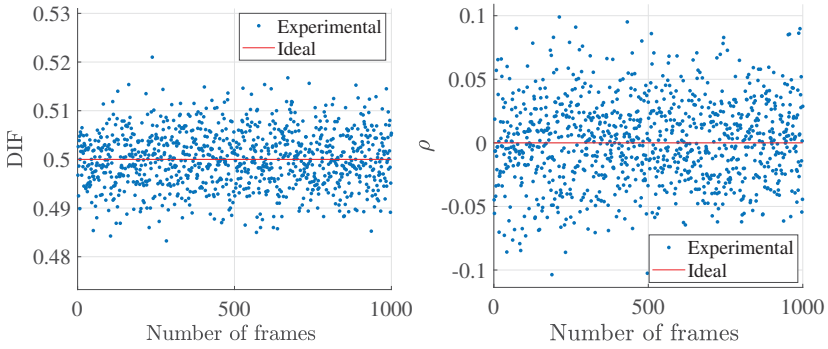
The correlation between the plaintext and ciphertext has also been calculated. As shown in Figures 9.9b and 9.10b, the correlation coefficients have a normal distribution centered around the desired value, 0.

This, in turn, confirms that the proposed solutions are immune to statistical attacks.



(a) Difference (b) Correlation

Figure 9.9: Independence test: (a) Difference measurements versus the number of transmitted frames. (b) Correlation coefficients between the plaintext and ciphertext (OFDM-based scheme)



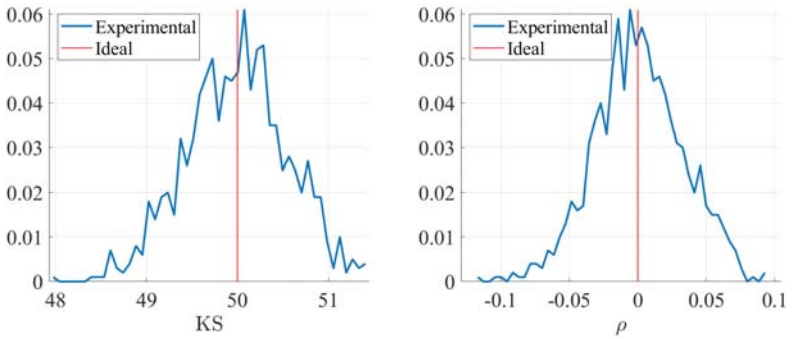
(a) Difference (b) Correlation

Figure 9.10: Independence test: (a) Difference measurements versus the number of transmitted frames. (b) Correlation coefficients between the plaintext and ciphertext using the MIMO-based scheme

9.4.2 Key Sensitivity

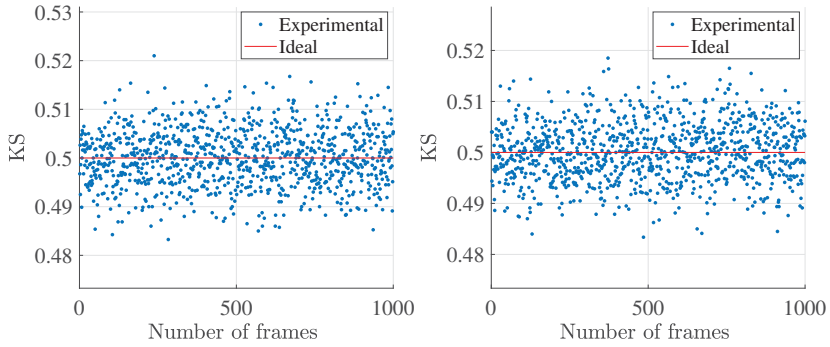
This test measures the bit-difference in encrypted frame symbols when using slightly different keys (one-bit change). The difference value between the ciphered frames should always be near 0.5 or 50% (at the bit level).

Figures 9.11a and 9.12a prove that the key sensitivity property is achieved using both of the proposed solutions, since the key sensitivity values are close to 50%. Also, Figures 9.11b and 9.12b show that different nonce values are independent and uncorrelated. Hence, linear and differential attacks are mitigated.



(a) Secret key sensitivity (b) Correlation of nonce

Figure 9.11: (a) The sensitivity measurements of the secret key versus the number of transmitted frames. (b) Correlation coefficient between the different nonce values (OFDM-based scheme)

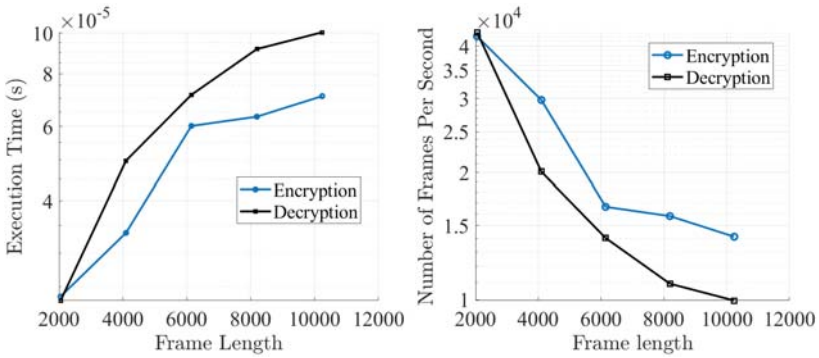


(a) Secret key sensitivity (b) Nonce sensitivity

Figure 9.12: The (a) secret key and (b) nonce sensitivity measurements versus the number of transmitted frames using the MIMO-based scheme

9.5 Performance Analysis

In this section, the proposed cipher solutions are evaluated in terms of error propagation and execution time. These criteria are crucial for assessing the performance and efficiency of any cipher approach. Generally, an efficient PLS-based cipher scheme should have low latency/delay (execution time) and low error propagation. For the proposed schemes, this can be easily verified since a one-round cipher structure that consists of simple operations is employed.



(a) (b)

Figure 9.13: (a) Execution time in seconds versus frame size using the proposed OFDM-based cipher scheme on one antenna. (b) Number of encrypted frame versus frame length

9.5.1 Error Propagation

Both of the proposed schemes require a single round and depend on simple operations to secure the data conveyed in MIMO systems. In particular, the proposed ciphering processes are applied at the lowest layer, which is the physical layer, using random and dynamic parameters extracted from the physical channel itself. Using these parameters and a secret key, users generate a dynamic key and derive the needed cryptographic primitives. Both schemes mainly depend on permutation which is a simple, efficient and robust security technique. This scheme avoids the diffusion operation which one of the main causes of error propagation in communication systems. Moreover, the shuffling/permutation operation does not result in the spread of error among bits, where error is restricted to one position, only. Consequently, the proposed schemes do not affect the bit error rate, and do not cause degradation in performance. This makes them good security candidates for emerging communication systems and technologies. It should also be noted that the proposed schemes are applied at the frame symbol level, that is on modulation symbols, hence, error does not propagate to other frame symbols in the frame.

9.5.2 Execution Time

Figures 9.13a and 9.14a show the execution time in seconds versus the input frame size. This simulation is done using MATLAB, and for one antenna only. The execution time on other antennas is also similar since the same operations are carried out on all antennas. As it can be inferred, the proposed solutions are efficient since they introduce low overhead (low complexity) and acceptable

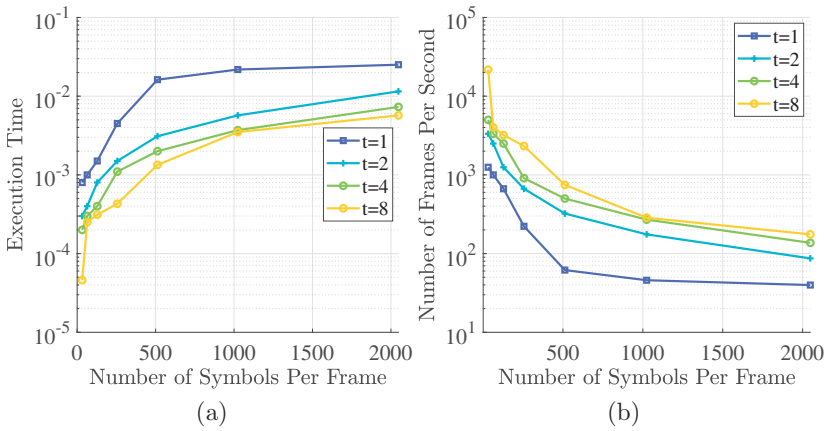


Figure 9.14: (a) Execution time in seconds. (b) Number of encrypted frame symbols versus frame symbol size ($\log_2$), using the MIMO-based scheme with ϕ =1, 2, 4 and 8

execution time. Figures 9.13b and 9.14b validates the results in Figures 9.13a and 9.14a since as the frame length increases, more time is required for execution and less encrypted frames are generated per second. From the presented figures, it is clear that the decryption process has a higher execution time than the encryption process. This is logical since the decryption process requires generating the inverse cryptographic primitives (such as inverse permutation tables), which introduces additional delay.

Chapter 10

Source Authentication and Message Integrity

Source authentication and message integrity, also referred to as message authentication, allow communicating users to validate the legitimacy and correctness of received messages. This security service proves that data has been generated by trusted entities and that it hasn't been manipulated while in transmission [27]. Traditional security mechanisms depend on upper-layer multi-round cryptographic primitives such as cryptographic hash functions to achieve source authentication and message integrity [273]. However, these techniques cannot be applied at the physical layer (computationally complex and expensive) and, hence, new alternatives should be studied and proposed.

In this chapter, two lightweight and secure PLS message authentication algorithms (keyed hash functions) are proposed for 5G communication systems, mainly the 5G IoT systems (for consistency purposes). The first algorithm is a generic keyed hash function that applies to all end-to-end communications (any multiple access scheme), whereas the second targets 5G IoT systems utilizing OFDM. Both schemes prevent existing source authentication and message integrity attacks at the physical layer. It should be noted that source authentication and message integrity is realized following the data confidentiality step, that is after performing encryption.

10.1 A Generic Message Authentication Algorithm

The first algorithm is based on two main factors, the random physical properties of wireless channels and a secret session key. Both are combined together to generate the cipher primitives used in the authentication process. The proposed keyed hash function is applied on post-modulation symbols in their complex format. Moreover, it is a dynamic structure whereby the cryptographic primitives

are updated for every frame symbol, in a lightweight manner, which substantially increases the security level since it minimizes the risk of an exposed key. The generic solution is based on a non-linear integer function and on the Merkle-Dangard structure [274, 275], and it requires only one round for each block. It should be noted that physical layer parameters are introduced in the proposed solution to increase the randomness and dynamicity levels. The round function excludes any diffusion operation at the block level to avoid the associated high execution time. Additionally, the input to the proposed message authentication algorithm are modulation symbols in their complex format. Therefore, the proposed scheme is a generalized scheme that is independent of the modulation technique and the multiple access scheme. The proposed solution can be applied at the frame symbol level or at the frame level. For the first case, the Message Authentication Codes (MACs) are appended at the end of each frame symbol. Whereas in the second case, the last frame symbol is reserved for the generated MAC.

Recently, chaotic mapping has been proposed and studied for the construction of novel hash functions. However, existing and related schemes are considered inefficient and not practical since they depend on non-integer transformation. In order to overcome this issue, several chaotic maps have been reformulated into integer representation such as the integer finite skew tent transformation [276, 277], but this method is rarely used in the design of cryptographic solutions. In principle, the chaotic cryptographic scheme should involve integer operations as is the case in traditional cryptographic algorithms. This is necessary for reducing the costs (energy, memory, delay) of the required computation and conversion operations. As such, the generic hash function, which is proposed next, is based on the non-linear integer finite skew tent transformation to achieve an efficient and secure message authentication scheme at the physical layer.

10.1.1 Sub-key Generation

In order to ensure message authentication, DK is divided into three sub-keys, which have lengths equal to 128, 128, and 256 bits, respectively (Fig. 10.1). The first sub-key (128 bits) is used to generate the permutation table $Pbox$, by employing a modified version of the key setup algorithm of RC4, M-KSA, as explained in [278]. This table permutes the vector, vec_{MS}, which contains all of the possible complex modulation symbols (based on the modulation order), for every frame symbol. Similarly, the second sub-key (128 bits) is used to derive the second permutation table $Pbox_P$, which is used to permute and update the first permutation table, $Pbox$, for every new input frame. Finally, the third sub-key (256 bits) represents the initial authentication mixing key, DSK_I.

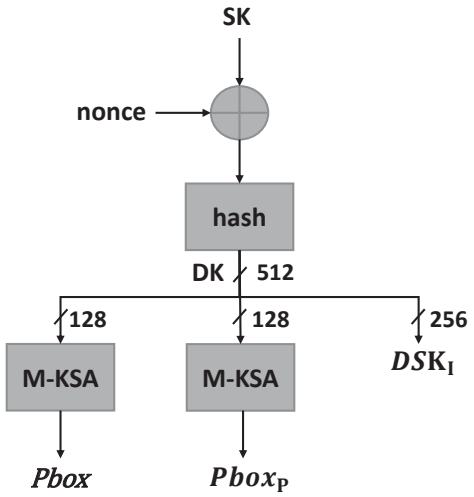


Figure 10.1: The proposed generic key generation scheme

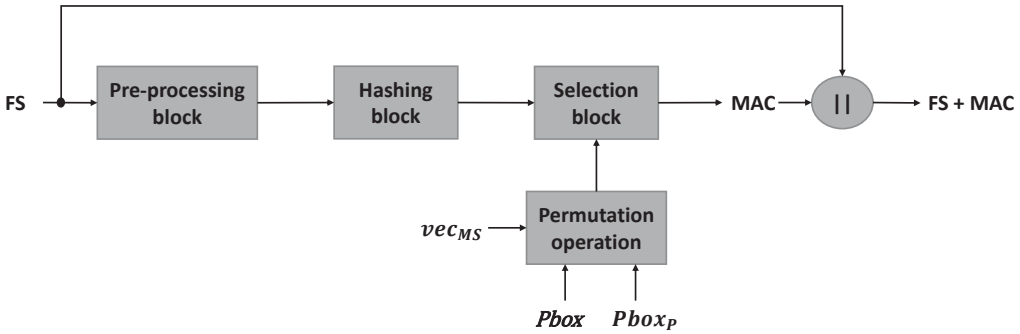


Figure 10.2: The proposed message authentication scheme at the physical layer

10.1.2 Proposed Generic Hash Function

The proposed hash scheme is divided into three main blocks (Fig. 10.2 and Algorithm 5), which are the pre-processing block, the hashing block, and the selection block which converts the obtained Message Authentication Code (MAC) integer value into complex representation (complex modulation symbols).

The Pre-Processing Block: After performing modulation (e.g modulation order M_O: $M_O = 2^{m_b}$ different modulation symbols, where m_b is the number of bits) and encryption, message authentication is applied at the frame symbol level. In particular, each frame is divided into NB_F frame symbols $F = FS_1||FS_2||\ldots||FS_{NB_F}$ and each frame symbol contains NB_{FS} complex modulation symbols.

In this step, one frame symbol enters the pre-processing block, at a time. The

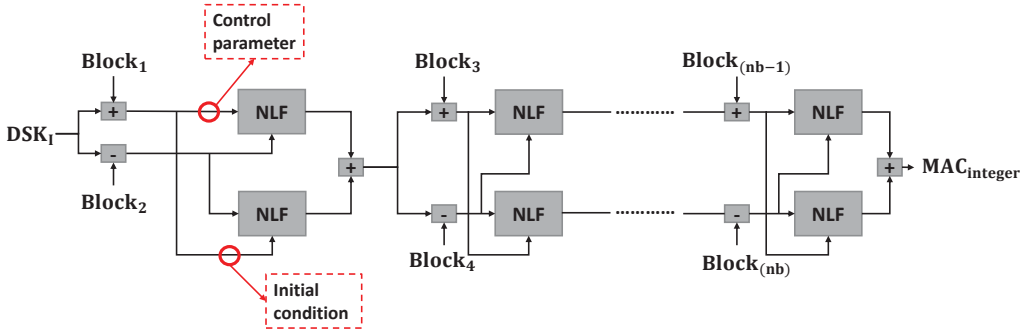


Figure 10.3: The proposed compression hashing function at the physical layer

complex modulation symbols in one frame symbol are separated into real and imaginary components, which have real values ranging between $-(log_2(M_O)-1)$ and $log_2(M_O)-1$. Then, these values are concatenated and shifted to positive integers by adding a value of $log_2(M_O)$ to all. Consequently, the resulting set will have values within the range of $\{1, 2(log_2(M_O))-1\}$.

Next, the obtained frame symbol vector, which consists of positive real values, is divided into $nb = \lceil \frac{2*NB_{FS}}{BL} \rceil$ blocks, each block having BL elements (Fig. 10.2). In case $\frac{2*NB_{FS}}{BL}$ is not an integer, padding is applied.

The Hashing Block: At this stage, every two blocks enter the non-linear function in a sequential manner to finally obtain the MAC value, which will be appended to the frame symbol itself, after $nb/2$ non-linear operations.

The detailed process is described as follows:

- The first two blocks of the frame symbol vector are combined with the initial mixing key, DSK_I. This key is added (modulo $2\sqrt{M_O}$) to the first block and subtracted (modulo $2\sqrt{M_O}$) from the second.
- The resulting blocks, then, enter two non-linear functions (NLFs) such that each block represents the control parameter in one function and the initial condition in the other, respectively (Fig. 10.3).
- The outputs of the two non-linear functions are added, then combined with the second pair of blocks (output added (modulo $2\sqrt{M_O}$) to the third block and subtracted (modulo $2\sqrt{M_O}$) from the forth block, in this case). The modified pair (modified third and forth blocks), again, enters two non-linear functions, where each block is the control parameter in one function and the initial condition in the other. Similar to the previous operation, the outputs here are also added and then combined with the third pair of blocks.
- This sequential process is repeated $nb/2$ times, until all of the nb blocks of one frame symbol are hashed. In case nb has an odd value, the proposed

non-linear transformation is repeated $(nb+1)/2$ times, and the final block $Block_{nb}$ is input twice to both non-linear functions after being added and subtracted to the output of the previous block.

- The output of the hashing operation is a block of size BL (elements).

Note that the utilized non-linear function takes as input two blocks (initial condition and control parameter) of size BL and containing values between 1 and $2(log_2(M_O))-1$, and outputs a block with a size equals to BL, containing integer values between 1 and $2(log_2(M_O))-1$. This output is referred to as the integer MAC, $MAC_{integer}$.

Algorithm 5 The proposed message authentication algorithm is applied on the i^{th} frame symbol (FS_i).

procedure ONE_ROUND_AUTHENTICATION(FS_i, DSK_I, vec_{MS}, Pbox)
  $temp \leftarrow DSK_I$
  $vec_{MS} \leftarrow vec_{MS}(Pbox)$
  for $ind = 1$ to NB_{FS} **do**
    $t[2 \times ind - 1] \leftarrow \mathsf{real}(FS_i[ind])$
    $t[2 \times ind] \leftarrow \mathsf{imag}(FS_i[ind])$
  end for
  $t \leftarrow Pre\_Processing(t,\ M_O)$
  for $it = 1$ to $\frac{2 \times NB_{FS}}{BL}$ **do**
    $Block_{\,it} \leftarrow t[(it-1) \times BL + 1 \rightarrow it \times BL]$
    $Block_{\,it+1} \leftarrow t[it \times BL + 1 \rightarrow (it+1) \times BL]$
    $O_1 \leftarrow NLF(temp + Block_{\,it},\ Block_{\,it+1})$
    $O_2 \leftarrow NLF(temp - Block_{\,it+1},\ Block_{\,it})$
    $temp \leftarrow O1 + O2$
  end for
  $MAC \leftarrow vec_{MS}(temp)$
  return MAC
end procedure

The Selection Block: In order to convert the output back to complex representation in an efficient way, a lightweight mapping operation is proposed. In particular, a vector vec_{MS} is initially constructed and it contains all the possible complex modulation symbols for a specific modulation order. For each frame symbol, this vector is permuted using the permutation table, $Pbox$. Finally, the MAC value, which will be appended to each frame symbol, is obtained using the following equation:

$$vec_{MS} = vec_{MS}(Pbox), \tag{10.1}$$

$$MAC = vec_{MS}(MAC_{integer}). \tag{10.2}$$

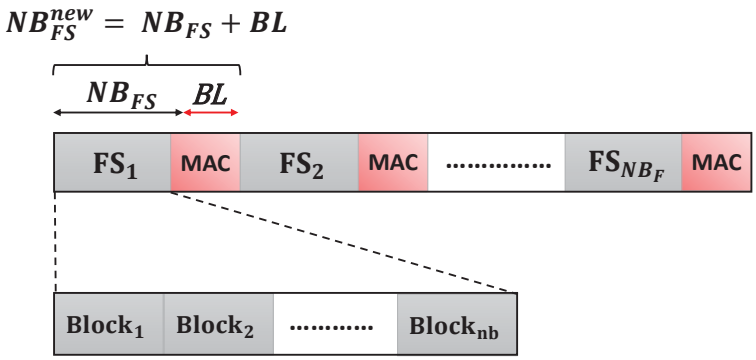


Figure 10.4: The first considered structure of one frame (first variant)

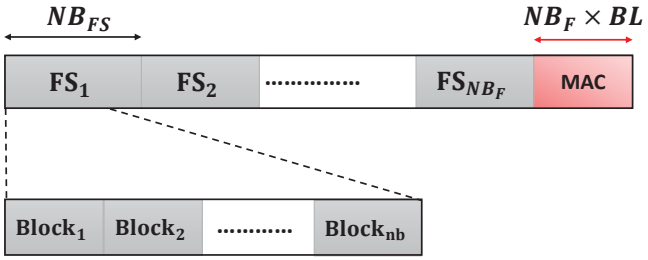


Figure 10.5: The second considered structure of one frame (second variant)

Note that for each input frame, the permutation table $Pbox$ is shuffled according to the second permutation table $Pbox_P$. The size of the final MAC value is BL. Hence, the size of the final frame symbol, which will be transmitted, is $NB_{FS}^{new} = NB_{FS} + BL$ (Fig. 10.4).

The overheard will be equal to $\frac{BL}{NB_{FS}^{new}}$. The value of BL is flexible and it depends on the application. In fact, as BL increases, more overhead is introduced (data rate will be reduced). However, fewer hash operations (non-linear transformations/functions) are required, reducing the overall delay. Hence, one can tune the value of BL to achieve the desired balance between the data rate and the required delay. Moreover, one main advantage is that the physical layer eliminates the need for implementing message authentication at the upper layers, which minimizes the required delay, consumed energy, computational complexity and overall required resources.

For transmitting the MAC values, a second variant of the proposed authentication scheme can be considered (Fig. 10.5). In particular, all of the obtained MAC values can be grouped into a separate frame symbol, which will be transmitted along with the rest of the frame symbols within one frame, instead of appending each MAC value to its corresponding frame symbol (Fig. 10.4).

At the receiver side, the proposed approach is used to calculate the MACs of

the received frame symbols. The obtained MAC values (complex) are compared to the received MACs. If both values are equal, the receiver proceeds with the data recovery process. It should be noted that prior to verifying the integrity and authenticity of received data, the receiver utilizes an equalizer to correct any possible error in the received messages.

10.1.3 Integer Non-Linear Finite Skew Tent Function

In this sub-section, the utilized non-linear function is discussed in more detail, for a better understanding of the proposed scheme.

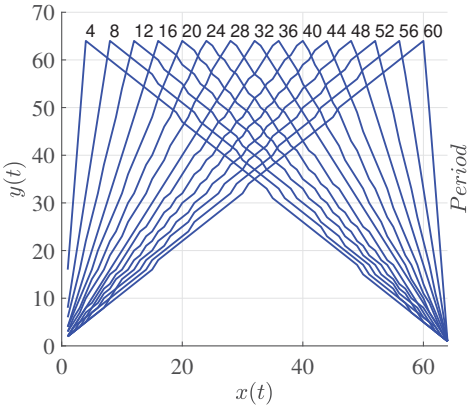
In principal, the proposed hash function depends mainly on the integer finite skew tent transformation, which is defined as

$$y = \begin{cases} \lceil \frac{Q \times x}{Pr_i} \rceil & x \leq Pr \\ \\ \lfloor \frac{Q(Q-x)}{(Q-Pr_i)} \rfloor + 1 & x > Pr \end{cases}$$

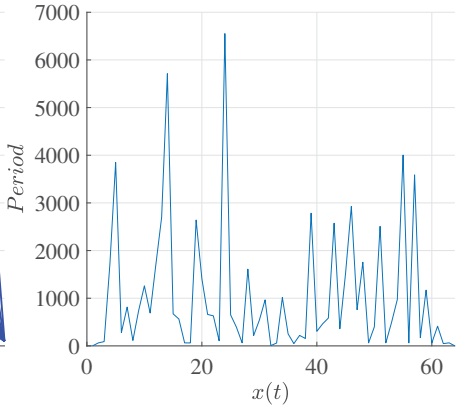
Where x and y are the input and output of this transformation, respectively. Note that the input should always have a positive integer value. In addition, $Pr = \{Pr_1,\ Pr_2,\ \ldots,\ Pr_i,\ \ldots\}$ represents a set of control parameters and Pr_i, which has values belonging to $\{1, \ldots, Q\}$, represents the i^{th} control parameter.

This function is non-linear, flexible and invertible (when the control parameters are known). In contrast, most traditional hash functions are non-linear, flexible and non-invertible. To ensure the one-way property, the proposed solution employs a dynamic key, which is based on a secret key and random physical channel parameters. This means that dynamic integer-complex mapping and acquiring/modifying the MAC of transmitted data will be extremely difficult for illegitimate users. Therefore, this function is considered safe for use in emergent networks.

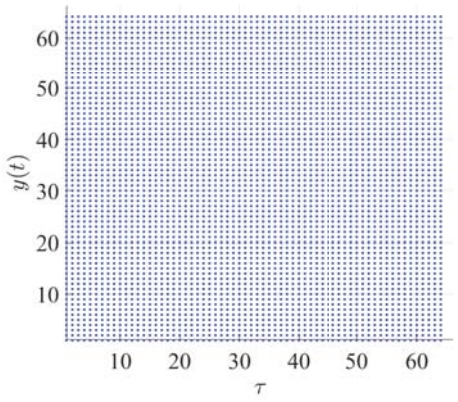
Figure 10.6a represents a piece-wise linear transformation, which is composed of two linear segments for the integer NLF, respectively (for different values of Pr). This figure proves the non-linearity of the utilized integer function in addition to the bijective property as shown in Fig. 10.6c. It has also been shown that the input and output values vary within the same range. Additionally, Figures 10.6b and 10.6d represent the periodicity and sensitivity of each control parameter, Pr. The periodicity results indicate that certain control parameters have higher periodicity than others. Low periodicity is obtained from low or high values of Pr. Therefore, by using variable control parameters, periodicity will be increased. Finally, it should be noted that the desired sensitivity value, which is 50%, is attained. Similar results are obtained for different values of Q.



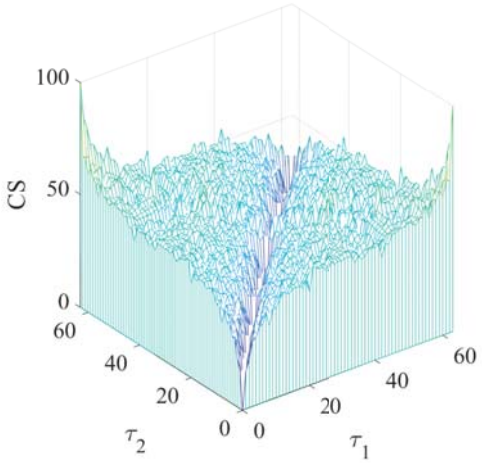
(a) Mapping Integer NLF



(b) Periodicity of selected Integer NLF



(c) Bijectivity Diagram tent



(d) Control Parameters Sensitivity

Figure 10.6: (a) The non-linear mapping of the integer skew tent function ($Q = 64$), (b) the bifurcation diagram represents the obtained values for each control parameter, which vary within $Pr \in \{1, 64\}$. (c) The corresponding periodicity of each control parameter Pr and (d) sensitivity

10.2 OFDM-Based Message Authentication Algorithm

The second message authentication algorithm targets 5G IoT systems utilizing OFDM. Similar to the first algorithm, this scheme also relies on the dynamic and random information extracted from the shared wireless channel, along with a secret parameter, to generate the message authentication code (MAC) of each OFDM frame symbol. This process is done in the time and frequency domains,

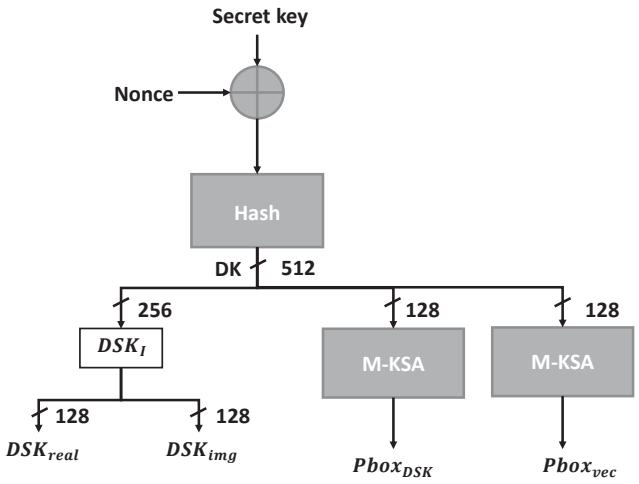


Figure 10.7: The proposed key generation scheme for the Pre-IFFT message authentication scheme

that is, before and after performing the IFFT transformation (two variants). Essentially, complex OFDM symbols are first pre-processed and then, key-hashed to ensure their integrity via a non-linear function (an integer function is used for the Pre-IFFT case and a non-integer function is used for the Post-IFFT case). After performing the proposed keyed-hashing algorithm on each OFDM frame symbol, all of the obtained MAC values are either appended to each frame symbol, or concatenated in a separate OFDM frame and transmitted along with the rest of the data. The proposed scheme exploits simple operations such as addition, mapping and permutation, which makes it very efficient and lightweight. It can be employed in multiple OFDM-based technologies, which include Internet of Things (IoT), VANETs, Power Line Communication (PLC) protocols and Device-to-Device (D2D) systems. The presented security and performance analysis prove that the proposed PLS source authentication and message integrity technique achieves the desired security properties and is suitable for resource-limited devices.

Since frequency-domain and time-domain symbols have different characteristics and properties, two different message authentications schemes are proposed. Specifically, one scheme is applied to the frequency-domain OFDM symbols (Pre-IFFT), while the other is applied to the time-domain symbols (Post-IFFT). In the security and performance evaluation sections, the effect of hashing in each case is quantified and assessed. Each of the two proposed schemes is divided into two phases: the sub-key generation process and the keyed-hashing function.

10.2.1 Frequency-Domain Message Authentication (Pre-IFFT)

In this variant of the scheme, hashing is applied to the frequency-domain OFDM symbols (complex), before the IFFT transformation. Here, it should be noted that frequency-domain OFDM symbols have a finite set of complex values, which result from the modulation operation (modulation order $M_O = 2^{m_b}$, m_b is the number of bits per constellation point). These values range from $-(log_2(M_O)-1)$ to $log_2(M_O) - 1$.

Sub-Key Generation: The dynamic key, which is 512 bits, is divided into three sub-keys:

- The first sub-key (256 bits) is used to generate the initial source authentication and message integrity key, DSK_I. This key is further divided into sub-keys, DSK_{real} and DSK_{img} (128 bits each), which will be combined with the real and imaginary vectors of each OFDM frame symbol to generate the corresponding MAC.
- The second sub-key (128 bits) is used to derive a permutation table $Pbox_{DSK}$ using the Modified key-Scheduling Algorithm (M-KSA) of RC4 [262] (Fig. 10.7). This permutation table is used to permute the initial key, DSK_I, for every new OFDM frame symbol.
- The third sub-key (128 bits) is used to generate a second permutation table, $Pbox_{vec}$, which is used to shuffle/permute a vector, vec_{MS}, which contains all of the possible complex modulation symbols corresponding to a specific modulation order (M_O), for every OFDM frame symbol.

Pre-IFFT Message Authentication Code: The proposed mechanism is divided into three main blocks (Fig. 10.8): the pre-processing block, the hashing block, and the post-processing block, which converts the obtained Message Authentication Code (MAC) integer value to complex representation (complex modulation symbols) (Algorithm 6).

- **Pre-processing block**: After performing modulation, each OFDM frame will contain NB_F OFDM frame symbols, $F = FS_1||FS_2||\dots||FS_{NB_F}$, and each OFDM frame symbol will include NB_{FS} complex modulation symbols (frequency-domain). Any M-QAM (Quadrature Amplitude Modulation) scheme, having $M_O = 2^{m_b}$ different modulation symbols, is applicable in this case (m_b is the number of bits).

  Every OFDM frame symbol is divided into $\frac{NB_{FS}}{BL}$ blocks, each having a size equal to BL (BL complex modulation symbols). If NB_{FS} is not divisible

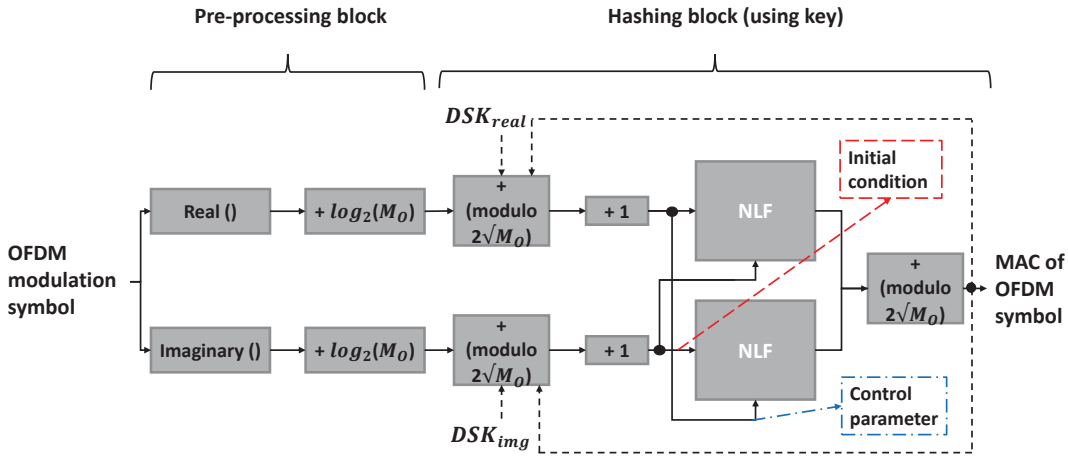


Figure 10.8: The proposed source authentication and message integrity scheme for frequency-domain OFDM symbols

Algorithm 6 The proposed frequency-domain message authentication algorithm

procedure FREQUENCY_AUTHENTICATION(FS_i, DSK_I, vec_{MS}, $Pbox_{vec}$)
  $temp1 \leftarrow DSK_{real}$
  $temp2 \leftarrow DSK_{img}$
  $vec_{MS} \leftarrow vec_{MS}(Pbox_{vec})$
  for $ind = 1$ to NB_{FS} **do**
    $Rt[ind] \leftarrow \mathsf{real}(FS_i[ind])$
    $It[ind] \leftarrow \mathsf{imag}(FS_i[ind])$
  end for
  $Rt \leftarrow PreProcessing(Rt,\ M_O)$
  $It \leftarrow PreProcessing(It,\ M_O)$
  for $it = 1$ to $\frac{NB_{FS}}{BL}$ **do**
    $BR_{it} \leftarrow Rt[(it-1) \times BL + 1 \rightarrow it \times BL]$
    $BI_{it} \leftarrow It[(it-1) \times BL + 1 \rightarrow it \times BL]$
    $O_1 \leftarrow NLF(temp1 + BR_{it},\ BI_{it})$
    $O_2 \leftarrow NLF(temp2 + BI_{it},\ BR_{it})$
    $temp1 \leftarrow O1 + O2$
    $temp2 \leftarrow temp1$
  end for
  $MAC \leftarrow vec_{MS}(temp2)$
  return MAC
end procedure

by BL padding is applied. There are no restrictions on the size of BL in the hashing scheme, hence padding is discarded after dividing the OFDM

frame symbol into smaller blocks. For each block, the real and imaginary components of each complex modulation symbol are grouped into two separate vectors (each of size BL). Next, the real and imaginary components in each vector, which have values between $-(log_2(M_O)-1)$ and $log_2(M_O)-1$, are shifted to positive integers by adding a value of $log_2(M_O)$, such that all values range between 1 and $2(log_2(M_O))-1$. This step is necessary since the input to the non-linear integer function (NLF) should always be a positive integer.

- **Hashing block**: In this stage, every OFDM symbol block is processed (hashed) in a sequential manner to finally obtain the MAC value of the OFDM frame symbol, which will be appended to the frame symbol itself or within a separate frame symbol, after $2 \times \frac{NB_{FS}}{BL}$ non-linear operations (two Non-Linear Integer Functions (NLFs) are utilized in the proposed hashing scheme). Since the obtained real and imaginary vectors, following the preprocessing block, contain integer values, a non-linear integer function has been utilized in the keyed-hashing process.

  For the first round of processing (first block), the real and imaginary vectors are added (modulo $2\sqrt{M_O}$) to DSK_{real} and DSK_{img}, respectively. Then, the two output vectors are added to a value of 1, in order to guarantee that the input of NLF is a positive integer, greater than zero. Afterwards, the obtained vectors enter the two NLFs, such that each vector represents the initial condition in one NLF and the control parameter in the other. The outputs of the two NLFs are again added (modulo $2\sqrt{M_O}$), to produce the MAC value of the first input block. For subsequent blocks, the same steps are applied, however, the obtained MAC value of the previous block is used instead of DSK_{real} and DSK_{img}. The MAC value (size BL) that is generated for the final block represents the integer MAC of the whole OFDM symbol.

  This two-step process is repeated for all of the OFDM frame symbols in each frame, and the output is either appended to each frame symbol separately or concatenated and inserted in one or several OFDM symbols at the end of each frame (depends on the size of MACs). For every frame symbol, an updated version of $DSK_I = DSK_{real}||DSK_{img}$ is generated, using $Pbox_{DSK}$ (Fig. 10.8). The hashing process is repeated NB_F times, which is the number of OFDM frame symbols in one frame.

  Here, it should be noted that the utilized non-linear integer function takes as input two blocks (initial condition and control parameter) of size BL, containing values between 1 and $2(log_2(M_O))-1$, and outputs a block of size BL, containing integer values between 1 and $2(log_2(M_O))-1$. This output is referred to as the integer MAC, $MAC_{Integer}$. These integer MAC values should be converted back to complex representation to satisfy system

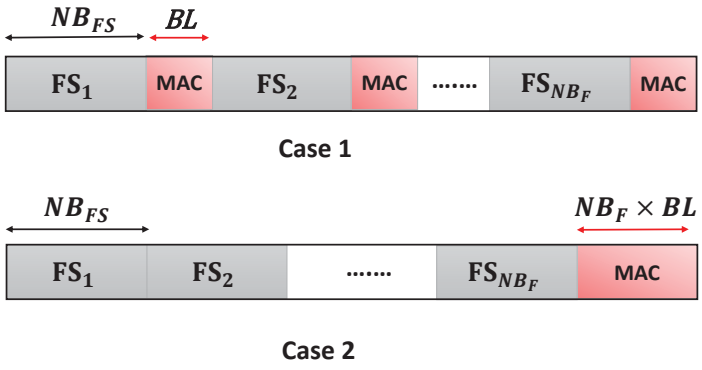


Figure 10.9: The two proposed structures of one OFDM frame

compatibility (input of IFFT transformation should be complex), which is described next.

- **Post-processing block**: In order to convert the $MAC_{Integer}$ values back to complex representation in an efficient manner, the same lightweight mapping operation that was proposed in the generic case is used: a vector vec_{MS} is initially constructed, containing all the possible complex modulation symbols for a specific modulation order (M_O). The complex MAC values are derived based on the following equation:

$$MAC_{Complex} = vec_{MS}(MAC_{Integer}). \tag{10.3}$$

  For every new OFDM frame symbol, vec_{MS} is permuted using $Pbox_{vec}$, which increases further the security level of the proposed source authentication and message integrity scheme. Finally, after hashing all of the OFDM frame symbols, the corresponding MAC values are either appended to their corresponding frame symbols or inserted within a separate OFDM frame symbol(s) (Fig. 10.9).

The resulting overhead of the proposed scheme is equal to $NB_F \times BL$. However, as it will be shown later in the performance analysis section, the proposed scheme is more efficient than currently employed schemes such as the SHA-512, CMAC, and GMAC, since it is based on the random characteristics of the physical layer (only one iteration round). Consequently, this eliminates the need for implementing message authentication at the upper layers, which minimizes the required delay, consumed energy, computational complexity and overall required resources.

At the receiver side, the same approach is used to generate the MACs of the received frame symbols. The obtained MAC values (complex) are compared

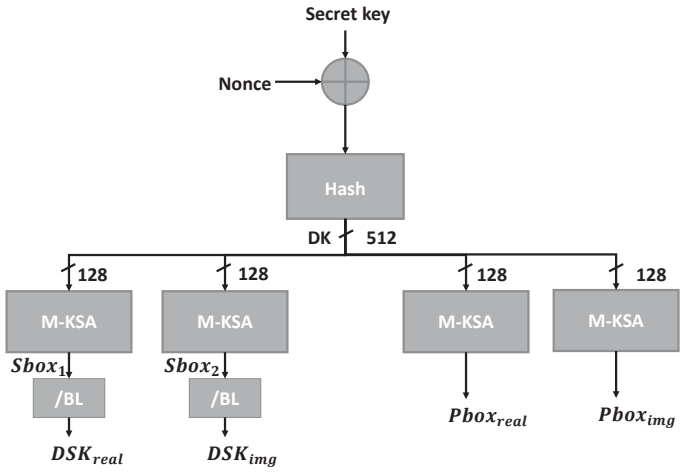


Figure 10.10: The proposed key generation scheme for the Post-IFFT message authentication scheme

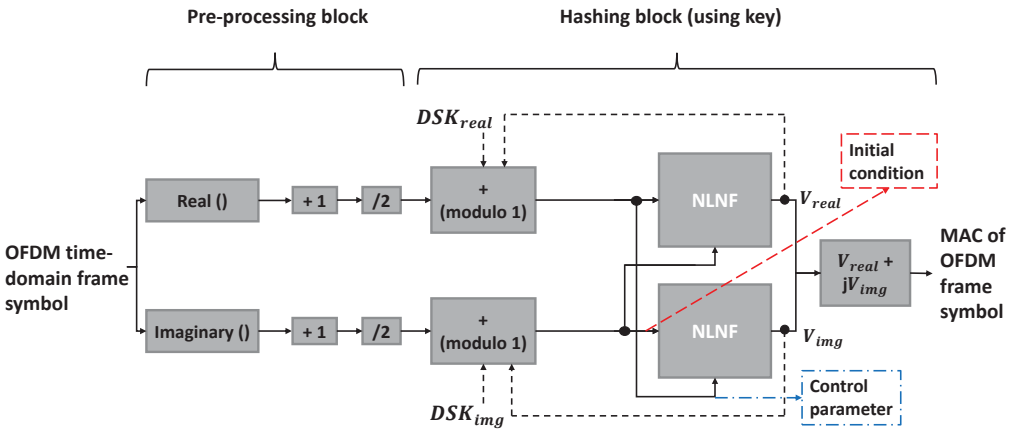


Figure 10.11: The proposed source authentication and message integrity scheme for time-domain OFDM symbols

to the received MACs. If both MAC values are equal, the receiver verifies the legitimacy and integrity of the transmitted messages.

10.2.2 Time-Domain Message Authentication (Post-IFFT)

The second message authentication scheme is proposed in this sub-section, where hashing is applied on complex time-domain OFDM symbols, following the IFFT transformation. In general, time-domain OFDM symbols span a very large space (large set of floating points), which results from the IFFT operation. Therefore,

the NLF, which is used in the previous scheme (Pre-IFFT), cannot be utilized. Instead, a Non-Linear Non-Integer Function (NLNF) is used in the hashing process.

Sub-Key Generation: The same key generation process of the Pre-IFFT case can be utilized in this scheme, however, a slightly different scheme is proposed as a second alternative/candidate.

Again, a pre-shared secret key and a channel-based nonce are combined (XORed), and hashed (SHA-512) to generate a 512-bit dynamic key (Fig. 10.10). This key is then divided into four sub-keys (128 bits each). The first two sub-keys are used to produce two substitution tables ($Sbox_1$ and $Sbox_2$) using the M-KSA algorithm. Both substitution tables contain BL entries (equal to the size of the obtained blocks), and values between 0 and BL. These values, are then divided by BL, to obtain DSK_{real} and DSK_{img} (decimal values). This step is crucial to ensure compatibility with time-domain OFDM symbols, which have floating values. The second pair of sub-keys is used to generate two permutation tables $Pbox_{real}$ and $Pbox_{img}$, which are used to permute/shuffle DSK_{real} and DSK_{img} for each OFDM frame symbol, respectively.

Post-IFFT Message Authentication Code: The proposed scheme is divided into two main blocks: the pre-processing block and the hashing block. Post-processing is not necessary in the time-domain, since the output MAC values have the same complex representation as that of time-domain OFDM symbols.

- **Pre-processing block**: Following the IFFT operation, the real and imaginary components of time-domain OFDM symbols will have values between -1 and 1. First, these frame symbols are divided into $\frac{NB_{FS}}{BL}$ blocks (each containing BL complex symbols) and then, separated into two vectors: one containing the real values (size equal BL), and the other containing the imaginary values (size equal to BL). Next, both vectors are added to 1 and divided by 2 to produce two vectors containing elements in $\{0, 1\}$. Pre-processing is mandatory in this scheme since the input of the NLNF should be within $\{0, 1\}$.

- **Hashing block**: In this phase, each time-domain frame symbol is key-hashed, separately (Fig. 10.11).

  First, the real and imaginary vectors of the first symbol block are added (modulo 1) to DSK_{real} and DSK_{img}, respectively. Then, the two outputs are fed to two NLNFs, each acting as a control parameter for one NLNF and as the initial condition for the second. Finally, the obtained outputs are combined to form the complex MAC value. The values, V_{real} and V_{img}, are used instead of DSK_{real} and DSK_{img}, for subsequent blocks. Again, the final MAC value can be appended to the frame symbol itself or inserted in a separate frame symbol with the rest of the MAC values at the end

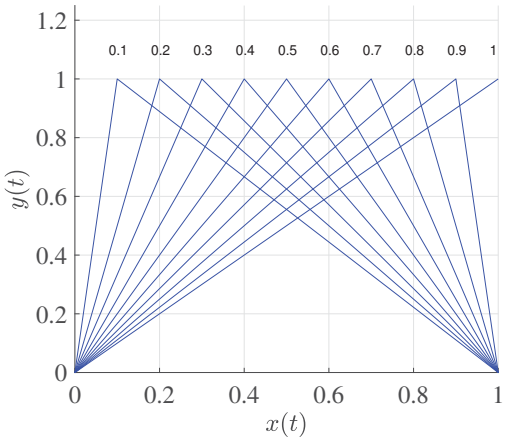


Figure 10.12: Non-linear mapping of the float skew tent function

of the frame. The output of the NLNF has a size BL, and the decimal values are between 0 and 1. Therefore, the produced MAC, at the end of this step, also has a size equal to BL and contains complex values between 0 and 1. Consequently, post-processing is not required. For every time-domain frame symbol, DSK_{real} and DSK_{img} are updated using $Pbox_{real}$ and $Pbox_{img}$, respectively.

10.2.3 Non-Linear Non-Integer Function

The utilized NLNF is defined by the following:

$$y = \begin{cases} \frac{x}{Pr_i} & x \leq Pr \\ \\ \frac{(x - Pr_i)}{(1-x)} & x > Pr \end{cases} \tag{10.4}$$

Similar to NLF, x and y are the input and output of this transformation, respectively. Both input and output values range between 0 and 1 (floating points). $Pr = \{Pr_1,\ Pr_2,\ \ldots,\ Pr_i,\ \ldots\}$ represents a set of control parameters and Pr_i, which has values within $\{1, \ldots, Q\}$, represents the i^{th} control parameter. On the other hand, this function is considered more efficient than NLF since it requires simpler and fewer operations.

Figures 10.6a and 10.12 represent a piece-wise linear transformation, which is composed of two linear segments for the integer and non-integer NLFs, respectively (for different values of Pr). Both of the presented graphs prove the non-linearity of the utilized integer and non-integer NLFs (bijective functions). It has also been shown that the input and output values vary within the same range.

In the following sections, the security and performance of each variant is evaluated in terms of different metrics. Moreover, the effect of the hashing operation in each domain is analyzed and discussed, and several conclusions are drawn. Since this work is the first work that proposes a keyed-hashing algorithm for OFDM systems based on the physical characteristics of wireless channels, it is very important to study and identify the advantages and limitations of both proposed techniques on the operation of the OFDM system. The proposed schemes (one round) are also compared with existing multi-round source authentication and message integrity schemes.

10.3 Security Evaluation

For the design and assessment of robust security schemes, a threat model is required to highlight on the capabilities of the adversary and the attack method. Consequently, two types of adversaries are considered here: the *honest-but-curious* adversary, who is a passive adversary, and the *malicious* adversary, who is an active adversary.

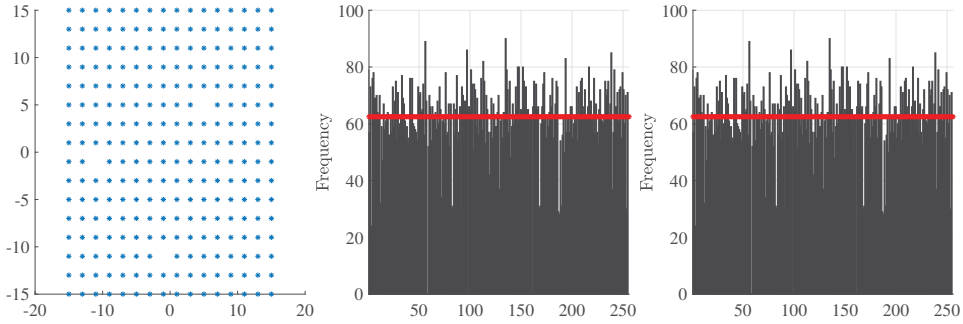
- The *honest-but-curious* passive adversary is a legitimate user that follows the protocol properly and does not deviate from it. However, these users try to acquire and learn as much information as possible, just by listening to the exchanged and received messages (passive eavesdropping), which compromises the security and privacy of communication [279].

- The *malicious* active adversary is an illegitimate user who attempts to deviate from the defined security protocol by injecting, modifying or deleting valid data. These users also try to impel other users to act maliciously and deviate from the defined protocol, by substituting their local inputs. In particular, the intentional modification of encrypted data by a malicious attacker compromises its integrity [280].

As such, the *honest-but-curious* user and the *malicious* user are considered to prove the proposed schemes' robustness in terms of source authentication and message integrity.

Next, several security tests are conducted to analyze and prove the feasibility and robustness of the proposed hash functions: the generic hash function and the OFDM-based hash function (Pre-IFFT and Post-IFFT). These schemes are compared to some of the well-known hashing schemes in the literature, namely HMAC-SHA-512 and CMAC. It should be noted that all existing schemes in the literature, including the HMAC-SHA-512 and CMAC algorithms, are applied at the bit level. In contrast, the proposed keyed-hashing functions are performed at the complex symbol level.

The security levels of the proposed Pre-IFFT and Post-IFFT schemes are also compared and assessed.



(a) (b) (c) CMAC

Figure 10.13: (a) The recurrence and (b) distribution of 1,000 MAC values (1,000 frames symbols) in complex representation using the proposed scheme for 256-QAM. (c) The distribution of MAC values for CMAC variant (same frames but at the bit level)

10.3.1 Randomness and Uniformity

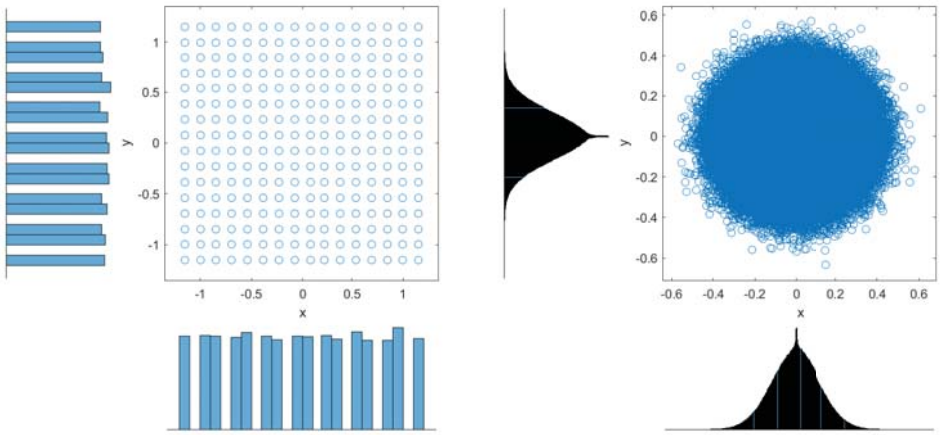
The security of any message authentication scheme is strongly related to the randomness and uniformity of the produced MAC values.

Generic Scheme: Figure 10.13a shows the obtained complex MAC values for all tested frames (for a block size equal to 16 elements). In addition to verifying the uniformity property, a histogram that evaluates the MAC values (complex number) of 1,000 input frame symbols is plotted in Fig. 10.13b for 256-QAM ($M_O = 256$). The x-axis represents all possible complex values for 256-QAM, which belong to the set of values $[0, 255]$ (256 complex values). The y-axis indicates the frequency of each complex value in the obtained MAC values.

From the results, it can be inferred that the mean of the generated integer hash values, which is approximately equal to 62.5, is close to the desired value of 58, where these values are uniformly distributed over the entire space. This clearly proves that the MAC output is uniformly distributed over the entire constellation space, for $M_O = 256$. In other words, for a specific input frame, the proposed keyed hash function generates hash values that span all the available region, which indicates a high level of randomness and uniformity.

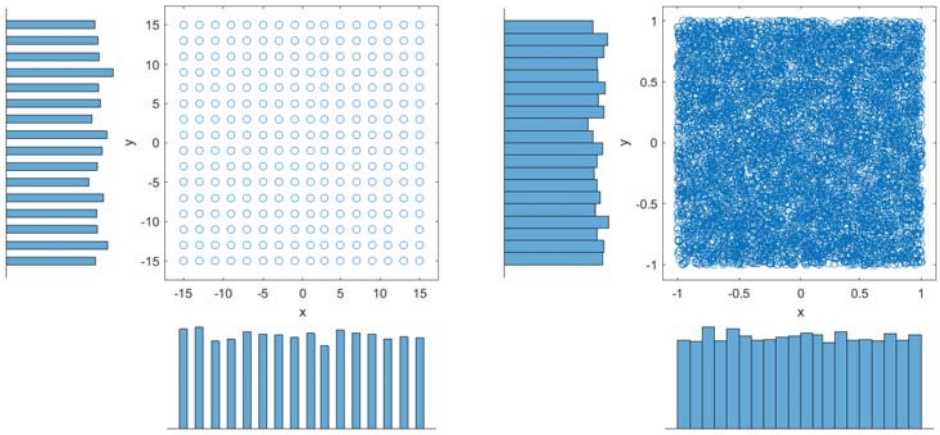
On the other hand, Fig 10.13c shows the distribution of 1,000 output hash values, using the CMAC algorithm. The presented results show that the proposed scheme achieves a similar distribution compared to the CMAC algorithm (very close to a uniform distribution).

OFDM-based Schemes: Figures 10.14a and 10.14b show the recurrence and distribution of 1,000 complex OFDM symbols before and after the IFFT



(a) Pre-IFFT (b) Post-IFFT

Figure 10.14: The recurrence and distribution of 1,000 frame symbols in complex representation before and after IFFT



(a) Pre-IFFT (b) Post-IFFT

Figure 10.15: The recurrence and distribution of 1,000 MAC values (1,000 frame symbols) in complex representation using the proposed scheme for 256-QAM after modulation, before and after the IFFT operation

transformation, respectively (before applying the proposed schemes). As mentioned previously, frequency-domain OFDM symbols have a finite set of complex values (normalized values in Fig. 10.14a), whereas time-domain symbols span a

larger space that includes an infinite set of floating-point values between −1 and 1 (Fig. 10.14b). The figures prove that frequency-domain symbols in the Pre-IFFT case are uniformly distributed over the entire constellation space (256 QAM). On the other hand, the real and imaginary components of Post-IFFT symbols have a lower randomness degree since these symbols have a normal distribution and are grouped within a specific area in the recurrence graph, as shown in Fig. 10.14b.

Next, the uniformity and recurrence of the obtained MAC values, for the two proposed schemes, are plotted in Fig. 10.15a and 10.15b. The results show clearly that the Post-IFFT case has a higher randomness degree than the Pre-IFFT case; it has a highly scattered recurrence plot and the generated MAC values occupy all of the available region. The recurrence plot of the frequency-domain symbol is also distributed over the entire space, however, the set of possible values is smaller. This observation is also validated using the histograms of the real and imaginary values, in which time-domain symbols are uniformly distributed over a larger set of points than frequency-domain symbols.

Comparing Figures 10.14 and 10.15, it is evident that both of the proposed schemes achieve the desired randomness degree and uniformity level and thus, both are considered secure and safe implementations for OFDM systems.

Finally, Fig 10.13c illustrates the histogram of 1,000 output hash values using the CMAC algorithm. The results validate that the proposed schemes achieve a similar distribution compared to the CMAC algorithm (very close to the desired uniform distribution). The distribution of the hash values using HMAC-SHA-512 has been omitted since it is very similar to that of the CMAC algorithm.

10.3.2 Key and Plaintext Sensitivity

In principle, sensitivity in security is related to key sensitivity and plaintext sensitivity. Key sensitivity refers to a major change in the hash value upon a slight change in the secret session key, SK, or the random channel-derived nonce. Whereas, plaintext sensitivity refers to having a major change in the hash value with any slight modification in the input message. If a message authentication scheme achieves the desired key and plaintext sensitivities, then the scheme satisfies the desired cryptographic properties and can resist analytic attacks, and more so for the proposed scheme since it is based on the dynamic key-dependent approach.

Generic Scheme: The key sensitivity test is plotted for 1,000 input frames having 1,000 different secret session keys. For each input frame, two secret session keys, SK and SK', are utilized. All the elements of SK' are equal to those of SK, except for the Least Significant Bit (LSB) of a random byte (LSB is flipped). The sensitivity value (difference) should always be close to 1 for the tested scheme to be considered secure. Figure 10.16a proves that the majority of the key sensitivity values (KS) are close to the ideal value, 1 (at the complex modulation symbol

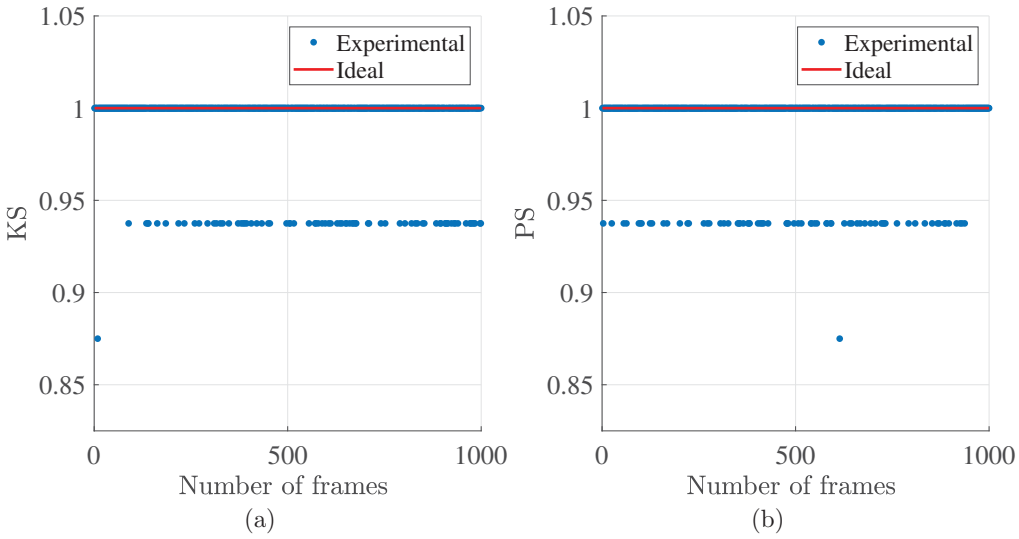


Figure 10.16: The Key Sensitivity (KS) and Plaintext Sensitivity (PS) values at the complex modulation symbols for 1000 input frames symbols, that are obtained by using the proposed general message authentication scheme

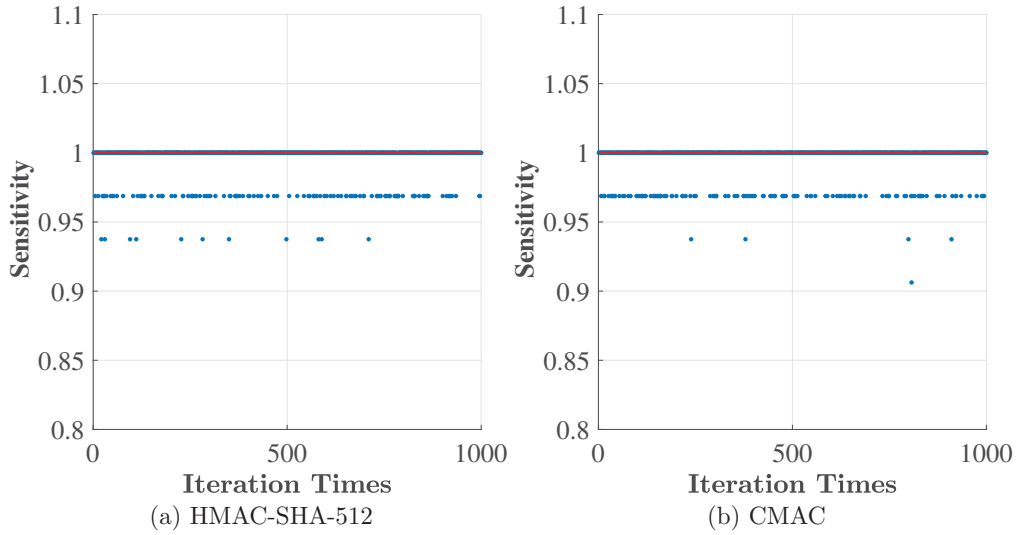


Figure 10.17: The input sensitivity values for 1000 input frames (bit level), that are obtained by using (a) HMAC-SHA-512 and (b) CMAC message authentication algorithms, respectively

level), where a one bit change in the secret session key results in a completely different complex MAC value at the output (mean of KS is approximately equal

to 97%).

Similarly, the plaintext sensitivity (PS) is plotted for 1,000 input frames. For each input frame, the PS value is calculated using two slightly different inputs (having all their values equal, expect for the LSB (flipped)). In Fig. 10.16b, the majority of the obtained results are close to the desired value of 1 (at the modulation complex symbol level). The average of the plotted PS values is equal to 96%. Hence, the proposed scheme satisfies the required plaintext sensitivity property. Consequently, the KS and PS test results prove that the proposed scheme exhibits a high sensitivity level.

In addition, Fig. 10.17 shows the key sensitivity of the HMAC-SHA-512 and CMAC algorithms. The obtained results are similar to those of the proposed scheme.

The same input frames are used in this comparison except that the HMAC and CMAC schemes are performed at the bit-level (blocks containing a fixed number of bits). The security of the HMAC and CMAC schemes against chosen/known plaintext attacks, is similar to that of the proposed scheme since the sensitivity test of the HMAC and CMAC schemes at the bit level is equivalent to that of the proposed solution at the modulation complex symbol level.

In fact, the high level of key and plaintext sensitivities of the proposed approach using the dynamic key-dependent approach result in high immunity against linear/differential attacks, chosen/known plaintext/ciphertext attacks and key-related attacks.

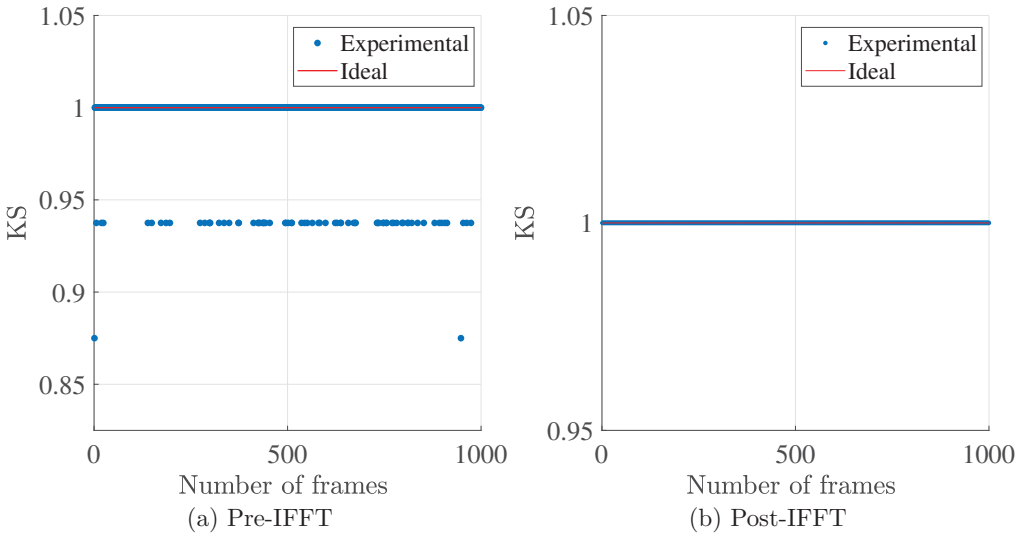


Figure 10.18: The Key Sensitivity (KS) values of complex modulation symbols, for 1000 input frames symbols, Pre-IFFT and Post-IFFT

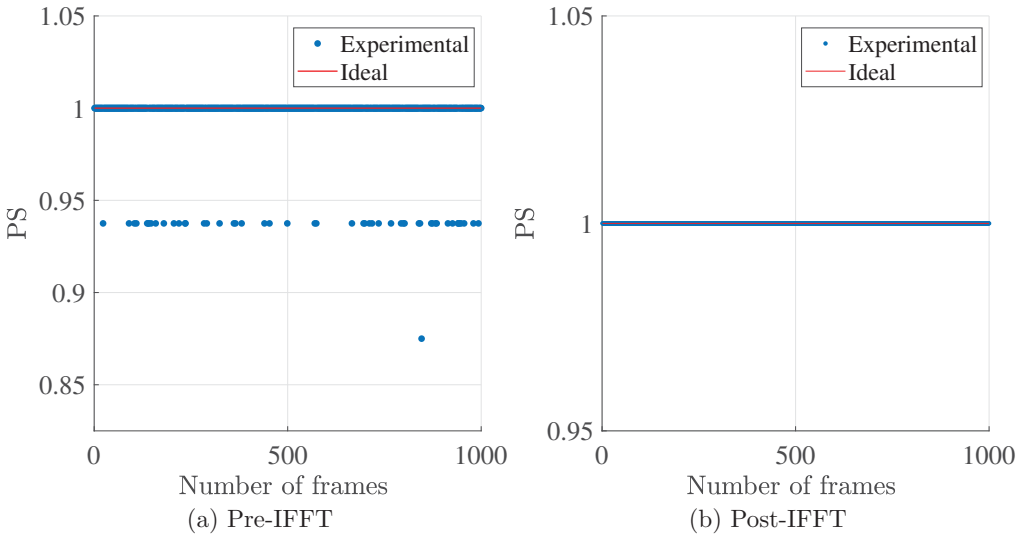


Figure 10.19: The plaintext Sensitivity (PS) values of complex modulation symbols, for 1000 input frames symbols, Pre-IFFT and Post-IFFT

OFDM-based Schemes: The proposed schemes are evaluated using the key sensitivity and plaintext sensitivity tests to prove that both schemes satisfy the desired cryptographic properties and are able to resist several attacks such as linear/differential attacks, chosen/known plaintext/ciphertext attacks and key-related attacks.

The key sensitivity test is performed on 1,000 input frames, using 1,000 different secret keys. For each input frame, two secret keys that differ by only one bit are utilized. In general, the proposed schemes are considered secure if the obtained key sensitivity (KS) is close to 1 (100% difference). The results in Fig. 10.18a prove that the majority of the KS values are close to the desired value of 1 (most values are above 0.9). Hence, a one-bit change in the secret key is sufficient to completely change the output MAC (complex value) using the proposed Pre-IFFT hashing scheme. The same results are obtained for the Post-IFFT scheme where all of the produced KS values are equal to the ideal value, 1 (Fig. 10.18b). This is logical since time-domain symbols have complex floating values and hence, the difference in the output is more explicit (strict difference).

Following the key sensitivity test, the proposed schemes have been analyzed in terms of plaintext sensitivity. Again, plaintext sensitivity (PS) is plotted for 1,000 input frames. For each input frame, two inputs that differ by only one bit are used. The PS value should always be close to 1 since a one-bit difference in the input should result in a 100% different output. Figures 10.19a and 10.19b demonstrate that both schemes, Pre-IFFT and Post-IFFT, achieve the desired plaintext sensitivity property.

The sensitivity values of the HMAC and CMAC algorithms have also been presented. Figure 10.17 shows the plaintext sensitivity of the HMAC-SHA-512 and CMAC algorithms, respectively. The results are similar to those of the proposed schemes. This confirms that the proposed schemes achieve the required security level as that of currently used hashing techniques.

10.4 Cryptanalysis

In this section, the proposed message authentication schemes are evaluated in the context of different authentication attacks, in order to prove their safe and efficient deployment in current and future communication systems that employ physical layer security. In principle, any cryptographic algorithm should ensure the desired confusion and diffusion properties, which are assessed based on statistical analysis, in addition to the key and plaintext sensitivity tests. The conducted tests in the previous section, have proven that the proposed schemes achieve the message and key avalanche effects.

The cryptographic security of the proposed schemes relies on two main properties:

1. Confusion and diffusion primitives that preserve the unpredictability and high sensitivity of the output MAC, the secret key and the dynamic physical channel.

2. A compression function that ensures the desirable cryptographic properties.

In the following, several cryptanalysis tests are presented and discussed.

10.4.1 Key Space Analysis

In order to safeguard against brute force attacks, the message authentication algorithm should have a large key space. In general, the key space of a secure cryptographic algorithm should not be less than 2^{128} to be considered secure (safe against brute force attacks). The size of the schemes' secret session key can be 128, 196 or 256, while the size of the dynamic key is 512 bits. Consequently, the proposed schemes have a sufficiently large key space, rendering the brute force attacks unfeasible.

10.4.2 Pseudo-Collision Resistance

Normally, in collision attacks, adversaries exploit the vulnerabilities and weaknesses of compression functions to intercept transmitted messages and/or modify them. More specifically, the adversary tries to find two different messages (M) and (M') that have the same hash value (collision) such that $h(M) = h(M')$;

$h(.)$ is a specific compression function. This is referred to as the pseudo-collision attack [281].

The plaintext sensitivity test, presented in the previous section, showed that any single change in the original frame results in a completely different MAC value when using the generic message authentication scheme. In addition, each input block is mixed with the output of the previous block and is then processed twice (enters one function as control parameter and enters the second as initial condition) using a non-linear function, which increases the randomness and sensitivity of the final MAC value. This process is applied to all frame symbol blocks expect for the first pair, which is mixed with a fragment of the dynamic key, DSK_I. This part of the dynamic key is related to a secret session key, which is shared between devices, and the common dynamic channel parameters (physical). Finally, a dynamic mapping operation is also employed at the end of the processing phase, and it is based on a dynamic permutation table, which is updated for each new frame symbol for the first variant and for each frame for the second variant. Hence, the proposed generic message authentication scheme has a sufficient number of random and dynamic input parameters and operations, and thus, it is highly resistant against pseudo-collision.

As for the OFDM-based scheme, the presented results in the previous section also showed that the proposed Pre-IFFT and Post-IFFT schemes exhibit the desired plaintext sensitivity property, where any bit change in the original message leads to a completely different hash value. Moreover, each OFDM symbol (in both schemes) is hashed in two steps: first the real and imaginary components of each OFDM symbol are separated and pre-processed. Then, each output is used as an initial condition for one non-linear function (integer or non-integer) and a control parameter for the other. The output of each non-linear function are combined to form the final MAC value. As a result, a higher randomness degree is achieved. The proposed schemes depend on several dynamic and random parameters, which change for every input frame symbol and hence, this increases their resistance against pseudo-collision attacks.

10.4.3 Resistance Against Birthday Attacks

The birthday attack, which is a sub-set of brute-force attacks, is a common type of cryptographic attacks and it is based on the birthday paradox problem. The success of this attack mainly depends on the high likelihood of collisions for a certain hash function [282].

More specifically, the attacker attempts to find two different messages having identical hash values with less than $2^{\frac{hb}{2}}$ trials; where hb is the number of bits in the MAC. In principle, this value should be large enough to resist brute force attacks ($hb \geq 128$ bits). Here, it should be noted that the time complexity and memory complexity are also equal to $2^{\frac{hb}{2}}$. In the proposed schemes, the size of the MAC (keyed hash block) is flexible and it is adjusted according to the

application requirements and limitations. The size of the MAC value can be increased to reach a higher security level and better performance, if devices are not limited. Also, the presented security tests proved that the proposed schemes exhibit high resistance against collision.

10.4.4 Resistance Against Meet-in-the-Middle Attacks

The meet-in-the-middle attack targets a specific category of hash functions, which is block cipher functions, since the success of this attack is directly related to the invertibility of the hash function. This attack aims at reducing the number of brute force attempts, where brute force is applied on both the plaintext and its corresponding ciphertext.

The proposed schemes do not rely on a block cipher, but on a non-linear function. Hence, this attack is not possible nor applicable in the proposed approaches. The employed non-linear function is an invertible function. However, in these schemes, it can be considered as a hard problem since the same input is used as initial condition and control parameter. In addition, the proposed schemes depend on a secret dynamic key (channel-based) and on several dynamic permutation tables.

As a result, acquiring any useful information or modifying the transmitted message is not possible.

10.5 Performance Analysis

This section assesses the performance and efficiency of the proposed techniques in terms of different metrics, and compares them against two well-known message authentication algorithms, namely the CMAC and HMAC algorithms.

10.5.1 Space Complexity

The generic message authentication scheme is applied at the frame symbol level. In addition, each frame symbol contains NB_{FS} complex modulation symbols. This means that it does not require previous or following frame symbols. Hence, the total space complexity is equivalent to $O(NB_{FS})$ (each frame symbol is processed independently).

On the other hand, the OFDM-based message authentication schemes are applied at the OFDM symbol level for both variants (frequency-domain and time-domain frame symbols). Each frame symbol contains NB_{FS} complex modulation symbols, and is divided into several blocks of size BL (BL complex modulation symbols). Moreover, the proposed solutions are designed in such a way that generating the MAC of one OFDM symbol does not require any previous or following frame symbols. Hence, the total space complexity is equivalent to $O(BL)$

for both variants (each frame symbol block is processed independently).

10.5.2 Block Size BL

Generally, there exists an inevitable time-space trade-off in security schemes. Hence, a good balance should be achieved between computational time and memory consumption, depending on the application requirements and device limitations. The performance of the proposed schemes highly depends on the parameter BL, which has a significant impact on the required memory size. In fact, when the value of BL increases, a large memory space is needed for storing the frame symbols and the resulting MACs (MACs also have a size BL). A large value of BL also reduces the data rate since the MAC value will have a large size. In contrast, when BL decreases, the number of blocks, and the time it takes to hash one frame symbol (computational time) both increase (the number of non-linear operations increases). For this reason, the proposed schemes use a flexible and variable value of BL, which is chosen according to the used application and available resources. For example, a large value of BL can be employed (128 or 256) for devices with no memory constraints. However, for small and resource-limited devices (low memory capacity), a low value of BL can be chosen (16 or 32).

10.5.3 Computational Complexity and Delay Cost

In order to assess the efficiency of the proposed schemes, their computational complexity and delay cost are evaluated in comparison with currently employed schemes. These metrics are essential for studying the performance of a certain security scheme since they reflect the associated overhead and introduced delay.

The following terms are defined:

1. T_S denotes the required execution time for the substitution operation.
2. T_{add} denotes the required execution time for the arithmetic "addition" operation between two blocks having BL elements.
3. T_{NLF} denotes the time required by the non-linear integer function.
4. T_{NLNF} denotes the time required by the non-linear non-integer function.
5. T_{XOR} denotes the time required by the XOR operation.
6. T_D denotes the time required by the AES "mix column" operation (for all 4 columns).
7. T_{SR} denotes the time required by the AES "shift row" operation.

The total Computational Delay (CD) of the proposed generic scheme to authenticate two blocks is:

$$CD_{Generic} = 3 \times T_{add} + 2 \times T_{NLF}. \tag{10.5}$$

The total Computational Delay (CD) that is required to authenticate one block using the proposed Pre-IFFT scheme is (hashing block):

$$CD_{Pre-IFFT} = 3 \times T_{add} + 2 \times T_{NLF} \tag{10.6}$$

On the other hand, the total Computational Delay (CD) that is required to authenticate one block using the proposed Post-IFFT scheme is equal to (hashing block):

$$CD_{Post-IFFT} = 2 \times T_{add} + 2 \times T_{NLNF} \tag{10.7}$$

Next, the total computational delay that is required to process one block using the standard AES method [142] is measured. Specifically, this quantification is done since AES is used in different authentication schemes such as GMAC and CMAC.

The total computational delay of the AES standard [142] is:

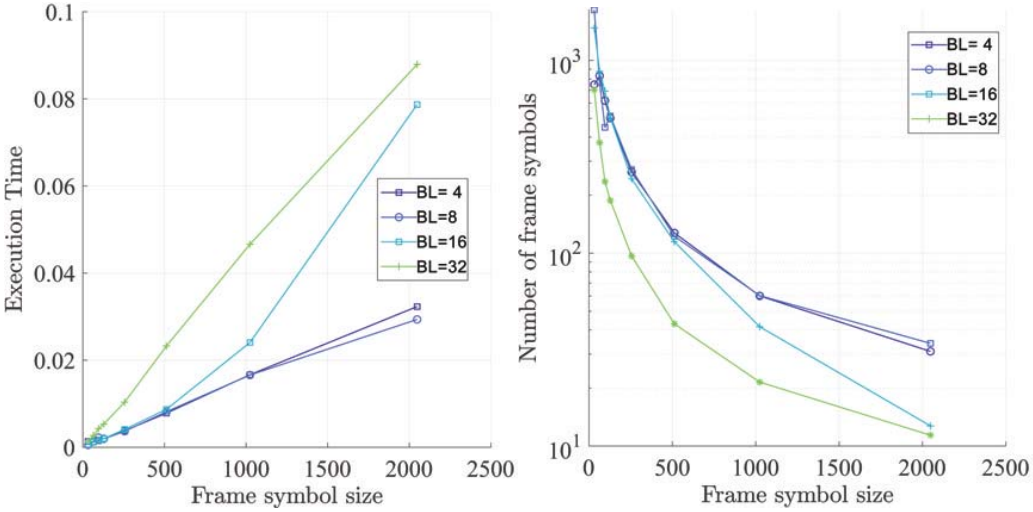
$$r_o CD_{AES} = r_o T_S + (r_o + 1)T_{xor} + (r_o - 1)T_D + r_o T_{SR}, \tag{10.8}$$

where r_o represents the number of rounds (for a 192- and 256-bit secret keys, r_o is equal to 12 and 14, respectively). The minimum value of r_o is 10 for a 128-bit secret key. Hence, the minimum computation delay for the AES scheme is equal to:

$$CD_{AES(r_o=10)} = 10T_S + 11T_{xor} + 9T_D + 10T_{SR}. \tag{10.9}$$

Comparing the previously listed schemes, it is evident that AES introduces a much higher computational delay than all proposed schemes. This is mainly attributed to the fact that the proposed solutions avoid diffusion operations such as the "mix column" operation, which is the most expensive AES operation in terms of resources and time (the delay of the addition and substitution operations are far less than that of the AES "mix-column" operation) [283].

On another note, the computation delay of the HMAC algorithm using $SHA-2$ variants, such as $SHA-256$ and $SHA-512$, is also related to the number of rounds. The number of rounds in the $SHA-256$ and the $SHA-512$ schemes are 64 and 80, respectively. For each iteration, a round compression function is applied. Therefore, AES is more optimized than SHA-2 and it is more preferable for resource-limited devices. This confirms that the proposed schemes are more efficient than currently existing schemes.



(a) Execution Time (sec) (b) Number of encrypted symbols per second

Figure 10.20: (a) The execution time (sec), and (b) the number of authenticated frame symbols as a function of frame symbol size ($\log_{10}$) for different sizes of BL, using the generic message authentication scheme

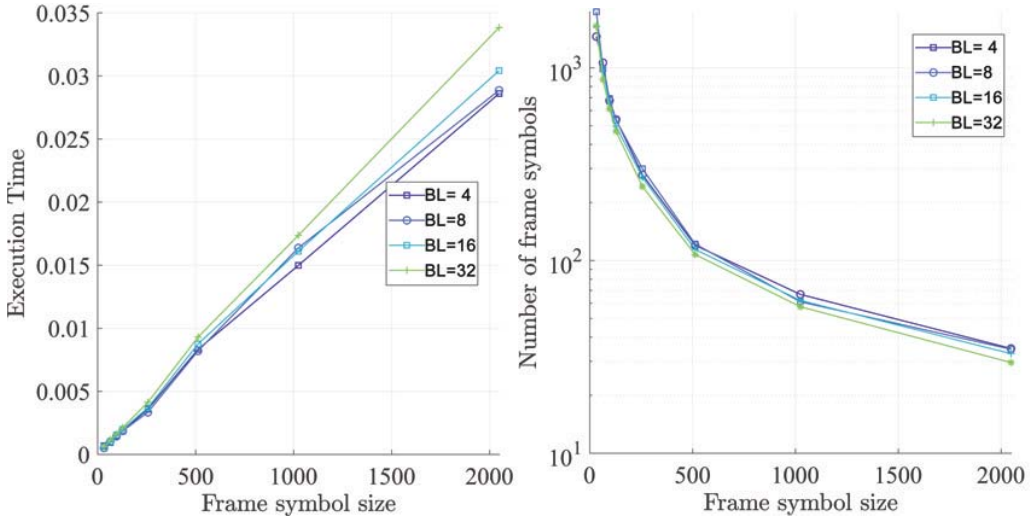
10.5.4 Execution Time

For a security technique to be considered efficient, it should have a low computational complexity, low execution time, and low energy consumption, all of which are desirable properties for limited devices (battery life). For this purpose, the average time (1,000 simulation rounds) has been calculated to hash one frame symbol having a variable size of $32, 64, 128, 256, 512, 1024$, and 2048. The following software and hardware conditions describe the setup: **Matlab R2018b simulator, Intel Core i7, 3 GHz CPU, 2 GB RAM Intel and the Microsoft Windows 7 operating system**.

Results in Figures 10.21 and 10.22 illustrate the execution time required by the Pre-IFFT and Post-IFFT schemes, respectively. In both schemes, the execution time increases with the increase of symbol length, NB_{FS}, since more processing time is required for hashing a larger frame symbol. This is also true for the generic scheme as illustrated in Fig. 10.20. On the other hand, it has been shown that the Pre-IFFT algorithm (Fig. 10.21a) has a higher execution time than Post-IFFT (Fig. 10.22a). This is due to the non-linear integer function, which includes a larger number of operations than the non-integer one (integer function requires multiplication in addition to division). The Post-IFFT case is designed with float operations (non-linear non-integer function), since the output of IFFT operation has float values. Moreover, the execution time of the generic case is close to that of the Pre-IFFT scheme, since both utilize NLF (especially, for $BL = 4$ and

8). Figures 10.21b and 10.22b complement the previous observation, in which a larger frame symbol size results in a higher processing time, and fewer number of encrypted frame symbols per second.

Finally, the time ratio between the frequency-domain and time-domain schemes has been plotted as a function of frame symbol size, for different block sizes, BL. The results from Fig. 10.23 show that as the symbol size increases, the time ratio between the two schemes converges to a fixed ratio, which is approximately equal to 2.1, for all BL (the time required by the Pre-IFFT case is twice the time required by the Post-IFFT case). For smaller symbols, the time ratio increases with a higher value of BL.



(a) Execution time (sec) (b) Number of encrypted symbols per second

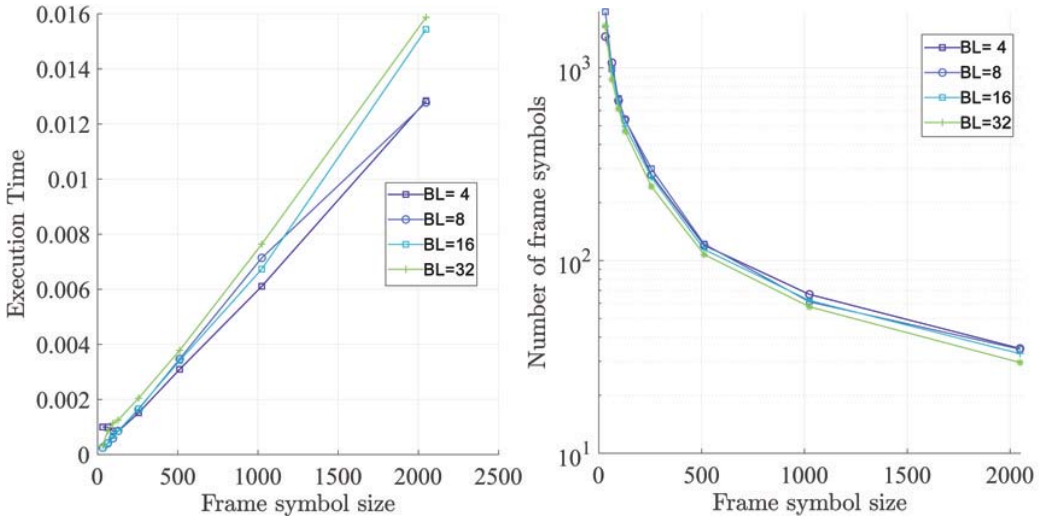
Figure 10.21: The variation of the execution time (sec), and the number of authenticated frame symbols (natural logarithm log) as a function of the frame symbol size, for different sizes of BL in the Pre-IFFT scheme

10.5.5 Flexibility

The proposed hash functions are applied on multiple blocks, having a variable and flexible length equals to BL (BL elements). Specifically, this value can be adjusted according to the devices' constraints and applications' requirements.

10.5.6 Hardware and Software Implementations

The hardware and software implementations, mainly, depend on the utilized operations in a security scheme. In this chapter, two message authentication algorithms are proposed: the first one is based on the non-linear integer skew tent



(a) Execution time (sec) (b) Number of encrypted symbols per second

Figure 10.22: The variation of the execution time (sec), and the number of authenticated frame symbols (natural logarithm log) as a function of the frame symbol size, for different sizes of BL in the Post-IFFT scheme

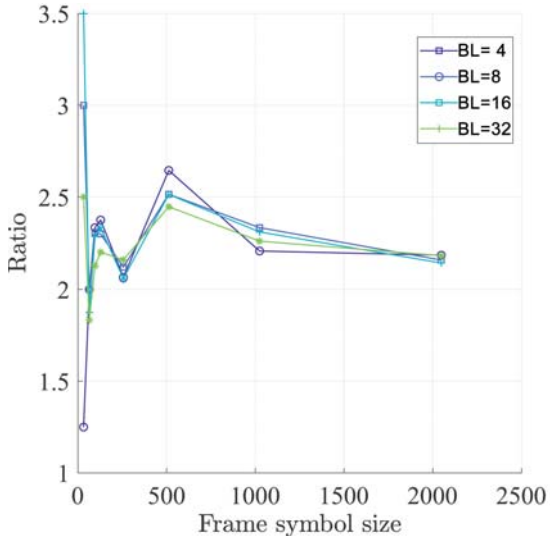


Figure 10.23: The time ratio between the proposed Pre-IFFT and Post-IFFT schemes as a function of frame symbol size and for different sizes of BL

function, while the second uses, both, the original float (non-integer) skew tent function and the non-linear integer function.

The Pre-IFFT scheme and the generic scheme can be optimized by using a look-up table for the division and multiplication operations, which reduces the required delay. Therefore, instead of performing division and multiplication in the integer case, look-up division and multiplication tables are called. This makes the corresponding hardware and software implementation of the generic and Pre-IFFT message authentication schemes simple and efficient, similar to that of the non-integer scheme. In fact, the non-linear non-integer operation depends on simpler and fewer operations (without multiplication). Moreover, it should be noted that the proposed optimization cannot be applied to the Post-IFFT variant, since the division operation is realized with the float precision.

The proposed message authentication schemes also use other operations such as permutation (shuffling), addition and selection, all of which are considered simple and efficient in terms of execution time.

10.5.7 Parallel Computation

As previously discussed, each frame symbol can be processed separately and independently from the others, whether in the frequency domain or time domain. Consequently, one can benefit from parallel computing, which results in a considerable reduction in the execution time (overhead delay).

In summary, the proposed message authentication algorithms are based on a one-round structure that consists of simple operations to authenticate message blocks. The results clearly confirm the reduction in the execution time and consequently, a reduction in the required energy consumption. Also, the proposed solutions are considered more efficient than existing schemes since they exhibit a relatively small overhead.

Chapter 11

Availability

In order to ensure the availability of data, three approaches can be exploited, namely Random Linear Network Coding (RLNC), multi-homing and PLS. Specifically, the inherent features and characteristics of these emerging technologies, along with PLS, pave the way for novel security solutions that are able to overcome effects of availability and jamming attacks. In this chapter, two RLNC-based PLS solutions are presented (integer RLNC and binary RLNC).

11.1 Availability based on Integer Random Linear Network Coding

A lightweight and efficient scheme based on RNLC is proposed to secure multi-homed IoT devices. Specifically, this scheme leverages three concepts, which are multi-homing, RLNC and PLS, to enhance the reliability of future networks (including 5G IoT systems). It should be noted that the proposed solution can be employed by any multi-homed device in 5G networks, but a 5G IoT system is considered for consistency. The term "multi-homed IoT" is not new. In fact, multi-homing is an important attribute of IoT, especially that this technology supports multiple protocols, few of which are Bluetooth, WiFi, cellular, ZigBee and LoRaWAN.

Using the proposed scheme, multi-homed IoT devices utilize the previously obtained dynamic key to derive the cryptographic primitives needed for the encryption/encoding process, which consists of simple and lightweight operations, mainly two permutation operations and a matrix multiplication operation. The first permutation operation is realized at the byte level, while the second one is realized at the block level. In addition, a set of encoding matrices, which is derived from the obtained dynamic key, serves two purposes: encryption and encoding. This is attributed to the fact that the coding coefficients (in Galois field) of these matrices are dynamic, pseudo-random and known to the communicating entities only. Hence, the proposed scheme jointly enhances the system's security

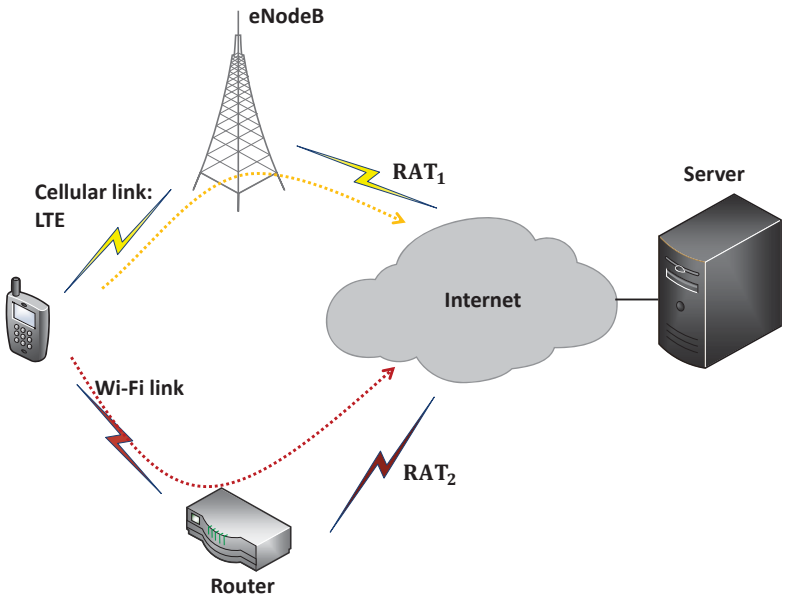


Figure 11.1: A special case of a multi-homed system with two RATs ($Rt = 2$), including LTE cellular communications and Wi-Fi direct

(confidentiality), reliability and availability.

Before discussing the proposed scheme in more detail, the basic methodology of RLNC and the considered system model are presented for a better understanding of the scheme.

11.1.1 System Model

Since most devices are currently equipped with several radio interfaces (multiple Radio Access Technologies RATs) of different access technologies (wireless and cellular), multi-homing is being widely promoted as a propitious solution to overcome the problem of massive traffic growth. More specifically, multi-homing allows the simultaneous transmission of data over multiple data links using different IP addresses, where the data stream is split into multiple sub-streams. This increases the system's reliability, efficiency and throughput, substantially [284, 285].

Figure 11.1 illustrates a special case of multi-homed systems having two RATs ($Rt = 2$). Throughout this section, a general multi-homed system with Rt RATs (rows) is considered. A message of size NB_M bytes is to be transmitted over Rt available channels/RATs, where each RAT (link) has a different data rate. Using the proposed scheme, data will first be encrypted, encoded and then transmitted (divided) over different RATs. Moreover, data communication is assumed to be end-to-end. Opposed to traditional RLNC scenarios, the encoding coefficient

vectors in the proposed scheme are not included in the header of the communication message. In fact, intermediate nodes do not apply any encoding operations, unlike traditional network coding schemes. Hence, the required computation and resource overhead are only introduced at the source and destination.

11.1.2 RLNC: Basic Concept

Network Coding (NC) is an encoding/decoding technique designed to increase the network's overall performance, and reduce the associated delays [286]. Using this technique, intermediate nodes process and encode received packets before sending them again. In general, there are three types of coding mechanisms in communication networks: 1) source coding which aims at reducing data redundancy and resources using compression, 2) channel coding which introduces redundancy to enhance reliability over a lossy medium, and 3) network coding which falls in between these two layers [287]. In addition, Random Linear Network Coding (RLNC) is a new and powerful network coding variant [288], that has been gaining so much popularity lately, due to its efficient and close-to-optimal performance [289, 290].

In particular, RLNC is an emerging coding discipline that enables network nodes to generate linear mappings of input to output data symbols over a finite field (independently and randomly) [287]. It is flexible in the sense that it can be applied at any position within the protocol stack (application layer, network layer, physical layer, ...). The encoding process is briefly described as follows: first, the original stream of data is divided into symbols of fixed sizes. Then, each symbol is multiplied with scalar coding coefficient that is randomly chosen from a Galois field (GF(2^{qf}), where qf is the field size), as illustrated in Equation 11.1:

$$\begin{aligned} PC &= G \odot PA \\ &= \begin{bmatrix} pc_1 \\ pc_2 \\ \vdots \\ pc_{Rt} \end{bmatrix} = \begin{bmatrix} G_{1,1} & \cdots & G_{1,ht} \\ G_{2,1} & \cdots & G_{2,ht} \\ \vdots & \ddots & \vdots \\ G_{Rt,1} & \cdots & G_{Rt,ht} \end{bmatrix} \odot \begin{bmatrix} pa_1 \\ pa_2 \\ \vdots \\ pa_{ht} \end{bmatrix} \end{aligned} \tag{11.1}$$

where Rt is the number of packets stored at the node's network coding level buffer, $G_{i,j}$ corresponds to the j^{th} random linear network coding coefficient of the i^{th} vector, $i = 1,\ 2,\ \ldots,\ Rt$ and $j = 1,\ 2,\ \ldots,\ ht$. In addition pa_j and pc_i are the original and encoded packets, respectively [291]. This simple operation can be viewed as creating linear combinations of input packets (data symbols), using encoding vectors that contain random coding coefficients. As a result, network reliability and efficiency are both enhanced, simultaneously [287, 292, 293]. Here, qf is set to 8.

In this chapter, the proposed schemes are applied before modulation.

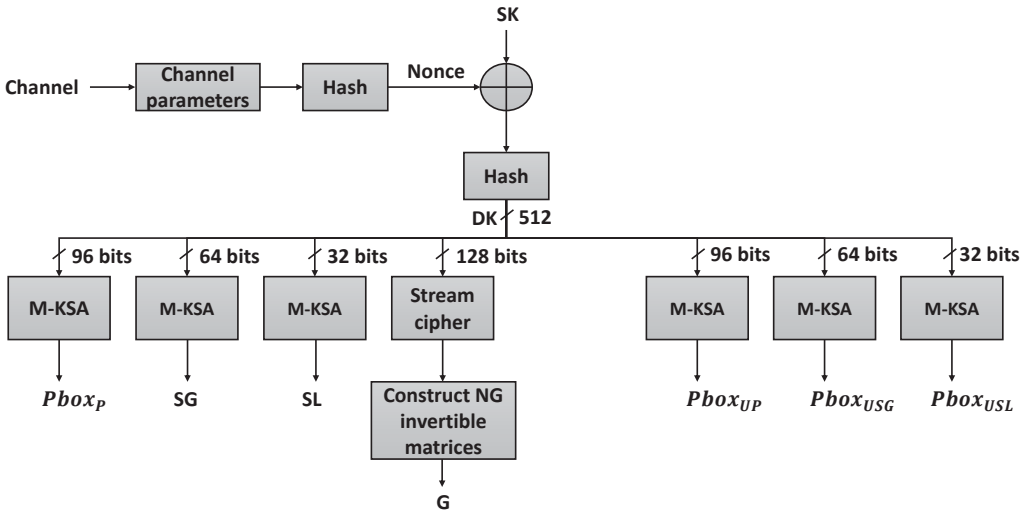


Figure 11.2: The proposed key derivation scheme and cipher primitive construction process

11.1.3 Proposed Integer RLNC Scheme

To derive the Dynamic Key, (DK), a multi-homed user extracts Rt different nonces from each of the Rt shared channels. These nonces are combined and hashed using a secure cryptographic hash function (SHA-512), and then XORed (Exclusive OR) with the pre-shared secret session key, SK. The resulting output is also hashed (h(.)) using SHA-512, to obtain DK (512 bits), as presented in the following equation:

$$DK = h(SK \oplus h(\oplus_{i=1}^{Rt} N_{0_i})) \tag{11.2}$$

Next, DK is divided into seven sub-keys as shown in Fig. 11.2. The first sub-key (96 bits) is used to derive the permutation table, $Pbox_P$, which has a size equal to the size of input data, $1 \times NB_M$ (message is NB_M bytes). This permutation table is used to permute the original message. The Modified key-Scheduling Algorithm (M-KSA) of RC4, which is presented in [262], is used in this step. The second sub-key (64 bits) is used to generate the pseudo-random selection table, SG, which has a size equal to $1 \times \Omega$ ($\Omega = \lceil \frac{NB_M}{ht^2} \rceil$), and pseudo-random values between 1 and NG (1, 2, ..., NG). More specifically, SG is used to randomly select a secret encoding matrix, $G_{SG(j)}$ ($1 \leq j \leq \Omega$, where $\Omega \geq NG$), to each group of data. Here, data is reshaped after permutation, into multiple sub-matrices, as it will later be described in the following subsection. SL, which is obtained using the third sub-key (32 bits), has a size equal to Rt and random values between 1 and Rt; Rt is number of active sub-channels used to transmit data. This $1 \times Rt$ vector is used to randomly select the segment/row of data that will be transmitted on each sub-channel using a different RAT. The fourth

sub-key (128 bits), on the other hand, is used to construct a set of NG invertible secret coding matrices, namely G, where $G = G_1,\ G_2,\ \ldots,\ G_{NG}$ (each having Rt rows and ht columns ($ht \leq Rt$)). Each of these encoding matrices will be randomly selected and multiplied with a fraction of the permuted input data (reshaped and divided into several sub-matrices), using the selection table SG. This step, which is referred to as secret coding, ensures data availability in addition to data confidentiality. Finally, the last three sub-keys are used to construct the update permutation tables ($Pbox_{UP}$, $Pbox_{USG}$ and $Pbox_{USL}$), that are used to update cipher primitives $Pbox_p$, SG and SL for every new input message, respectively. This step is crucial for enhancing the security of the proposed technique, and reducing the required delay and resource overhead. More specifically, $Pbox_P$ is permuted after each input message using $Pbox_{UP}$, which has a size equal to that of $Pbox_P$ (generated using a 96 bit sub-key). Similarly, SG and SL are updated in the same manner using $Pbox_{USG}$ (generated using a 64 bit sub-key and has a length equal to that of SG) and $Pbox_{USL}$ (generated using a 32 bit sub-key and has a length equal to that of SL), respectively.

Proposed secret coding scheme: After generating the dynamic key and deriving the needed cryptographic primitives, secret coding and encryption are performed in order to ensure the secure and efficient transmission of data. First, input data is reshaped to form a vector of size $1 \times NB_M$ (message is NB_M bytes before modulation). Then, it is permuted using the produced dynamic permutation table $Pbox_P$ that has the same length as that of the input message ($1 \times NB_M$). The permuted data is then reshaped (after applying padding if necessary, since the corresponding length, NB_M, should be divisible by ht^2) into $\Omega = (NB_M/ht^2)$ sub-matrices, $PD = PD_1||PD_2||\ldots||PD_{(\Omega)}$. Each permuted sub-matrix PD_j has ht rows and ht columns ($ht \times ht$), for $j = 1, 2,\ \ldots,\ \Omega$.

Next, the selection table SG ($1 \times \Omega$ and $NG \leq \Omega$), which contains random numbers between 1 and NG, is used to randomly select an encoding matrix from the set $G = G_1, G_2, \ldots, G_{NG}$, for each input permuted sub-matrix. The size of each matrix in G is $Rt \times ht$, where $ht \leq Rt$. In fact, each $G_{SG(j)}$ is used to encode the j^{th} PD sub-matrix (PD_j) to produce the j^{th} encoded sub-matrix CM_j for $j = 1,\ 2,\ \ldots,\ \Omega$, as indicated in the following equation:

$$\begin{aligned} CM_j &= G_{SG(j)} \odot PD_j \\ &= \begin{bmatrix} G^{1,1}_{SG(j)} & \cdots & G^{1,ht}_{SG(j)} \\ G^{2,1}_{SG(j)} & \cdots & G^{2,ht}_{SG(j)} \\ \vdots & \ddots & \vdots \\ G^{Rt,1}_{SG(j)} & \cdots & G^{Rt,ht}_{SG(j)} \end{bmatrix} \odot \begin{bmatrix} PD^{1,1}_j & \cdots & PD^{1,ht}_j \\ PD^{2,1}_j & \cdots & PD^{2,ht}_j \\ \vdots & \ddots & \vdots \\ PD^{ht,1}_j & \cdots & PD^{ht,ht}_j \end{bmatrix} \end{aligned} \tag{11.3}$$

Where $\odot$ is the matrix multiplication operation. $PD^{i,z}_j$ is the j^{th} data sub-

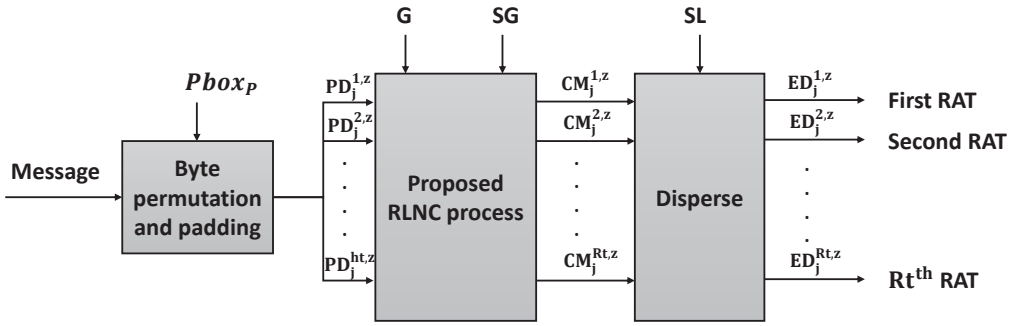


Figure 11.3: The proposed cryptographic solution that ensures data confidentiality and data availability

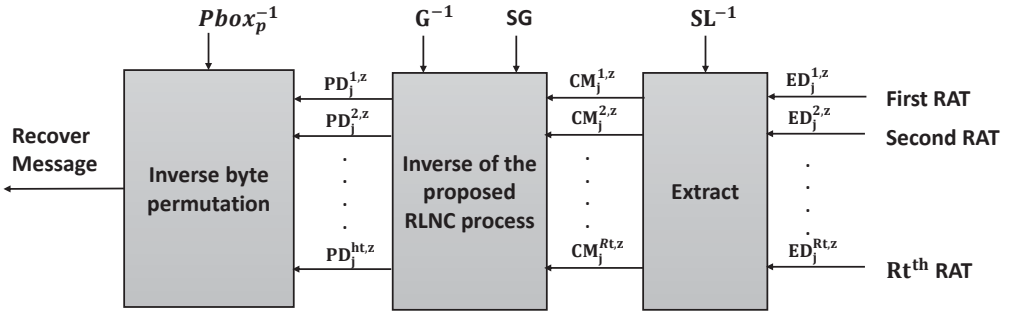


Figure 11.4: The inverse cryptographic solution at the legitimate destination

matrix ($1 \leq i \leq Rt$, $1 \leq j \leq \Omega$ and $1 \leq z \leq ht$), $G_{SG(j)}$ is the j^{th} pseudo-randomly selected secret encoding matrix based on SG, and CM_j is the resulting ciphered sub-matrix which has Rt rows and ht columns ($Rt \times ht$), unlike the original plaintext sub-matrix ($ht \times ht$).

Any $ht \times ht$ sub-matrix of the encoding matrices (G), having dimensions equal to $Rt \times ht$, should be invertible and should have coefficients in Galois field (2^8). This is achieved using the "Vandermonde" matrix with Galois field $GF(2^8)$ [294].

Here, it should be noted that the number of encoding matrices in the set G, should be at maximum equal to the number of data sub-matrices Ω, ($NG \leq \Omega$). In case of $\Omega = NG$, each data sub-matrix will be encoded using a unique encoding matrix $G_{SG(j)}$ ($1 \leq j \leq \Omega$), where a maximum level of protection can be reached, with additional overhead and memory at the initialization phase.

After performing encryption (permutation in addition to secret encoding), the ciphered sub-matrices enter the disperse block. At this stage, each row in the matrix, CM, should be transmitted on a different RAT. However, to increase the security of transmitted messages, the rows of the obtained ciphered sub-matrices (CM_j), are first shuffled (re-ordered) using the row selection table SL and then

transmitted. As a result, each row of ED_j ($Rt \times ht$), which is the resulting sub-matrix after row permutation, will be transmitted on a different RAT (Fig. 11.3).

The value of ht depends on the used application and the channel error, and it represents the minimum threshold for achieving reliable communication. This value should always be less than or equal to the number of active communication channels, Rt. More specifically, high data-rate applications require a value of ht equal or close to the number of active data channels (Rt), whereas, low data rate and error-sensitive applications withstand a lower value of ht (close to the minimum threshold). When $ht < Rt$, link reliability is achieved at the expense of a lower data rate, since $Rt - ht$ links can be used for data recovery in case of any link failure or poor channel conditions.

Data recovery: At the receiver, the same steps are followed but in a reversed order (Fig. 11.4). In addition, inverse permutation tables (SL^{-1} and $Pbox_P^{-1}$) and inverse encoding matrices (G^{-1}) are used. Upon the reception of data, the ciphered sub-matrix, ED_j ($Rt \times ht$), is de-shuffled (re-ordered) using the inverse pseudo-random row selection table SL^{-1} to obtain CM_j ($Rt \times ht$). Next, any $ht \times ht$ sub-matrix (ht rows) is extracted from CM_j, such that the rows corresponding to the ht best connections are selected. The resulting matrix is referred to as Ch_j ($ht \times ht$),

In order to recover the j^{th} plaintext sub-matrix, Ch_j is multiplied by its corresponding inverse decoding matrix, $Gh_{SG(j)}^{-1}$, using the following equation:

$$PD_j = Gh_{SG(j)}^{-1} \odot Ch_j,\ j = 1,\ 2,\ \ldots,\ \Omega \tag{11.4}$$

The set of encoding matrices G ensure the invertibility property. Therefore, any $ht \times ht$ sub-matrix of G, which is referred to as Gh, is an invertible matrix. Moreover, $Gh_{SG(j)}$ represents a collection of ht rows from the j^{th} secret encoding matrix, $G_{SG(j)}$, that is selected based on SG. $Gh_{SG(j)}^{-1}$ is its corresponding inverse.

This step mainly depends on the received data and the quality/availability of active connections. As mentioned earlier, the value of ht is chosen according to a specified threshold. Using Equation 11.4, PD_j ($ht \times ht$) is recovered by multiplying $Gh_{SG(j)}^{-1}$ ($ht \times ht$) with Ch_j ($ht \times ht$). The importance of the proposed decryption process is that it decreases the computational complexity and overcomes the issue of connection failure (availability attacks to some extent). This reduction in the overhead is attributed to the fact that only ht rows of CM_j are used in the decryption process, instead of using all Rt rows.

If any of the connections fails, data can still be recovered from other active connections ($Rt - ht$).

Afterwards, all of the recovered data sub-matrices are concatenated to obtain the original data matrix, $PD = PD_1||PD_2||\ldots||PD_\Omega$. Finally, the original message is obtained after reshaping and performing the inverse permutation operation using $Pbox_P^{-1}$.

For every input message, new cryptographic primitives are generated using the update permutation tables, $Pbox_{UP}$, $Pbox_{USG}$ and $Pbox_{USL}$. This guards against several existing attacks such as chosen/known plaintext/ciphertext attacks.

Discussion: The proposed scheme enhances not only the reliability and performance of communication channels, but also the security of transmitted data. In general, RLNC is a simple encoding scheme that allows the correct and efficient reception of data (close to optimal throughput). Using this approach, nodes are able to generate random and independent linear mappings of input to output data symbols over a finite field (coefficients chosen from a Galois field) [287].

However, when these coefficients become only known to the communicating users, data confidentiality is successfully achieved. In particular, random channel parameters are extracted from different channels (depending on the available RATs), and combined with a secret key only known to the communicating parties. The resulting dynamic key is used to derive the coefficients that will be used to encode and secure transmitted messages. Since communicating users share the same common physical channels (end-to-end communication systems), they will both be able extract the same channel-based nonce [27]. On the other hand, wireless communication channels are prone to many errors and threats that result from Doppler effects, noise, fading and availability attacks. Consequently, some links (channels) in multi-homed devices might suffer from bad quality, erroneous reception of data and failure. The proposed solution takes this issue into account, where a subset (according to a specific threshold) of the utilized sub-channels is sufficient to retrieve transmitted data, correctly. So, key generation/distribution, data confidentiality, and data availability are achieved using the proposed security solution.

11.2 Availability based on Binary Random Linear Network Coding

In this section, an efficient RLNC-based security solution is also proposed for multi-homed IoT systems (end-to-end), however, the proposed solution is performed at the bit level using novel binary RLNC matrices. The coefficients of these binary matrices are secured and modified in such a way that these matrices are only known to the communicating entities and have a determinant equal to one. This condition should be valid at all times in order to ensure the successful decoding of encoded data. Hence, a new approach for generating dynamic and invertible binary matrices is defined and verified, theoretically and experimentally. The coefficients of the encryption/encoding binary RLNC matrices are derived from a secret dynamic key, which is obtained by combining a secret session key

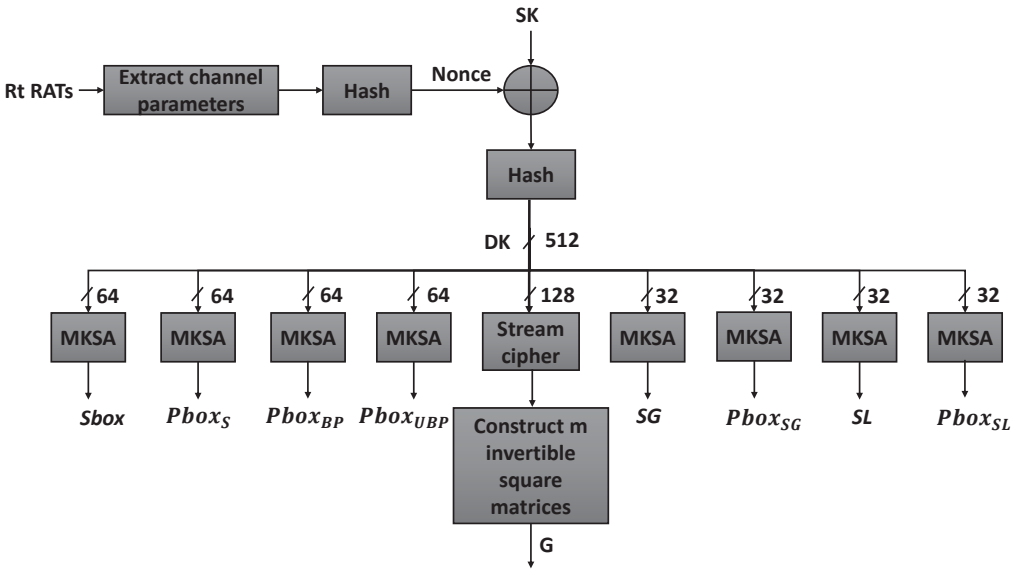


Figure 11.5: The proposed key distribution scheme

with a random channel-extracted nonce. The proposed scheme leverages the random and dynamic features of physical channels to increase the security level. The secret key is also used to generate the cryptographic primitives needed for the proposed ciphering/deciphering process such as substitution tables, block permutation tables and random selection vectors. By doing so, the proposed solution jointly improves the security and performance of multi-homed systems, since the binary field reduces the computational complexity of RLNC.

11.2.1 Proposed Binary Scheme

Here, DK is divided into nine sub-keys of different sizes (Fig. 11.5). The first two sub-keys (both of 64 bits) are used to generate the set of substitution boxes, $Sbox : Sbox_1, Sbox_2, \ldots, Sbox_{Rt}$, each with 256 elements, having values between 1 and 256, in addition to the update permutation tables for the substitution boxes, $Pbox_S : Pbox_{S_1}, Pbox_{S_2}, \ldots, Pbox_{S_{Rt}}$, having the same size. These are used to perform byte-substitution on input bytes before encryption/encoding. The first substitution box, $Sbox_1$, is generated, and all of subsequent substitution boxes are just shifted versions of it.

The second pair of keys, having a length of 64 bits, is used to derive Rt block permutation tables, $Pbox_{BP} : Pbox_{BP_1}, Pbox_{BP_2}, \ldots, Pbox_{BP_{Rt}}$, along with their update permutation tables, $Pbox_{UBP} : Pbox_{UBP_1}, Pbox_{UBP_2}, \ldots, Pbox_{UBP_{Rt}}$. This is done using the RC4 Modified Key-Scheduling Algorithm (M-KSA) [295]. These tables, each of size $(1 \times \omega)$, are used to perform block permutation for every Rt columns of encrypted/encoded data. The fifth sub-key, which is 128 bits long, is

Algorithm 7 Binary Multiplication Matrix algorithm

```
1: procedure Z =BINARY_MULTIPLICATION(X, Y)
2:     [m, n] = size(X)
3:     [n, q] = size(Y)
4:     for i ← 1 to m do
5:         for j ← 1 to q do
6:             tmp ← 0
7:             for k ← 1 to n do
8:                 tmp ← tmp ⊕ (X(i, k) ∧ Y(k, j))
9:             end for
10:            Z(i, j) ← tmp
11:        end for
12:    end for
13:    return Z
14: end procedure
```

used to obtain $\omega = NB_M/(Rt^2)$ binary invertible encryption/encoding matrices, $G : G_1, G_2, \ldots, G_\omega$ of size $(Rt \times Rt)$ and having values of 0's and 1's, where each unique matrix is randomly multiplied by an $(Rt \times Rt)$ binary sub-matrix of the input data (ω blocks/sub-matrices). Another pair of 32-bit sub-keys is used to generate a random matrix selector SG of size $(1 \times \omega)$, and an update permutation table for SG of the same size ($Pbox_{SG}$). SG has random values between 1 and ω and it is used to randomly select the encryption/encoding matrix which will be multiplied by each data sub-matrix. Similarly, two 32-bit sub-keys are used to derive the random link selector SL and the update permutation table $Pbox_{SL}$, both having a size of $(Rt \times 1)$ and values between 1 and Rt, which will be used to randomly select the rows that will be transmitted on each link/RAT.

It should be noted that $Pbox_S$, $Pbox_{UBP}$, $Pbox_{SG}$ and $Pbox_{SL}$ are used to update the cipher primitives, $Sbox$, $Pbox_{BP}$, SG and SL for each input message, respectively.

Binary RLNC secret coding: An input stream of NB_M bytes (before modulation) is divided into Rt rows and NB_M/Rt columns, forming a data matrix of size $(Rt \times (NB_M/Rt))$, where Rt is the number of available sub-channels. If NB_M is not divisible by Rt, padding is applied. Next, each of the NB_M/Rt bytes on each row is randomly substituted using one of the corresponding substitution boxes, $Sbox_1, Sbox_2, \ldots, Sbox_{Rt}$. The substituted bytes are then encrypted and encoded as follows:

- The matrix of substituted bytes is further divided into $\omega = (NB_M/Rt)/Rt$ sub-matrices, each having Rt rows and Rt columns (bytes).
- Using the random matrix selector SG, which contains random numbers

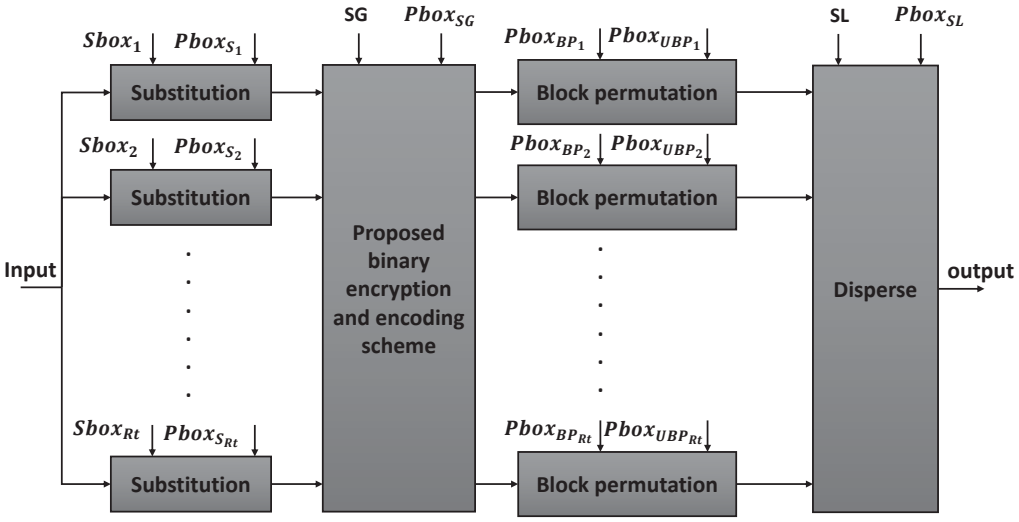


Figure 11.6: Proposed cryptographic scheme based on substitution, permutation and binary RLNC

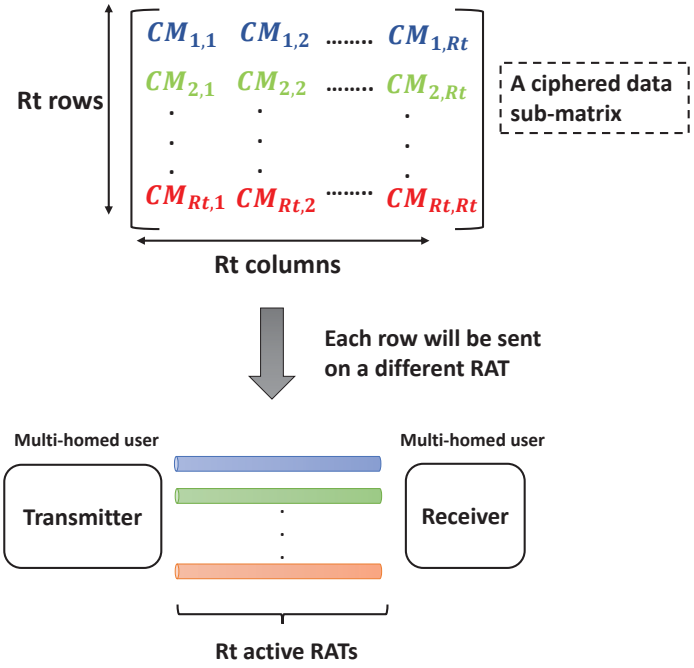


Figure 11.7: Proposed transmission mechanism over multiple RATs

between 1 and ω, each of the ω data sub-matrices ($Rt \times Rt$) is mixed with one of the invertible, unique, binary encryption/encoding matrices ($Rt \times Rt$)

from the set G. Here, binary matrix mixing/multiplication refers to row-column multiplication and byte-addition using the XOR operation. That is:

$$C_{4\times1} = G_{4\times4} \odot X_{4\times1}, \tag{11.5}$$

$$\begin{bmatrix} C_1 \\ C_2 \\ C_3 \\ C_4 \end{bmatrix} = \begin{bmatrix} 1101 \\ 0101 \\ 1000 \\ 0011 \end{bmatrix} \odot \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \end{bmatrix} = \begin{bmatrix} X_1 \oplus X_2 \oplus X_4 \\ X_2 \oplus X_4 \\ X_1 \\ X_3 \oplus X_4 \end{bmatrix}$$

where $\odot$ represents the binary matrix multiplication, $\oplus$ is the XOR operation, G is an example of a (4×4) binary encryption/encoding matrix, X is a (4×1) plaintext matrix which includes 4 bytes, X_1, X_2, X_3 and X_4, and C is the resulting ciphered/encoded matrix which has 4 rows and 1 byte column (8 binary columns). More specifically, the addition and subtraction arithmetic operations are replaced by the logical operation Exclusive-OR ($\oplus$), the multiplication operation is replaced by the logical AND operation ($\wedge$). This step is described in the pseudo code of Algorithm 7.

- The resulting ω encrypted/encoded matrices, CM_i $(1 \leq i \leq \omega)$, will also have Rt rows and Rt columns (byte values).

At the output of the encryption/encoding operation, each of the Rt rows undergoes block permutation. Specifically, the ω blocks on each row are randomly re-ordered and permuted using a different permutation table, $Pbox_{BP_1}, Pbox_{BP_2}$, $\ldots, Pbox_{BP_{Rt}}$. Each block consists of a single row with Rt bytes (Fig. 11.6). Finally, each row of the ω obtained encrypted/encoded sub-matrices is randomly transmitted on one of Rt available sub-channels (RATs) using the link selector (SL), which is an $(Rt \times 1)$ matrix having random values between 1 and Rt (Fig. 11.7).

For every new input message, new cryptographic primitives are generated by randomly permuting the old cryptographic primitives using $Pbox_S$, $Pbox_{UBP}$, $Pbox_{SG}$ and $Pbox_{SL}$.

In order to achieve data availability and link reliability, users can utilize a subset of the Rt rows, $ht < Rt$, to send legitimate data and the rest of the rows $(Rt - ht)$ to send redundant data. In this way, the receiver will be able to recover the data correctly, even when link failure occurs.

Generation of encryption/encoding RLNC binary matrices: The process of encoding in binary Galois field based on the matrix, G, has many advantages, some of which are the reduction of the required computational complexity,

increasing the throughput and decreasing the energy consumption. However, the generation of these matrices is not straightforward since they should possess specific properties in order to fit within the proposed scheme, and to ensure its efficient deployment in today's communication systems. Hence, in the following, a new scheme for generating binary invertible matrices is described, in addition to several matrix forms.

As mentioned previously, the encryption/encoding matrix G_i $(1 \leq i \leq \omega)$ should have Rt rows and Rt columns, where any G_i should be invertible. For this purpose, a generic $(Rt \times Rt)$ matrix, Gt_i, is first constructed using a binary square sub-matrix, Mu_i, and an identity matrix, Im_i, and then, G_i is derived. Note that Mu_i is a non-zero matrix having a size of $(\frac{Rt}{2} \times \frac{Rt}{2})$ and Im_i is an identity matrix of the same size. In order to ensure the invertibility of Gt_i, the determinant should always be equal to 1. The determinant, in the binary field, is equal to the determinant in the integer field modulo 2. The $(Rt \times Rt)$ binary invertible matrix is given by:

$$Gt_i = Gt_i^{-1} = \begin{bmatrix} Mu_i & Mu_i \oplus Im_i \\ Mu_i \oplus Im_i & Mu_i \end{bmatrix} \tag{11.6}$$

Theorem 1. *Gt_i is invertible and has a determinant equal to one for any sub-matrix Mu_i.*

Proof. A general form of a 2×2 matrix is shown below:

$$M = \begin{bmatrix} ab \\ cd \end{bmatrix}, det(M) = ad - bc. \tag{11.7}$$

In order to ensure the invertibility of M, $det(M)$ should always be equal to 1. It is assumed that d is equal to the value of a, hence, the determinant of M, $det(M) = 1 = a^2 - bc$, and $bc = a^2 - 1 = (a-1)(a+1)$. Consequently,

$$M = \begin{bmatrix} a & a+1 \\ a-1 & a \end{bmatrix}, det(M) = a^2 - a^2 + 1 = 1. \tag{11.8}$$

Replacing a with a matrix A, and 1 with an identity matrix Im, would also result in an invertible block matrix, Q. In general, the determinant of an invertible block

matrix, Q, consisting of four sub-matrices A, B, C and D, is equal to

$$Q = \begin{bmatrix} AB \\ CD \end{bmatrix} = \begin{bmatrix} A & A + I_m \\ A - I_m & A \end{bmatrix}, \tag{11.9}$$

$$\begin{aligned} det(Q) &= det(A) \times det(D - CA^{-1}B) \\ &= det(A) \times det(D - CBA^{-1}) \\ &= det(A) \times det(A - A^2A^{-1} + I_m^2 \times A^{-1}) \\ &= det(A) \times det(A - A + A^{-1}) \\ &= det(A) \times det(A^{-1}) \\ &= det(A \times A^{-1}) \\ &= det(I_m) \\ &= 1, \end{aligned} \tag{11.10}$$

where $C \times B$ is equal to $A^2 - Im^2$, $A^2 \cdot A^{-1}$ is equal to A and $A^{-1} \cdot Im^2$ is equal to A^{-1}.

Now, replacing the integer matrix A with a binary matrix Mu_i, having $Rt/2$ rows and $Rt/2$ columns, yields:

$$Gt_i = \begin{bmatrix} Mu_i & Mu_i \oplus Im_i \\ Mu_i \oplus Im_i & Mu_i \end{bmatrix}, \tag{11.11}$$

where Im_i is an identity matrix having the same dimensions as Mu_i. Here, it should be noted that the addition and subtraction arithmetic operations are replaced with the logical XOR ($\oplus$) operation, and the multiplication operation is replaced by the logical AND operation ($\wedge$).

Since, Gt_i is a binary block matrix, the determinant of this matrix is written as:

$$\begin{aligned} det(Gt_i) &= det(A) \wedge det(D \oplus C \wedge A^{-1} \wedge B) \\ &= det(A) \wedge det(D \oplus C \wedge B \wedge A^{-1}) \\ &= det(Mu_i) \wedge det(Mu_i \oplus Mu_i^2 \wedge Mu_i^{-1} \oplus \\ &\quad Im_i^2 \wedge Mu_i^{-1}) \\ &= det(Mu_i) \wedge det(Mu_i \oplus Mu_i \oplus Mu_i^{-1}) \\ &= det(Mu_i) \wedge det(Mu_i^{-1}) \\ &= det(Mu_i \wedge Mu_i^{-1}) \\ &= det(Im_i) \\ &= 1, \end{aligned} \tag{11.12}$$

where $C \wedge B = (Mu_i^2 \oplus I_m^2)$, $Mu_i^2 \wedge Mu_i^{-1} = Mu_i \wedge Mu_i \wedge Mu_i^{-1} = Mu_i$ and $Mu_i^{-1} \wedge Im_i^2 = Mu_i^{-1}$. Therefore, Gt_i is invertible for any sub-matrix Mu_i, and $Gt_i = Gt_i^{-1}$. □

Hence, Gt_i always has a determinant equal to one, and Gt_i^{-1} is equal to Gt_i, which reduces the delay of decoding.

In order to obtain G, which is a set of $(Rt \times Rt)$ diffusion matrices, ω Mu_i matrices are first generated. The coefficients of each Mu_i matrix, which is used to derive one unique encryption/encoding matrix (G_i for all $i \in [1, \omega]$), are generated using DK. Using the proposed matrix generation technique and the generic matrix form, one can ensure that all of the obtained encryption/encoding matrices are secure and invertible. Consequently, this enhances the performance and security of end-to-end wireless communication systems.

Decryption and data recovery: At the receiver, the same steps are followed but in a reversed order. In addition, the inverse cipher primitives ($Sbox^{-1}$, SG^{-1}, $Pbox_{BP}^{-1}$, and SL^{-1}) are used. First, the ciphered data received on each of the Rt rows are de-shuffled using the inverse link selector SL^{-1} (for all ω submatrices). Next, inverse block permutation is performed to obtain the ciphered matrices CM_i $(1 \leq i \leq \omega)$, using the set $Pbox_{BP}^{-1}$.

Afterwards, CM_i $(Rt \times Rt)$ and G_i^{-1} (which is equal to G_i) $(Rt \times Rt)$ are multiplied using Equation 11.5. The resulting bytes are substituted using $Sbox^{-1}$ to obtain the transmitted plaintext bytes.

The advantage of the proposed decryption process is that it decreases the computational complexity and overcomes the issue of connection failure due to fading, Doppler effects or availability attacks. In particular, original data are encrypted/encoded at the byte/bit level using simple and lightweight cryptographic operations, which jointly, enhances the security and performance of multi-homed systems.

For every input message, new cipher primitives are generated using the update permutation tables, $Pbox_{UBP}$, $Pbox_S$, $Pbox_{SG}$ and $Pbox_{SL}$. This guards against many attacks such as chosen/known plaintext/ciphertext attacks.

Discussion: The proposed scheme aims at jointly enhancing the security and performance of multi-homed systems using RLNC. More specifically, random physical properties, which are extracted from multiple end-to-end wireless channels, are combined with a secret session key to generate a dynamic key [27]. The resulting key is then used to derive the needed cipher primitives, which include a substitution table, a block permutation table, the binary RLNC matrices and two random selection tables. By doing so, the cryptographic primitives used in the ciphering and encoding process become only known to the communicating entities, and hence, data confidentiality is successfully achieved. Unlike the work presented in the literature, this paper presents a scheme that secures and encodes binary input data, which is a hard challenge. In general, RLNC a simple network coding scheme that allows intermediate nodes to generate random and independent linear mappings of input to output data symbols over a finite field (coefficients chosen from a Galois field) [287]. Consequently, this technology im-

proves the network performance substantially. On the other hand, ensuring the inveritibility of binary RLNC matrices is a crucial condition for the proposed deciphering process, therefore, original RLNC matrices are modified in such a way that these matrices are invertibile and have a determinant equal to one. Moreover, a general form for RLNC binary matrices is proposed and verified theoretically. Another aspect of the proposed scheme is that it overcomes link errors and availability attacks. More specifically, wireless channels are subjected to many errors that result from Doppler effects, noise, fading and jamming. Consequently, some links (sub-channels) in multi-homed systems might suffer from bad quality, erroneous reception of data or failure. Using the proposed solution, communicating entities can utilize a subset of the available RATs/links to carry redundant data, which enhances the reliability of wireless communication (transmitted data are correctly recovered).

11.3 Security Analysis and Cryptanalysis of the Proposed Schemes

In this section, the robustness of the proposed schemes is evaluated and assessed against well-known attacks, which include statistical attacks, linear/differential attacks, chosen/known plaintext/ciphertext attacks, brute force attacks and key-related attacks.

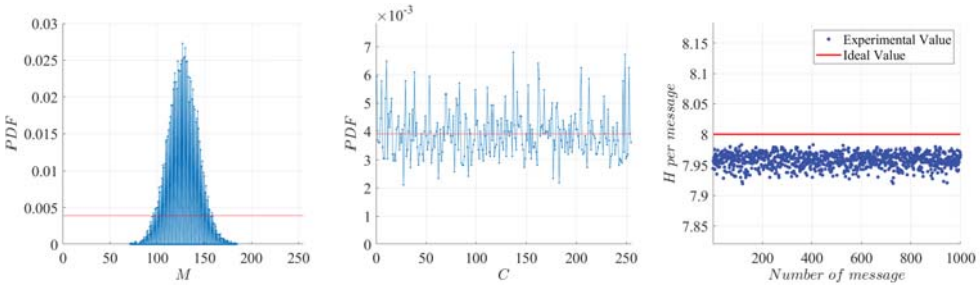
The proposed schemes are considered public and the cryptanalyst is assumed to have complete knowledge regarding all the required steps, but none regarding the dynamic key that is used for data encryption.

11.3.1 Statistical Attacks

In order to resist statistical attacks, ciphered data should have a high randomness degree. For this purpose, several randomness tests, which include entropy analysis, Probability Density Function (PDF), correlation, and recurrence, are presented to prove the proposed schemes' robustness against statistical attacks.

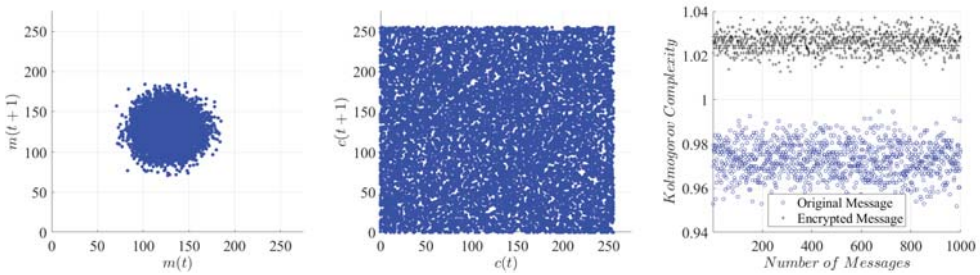
In the simulation tests, the proposed security schemes are considered as a black box. A set of initial messages, having a size of 15000 bytes, are randomly chosen. These messages follow a normal distribution with mean equal to 128 and standard deviation equal to 16. In the following, the above mentioned properties are analyzed.

Uniformity using Probability Density Function (PDF): One way to quantify the uniformity property is by plotting the PDF of encrypted symbols and verifying visually the type of the distribution. This simple method is sufficient to prove that the proposed schemes have a good mixing level (randomness).



(a) PDF of the original message (b) PDF of the obtained ciphertext (c) Entropy of the ciphertext message

Figure 11.8: The probability density function of the (a) original input and the (b) ciphered/encoded output for 1000 iterations and using 256 QAM modulation (using binary RLNC). (c) The entropy of the ciphertext (using binary RLNC)

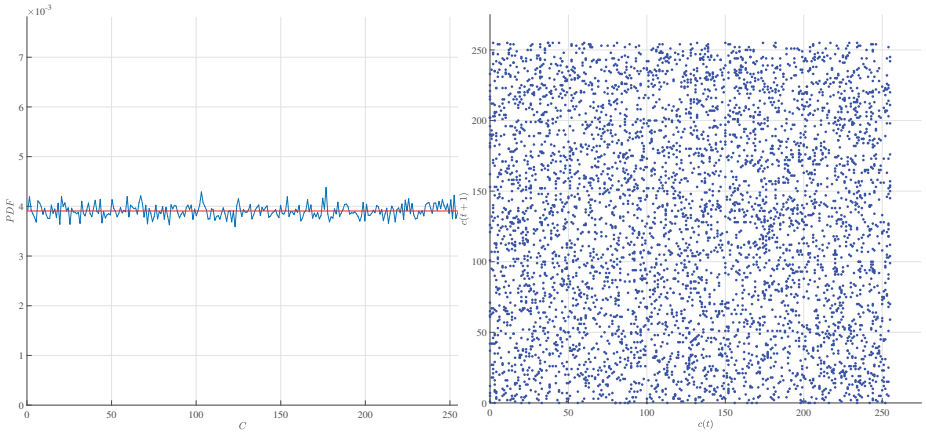


(a) Recurrence of the original message (b) Recurrence of ciphertext (c) Kolmogorov complexity of binary ciphertext

Figure 11.9: The recurrence plots of the (a) original message and its (b) encrypted version (using binary RLNC)

Figures 11.8a, 11.10a and 11.8b represent the PDF of the original and encrypted data using integer and binary RLNC, respectively. The plots clearly verify that the original data is normally distributed, while encrypted symbols (using integer and binary RLNC) are uniformly distributed (random output). In order to validate this result, the entropy test is performed. Specifically, a uniform probability (distribution) represents maximum uncertainty, which yields to maximum entropy (the entropy of the obtained encrypted symbols should be equal to 1 and 8 at the bit and byte levels, respectively).

Entropy: In principle, the entropy values of encrypted symbols at the byte level should be equal to 8, which corresponds to maximum uncertainty (randomness). Figures 11.11a and 11.8c show that the encrypted symbols (byte level) have an entropy value close to the desired value of 8, using both schemes. There-



(a) PDF of encrypted message (b) Recurrence of encrypted message

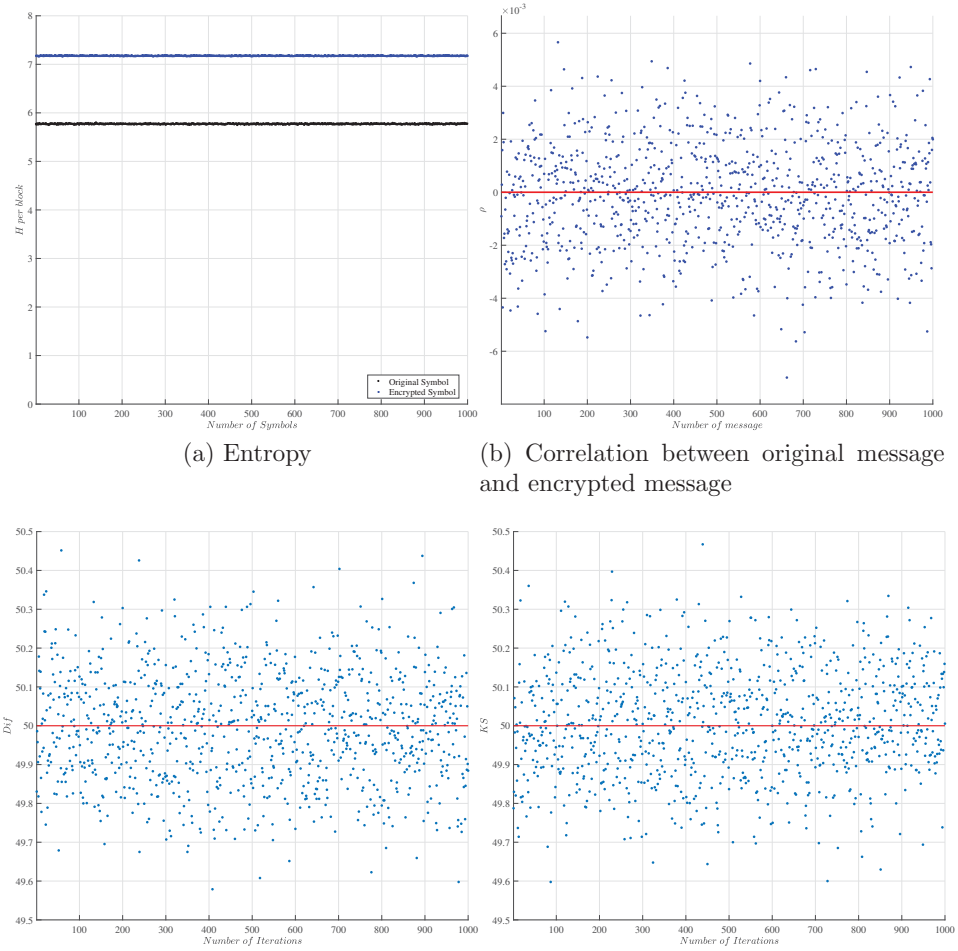
Figure 11.10: The PDF and recurrence of the original and encrypted messages (using integer RLNC), respectively

fore, the obtained results prove that uniform distribution is ensured among the encrypted messages.

The Kolmogorov entropy (complexity) of the original and encrypted messages (using binary RLNC) has also been plotted. The Kolmogorov complexity of the original data is less than one, whereas that of the encrypted data is above one (Fig. 11.9c). This confirms that the proposed scheme produces encrypted symbols that are, indeed, uniform and random.

Recurrence: The recurrence function measures the correlation between a specific symbol and its delayed version. Typically, encrypted symbols should have a highly scattered recurrence plot, which proves that the tested scheme has the proper randomness level. The presented results in Fig. 11.10b prove that the proposed integer RLNC scheme (secret encoding process with dynamic byte permutation) hides any clear pattern (scattered over the entire space), hence, it is very difficult to get useful information regarding the original message. Similarly, Fig. 11.9b shows that the recurrence plot of the encrypted data using the binary RLNC scheme, spans the entire cipher symbol space (randomized and scattered). In contrast, the recurrence points of the original data are grouped within a specific area (Fig. 11.9a) (not random).

Correlation: In general, encrypted data should greatly differ from the original data for a cipher scheme to be considered secure. More specifically, the encrypted data should have very low redundancy and correlation. First, the correlation coefficient between the original segment and its corresponding encrypted segment is measured, and then the correlation coefficient between the original



(a) Entropy

(b) Correlation between original message and encrypted message

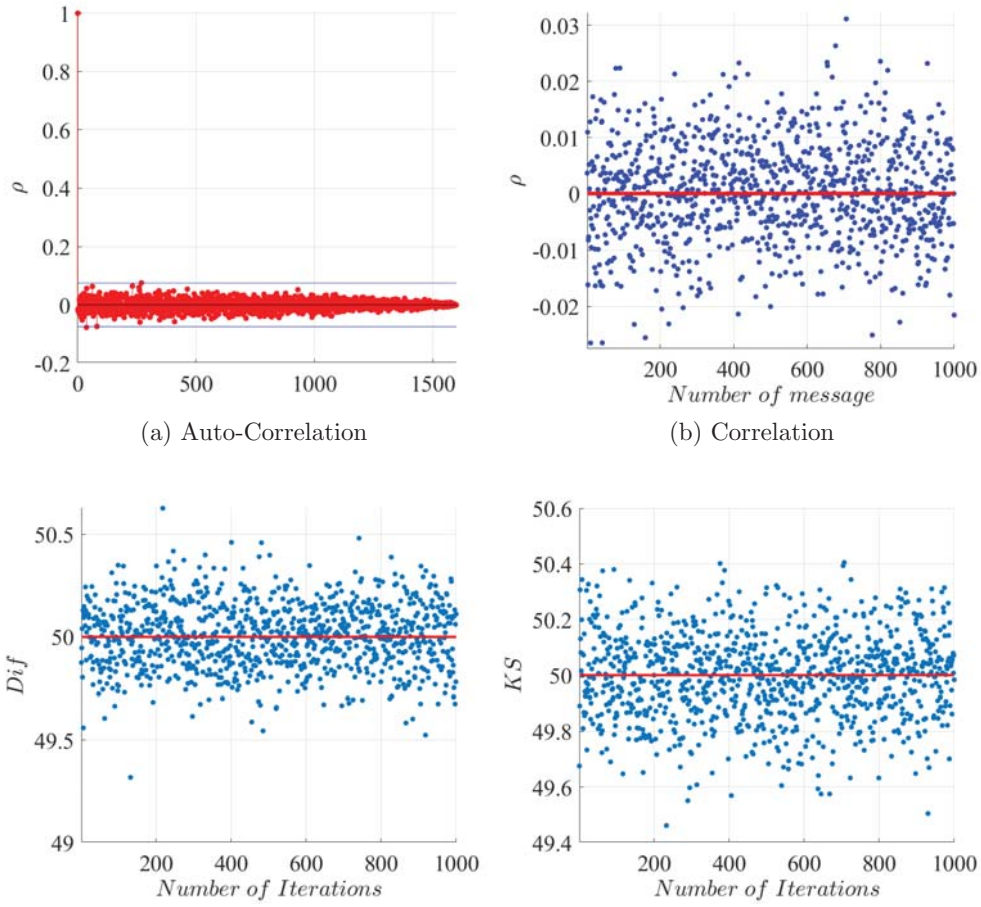
(c) Difference

(d) Key sensitivity

Figure 11.11: The variation of entropy, correlation, difference and key sensitivity, respectively (using integer RLNC)

segment and different encrypted segments is presented. Figure 11.11b and Table 11.1, show the average correlation coefficient between the original segments and their encrypted versions (using the first scheme), for 10,000 different secret matrices having $ht = 4$ and $Rt = 8$. The presented results indicate that no detectable correlation exists between the original and the encoded/encrypted segments (correlation is close to zero). Additionally, the correlation coefficient corresponding to all of the encoded/encrypted segments is shown in Table 11.2 in a matrix form. Results show that the average of the correlation coefficient matrix for different encrypted segments is close to zero, which verifies that no detectable correlation exists between the different encrypted segments.

Additionally, the presented results of the binary RLNC scheme in Fig. 11.12b



(a) Auto-Correlation

(b) Correlation

(c) Difference

(d) Key sensitivity

Figure 11.12: The sample (a) auto-correlation function for a random dynamic key, (b) inter-correlation, (c) independence and (d) key sensitivity results for 1000 random dynamic keys (using binary RLNC)

indicate that the cross-correlation coefficient ranges between $\{-0.02, 0.03\}$ and has a mean of 0, which is the desired outcome. Also, the auto-correlation of the obtained ciphertext confirms the high randomness degree of the proposed scheme since all values are centered around zero (Fig 11.12a).

11.3.2 Linear/Differential Attacks

Here, the adversary exploits the relationship between encrypted symbols to recover the original data. In the proposed schemes, encrypted symbols are highly uncorrelated since the cipher primitives are regularly updated. Consequently,

Table 11.1: Correlation coefficient between the original and the encoded/encrypted segments (using integer RLNC) for a random dynamic key with ht =4 and Rt =8

	ED_1	ED_2	ED_3	ED_4	ED_5	ED_6	ED_7	ED_8
O_1	0.0309	0.0219	0.0270	0.0029	-0.0351	0.0068	0.0003	0.0052
O_2	0.0189	-0.0141	0.0063	0.0181	-0.0031	-0.0249	0.0003	-0.0189
O_3	-0.0463	0.0034	0.0047	0.0220	-0.0193	0.0088	0.0041	-0.0304
O_4	-0.0173	0.0036	-0.0204	-0.0061	0.0243	-0.0084	0.0074	0.0249

Table 11.2: Correlation coefficient among encoded/encrypted segments (using integer RLNC) for a random dynamic key with ht =4 and Rt =8

	ED_1	ED_2	ED_3	ED_4	ED_5	ED_6	ED_7	ED_8
ED_1	-	0.0218	-0.0375	0.0077	0.0042	0.0130	-0.0131	0.0080
ED_2	0.0218	-	-0.0081	-0.0120	0.0171	0.0176	0.0032	0.0101
ED_3	-0.0375	-0.0081	-	-0.0178	-0.0119	-0.0145	0.0017	-0.0171
ED_4	0.0077	-0.0120	-0.0178	-	-0.0042	-0.0088	-0.0126	0.0058
ED_5	0.0042	0.0171	-0.0119	-0.0042	-	0.0036	0.0065	-0.0231
ED_6	0.0130	0.0176	-0.0145	-0.0088	0.0036	-	-0.0138	0.0072
ED_7	-0.0131	0.0032	0.0017	-0.0126	0.0065	-0.0138	-	0.0051
ED_8	0.0080	0.0101	-0.0171	0.0058	-0.0231	0.0072	0.0051	-

chosen/known plaintext/ciphertext attacks, single data failure and accidental key disclosure are avoided. The difference test is presented to prove the independence of encrypted data.

Independence: To satisfy the independence criterion, the difference value should always be close to 50%. This is based on the fact that any bit difference in the original data should change at least half of the bits at the output. The difference between original and encrypted segments using the first approach is evaluated at the bit level. The percent variation of the bit difference between the original and the corresponding encrypted segments (using the first approach) for $1,000$ random dynamic keys is shown in Fig. 11.11c. This result indicates that the percentage difference is always close to 50%. Hence, the independence

Table 11.3: Independence between original fragments (column index) and encrypted ones (row index) Rt=8 (using binary RLNC)

Indep	**O1**	**O2**	**O3**	**O4**	**O5**	**O6**	**O7**	**O8**
E1	49.9531	50.0625	50.4297	49.3906	51.0234	50.0234	50.4922	49.5078
E2	49.6719	50.2500	49.3672	49.8750	50.2422	49.3203	49.9766	49.4922
E3	50.1016	49.9766	49.9219	50.3047	50.3281	49.5781	50.0938	49.4219
E4	49.9609	50.7266	49.8438	50.0859	50.3438	49.7500	50.4687	51.0000
E5	49.1641	49.6328	50.2344	50.0391	49.3750	48.7969	51.1094	50.2969
E6	49.7578	50.3516	50.6094	50.9922	49.9063	50.5313	50.1094	50.5938
E7	50.0859	49.6641	49.4531	50.1328	50.1406	49.9063	50.4063	48.9375
E8	50.3047	50.1172	51.4531	49.6484	50.1875	49.7656	49.6719	50.2500

Table 11.4: Independence among encrypted fragments (using binary RLNC)

Indep	**E1**	**E2**	**E3**	**E4**	**E5**	**E6**	**E7**	**E8**
E1	-	49.4063	50.2734	49.9219	50.3359	50.1953	50.6797	49.3359
E2	49.4063	-	49.8984	49.9688	50.1953	50.3203	50.4609	50.7891
E3	50.2734	49.8984	-	49.4609	49.8281	50.3125	50.9844	49.9844
E4	49.9219	49.9688	49.4609	-	50.4453	50.4609	49.9609	50.2578
E5	50.3359	50.1953	49.8281	50.4453	-	50.1875	48.6094	49.9375
E6	50.1953	50.3203	50.3125	50.4609	50.1875	-	50.1563	49.2969
E7	50.6797	50.4609	50.9844	49.9609	48.6094	50.1563	-	49.6094
E8	49.3359	50.7891	49.9844	50.2578	49.9375	49.2969	49.6094	-

Table 11.5: Key sensitivity between two sets of encrypted fragments (One bit difference between both dynamic keys (DK and DK') and for Rt =8 (using binary RLNC)

Indep	**E1**	**E2**	**E3**	**E4**	**E5**	**E6**	**E7**	**E8**
E1'	49.7500	49.9531	49.8203	49.4609	50.0234	50.5938	50.3359	49.6250
E2'	50.2813	49.8438	50.0703	50.2891	51.2266	50.0000	50.5703	50.0000
E3'	50.9141	50.4922	49.7813	50.8125	50.7031	49.6797	49.8281	49.6953
E4'	49.9531	49.8281	49.4297	49.3359	49.8203	50.1406	49.8203	50.1406
E5'	50.4609	49.8672	50.0781	49.7813	49.7656	49.9922	50.0156	49.3047
E6'	49.9297	50.1953	50.0313	50.1094	49.8438	49.6016	50.0625	49.3516
E7'	49.9453	50.6953	50.0938	50.0469	50.2500	50.2891	49.7344	49.7578
E8'	49.5078	49.3359	50.2031	50.1406	50.3281	50.3359	49.8594	50.0703

criteria is satisfied.

In Fig. 11.12c, the results show that the plotted difference values, between the original and the encrypted messages using the second approach, have a mean equal to the desired value, 50%. Moreover, the independence property has been assessed at the fragment level (row level of an $Rt \times Rt$ block matrix) for a fixed value of $Rt = 8$. The results, in Table 11.3, complement those in Fig. 11.12c, where the independence values range between 49% and 51%. This is also true for Table 11.4, where all of the difference values, between the encrypted fragments, are very close to 50%.

11.3.3 Weak Keys and Key-Related Attacks

The proposed key generation technique in both schemes derives a dynamic key by combining channel-based information with a secret session key that is only known to the communicating entities. Using this key, the cryptographic primitives, needed for the ciphering process, are obtained and updated. In order to assess the schemes' security against key-related attacks, the sensitivity of DK is evaluated.

Key sensitivity: A cipher algorithm is considered secure and robust against related key attacks if the dynamic key ensures good sensitivity. In particular, any change in the key should result in completely different set of encryption/encoding matrices, cipher primitives, and consequently, different encrypted/encoded segments. For this purpose, two dynamic secret keys are used in this test: DK_1 and

DK_2, which differ by only one bit. In Fig. 11.11d, the key sensitivity test is done using 1000 random dynamic keys. The presented results show that key sensitivity is always close to the desired value, which is 50%. This proves that the proposed integer RLNC scheme is sensitive and robust against any slight change in the secret key or nonce (low standard deviation equal to 0.3128), thus, attaining the required key sensitivity level and ensuring high resistance against different types of key-related attacks.

Also, Fig. 11.12d plots the key sensitivity values corresponding to $1,000$ randomly generated dynamic keys using the binary RLNC approach. The results clearly show that the key sensitivity value is always close to 50%. Table 11.5 evaluates the key sensitivity values between two sets of encrypted fragments, both having a dimension of (8×8). Again, the key sensitivity property is also confirmed in this study.

Message Sensitivity: The proposed solutions have a dynamic structure and depend on variable factors that change frequently (channel nonce). Consequently, the obtained ciphertext satisfies the necessary avalanche effect and the desired security level, since for every input message, new cipher primitives are generated using the proposed update processes. This, in turn, reduces the error propagation since only one round is required (unlike conventional multi-round structures).

11.3.4 Brute Force Attacks

The size of the secret session key (SK) can be equal to 128, 196 or 256, and the proposed dynamic key (DK) has a length of 512 bits which is sufficient to guard against brute force attacks.

All of the presented results prove that a communication session, between any two legitimate entities, is secured using the proposed schemes. The dynamic secret key is based on multiple factors, secret and channel-based, which makes the schemes robust against *passive* and *active* adversaries.

11.4 Performance Evaluation of the Proposed Schemes

This section analyzes the proposed schemes in terms of computational complexity, execution time, communication overhead, efficiency, transparency, flexibility and propagation of error, in order to prove their efficient deployment in today's networks.

11.4.1 Computational Complexity

The proposed schemes are based on a single round and simple operations, mainly matrix multiplication, substitution and permutation, to secure and encode integer and binary input data.

In the first scheme, the input data is divided into $(NB_M/ht)/ht$ sub-matrices, each having ht rows/segments and ht columns. Then, each data sub-matrix is permuted and multiplied by its corresponding (unique) encryption/encoding matrix. As for the second scheme, data is first divided over Rt rows, and then each row is divided into Rt columns. Hence, one can strongly benefit from parallelization to reduce computational complexity, execution time and associated delays, as each sub-matrix can be coded and recovered independently from the others. Moreover, the proposed schemes require only one round to secure transmitted data, instead of multiple rounds as the case of most standard ciphers (Advanced Encryption Standard (AES)) [296].

In order to validate the efficiency and simplicity of the proposed solutions, both schemes are evaluated in comparison to the AES scheme according to several parameters such as the associated delays. To assess the total computational delay, different types of delays are identified as follows:

1. T_S denotes the time required by the substitution operation for one message block (iteration rounds).
2. T_P denotes the time required by the permutation operation for one message block.
3. T_D represents the time required by the mix-column operation in AES.
4. T_{ML} denotes the time required by the matrix multiplication operation for a sub-matrix.
5. T_{SL} denotes the time required to shuffle (select) input blocks.
6. T_{xor} denotes the time required by the XOR operation.
7. T_{SR} denotes the time required by the shift rows operation in AES.

The total Computational Delay (CD), required to encrypt one input block using the integer RLNC scheme (first scheme), with and without the chaining operation mode, is:

$$CD_{RLNC_I} = T_P + T_{ML} + T_{SL}. \tag{11.13}$$

The total Computational Delay (CD), required to encrypt one input block using the binary RLNC scheme, with and without the chaining operation mode, is:

$$CD_{RLNC_B} = T_S + T_P + T_{ML} + T_{SL}. \tag{11.14}$$

In contrast, the total CD required to encrypt one block using the standard AES [297] is:

$$CD_{AES} = r_o T_S + (r_o + 1)T_{xor} + (r_o - 1)T_D + r_o T_{SR}, \tag{11.15}$$

where r_o represents the number of rounds in AES. The minimum value of r_o is 10 for a 128-bit secret key, hence, the minimum AES computation delay is given by:

$$CD_{AES(r_o=10)} = 10T_S + 11T_{xor} + 9T_D + 10T_{SR}. \tag{11.16}$$

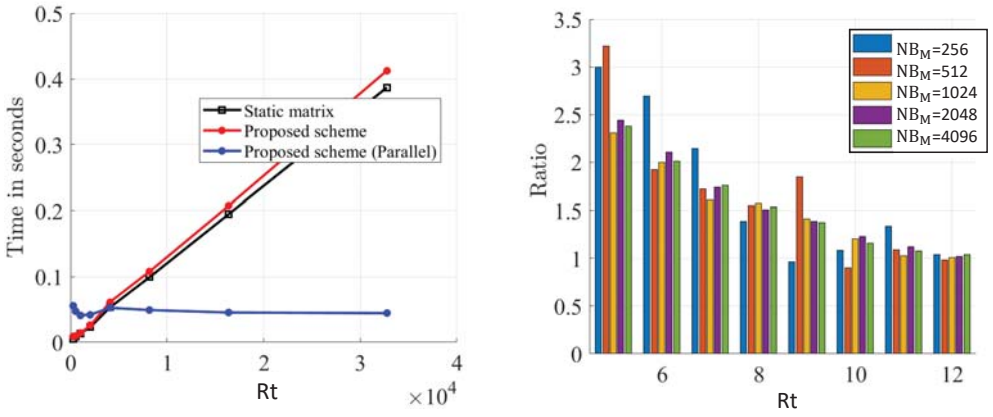
Clearly, the AES computational delay is larger than that of the proposed schemes, which avoid multi-diffusion operations towards reducing the required delay. Specifically, both schemes require only one diffusion operation in comparison to the AES algorithm, which includes 9 diffusion operations.

As a result, the required computation complexity of the proposed schemes, is relatively low compared to that of existing solutions which employ standard cryptographic algorithms [260].

11.4.2 Execution Time

For the first RLNC scheme, the average execution time (in seconds), needed to encrypt/encode a message with a variable length NB_M, is plotted in Fig. 11.13a for $Rt = 8$ and $ht = 4$ using a static matrix and the proposed modified matrix. These calculations are conducted using the following software and hardware environment: Matlab on 2018 and micro-computer Intel(R) Core(TM) i7-7600U CPU of 2.80GHz (4 CPUs), 2.9GHz with 2 GB RAM Intel, under Windows 10 Professional 64-bit. Clearly, the variation of the average execution time is linear when using only one encoding matrix on the entire input data and also when using the proposed solution with a set of secret encoding matrices ($G = G_1,\ G_2,\ \ldots,\ G_{NG}$) on a set of data sub-matrices ($ht \times ht$). In fact, the required execution time of the proposed scheme without parallel computing is really close to the conventional approach (with a static matrix). Using the linear interpolation algorithm, the required time as a function of NB_M is $1.0521e^{-05} \times NB_M + 0.0037958$ for $Rt = 8$ and $ht = 4$.

Moreover, the ratio between the proposed encoding solution without parallel computing and the static encoding scheme (conventional RLNC) for the different values of Rt is shown in Fig. 11.13b with $ht = 4$. The obtained results clearly show that the proposed method has a sufficiently low overhead ratio and it is close to 1.5 for $Rt = 2 \times ht = 8$. In addition, higher values of Rt results in less overhead compared to the original matrix. Accordingly, the proposed scheme without parallel computing attains minimum ratio compared to the static traditional approach for a high value of Rt, while a lower value of Rt requires more execution time. This means that for a high value of Rt, the reduction in the execution time is significant.



(a) Execution Time versus NB_M (b) Ratio versus Rt

Figure 11.13: (a) The variation of execution time as a function of NB_M and (b) the ratio of the execution time of the proposed integer RLNC scheme compared to the static approach

In addition, a comparison is performed with the proposed scheme while performing parallel computing, and obviously, the proposed approach outperforms the static matrix approach, which cannot be paralleled. The results clearly indicate that parallel computing with 2 Workers reduces the execution time, further.

This means that the proposed scheme with parallel computing has a (stable) low execution time for different dimensions of sub-matrix Rt. As such, a significant reduction in terms of execution time is ensured by using lookup tables for the multiplication and division operations.

On the other hand, Fig. 11.14 illustrates the time needed to execute the binary RLNC scheme and the time savings with respect to the traditional RLNC mechanism. The figure shows the ratio of execution time of the binary RLNC solution over that of conventional RLNC. It is evident that binary RLNC requires a lower execution time for all matrix dimensions. In particular, a time reduction that ranges between 20% ($Rt > 20$) and 40% ($Rt < 20$) is attained. Consequently, the proposed solutions are suitable for multi-homed IoT systems and applications since simple ciphering/encoding operations are employed.

11.4.3 Storage/Communication Overhead

For the first integer RLNC scheme, the size of the matrices obtained after ciphering is $Rt \times ht$ ($ht \leq Rt$), whereas the size of matrix needed to recover the original data is $ht \times ht$. Therefore, the encryption/encoding procedure produces data overhead equal to $((Rt - ht) \times NB_M)$ bytes, in order to benefit from network coding in terms of availability and resistance against channel errors. In order

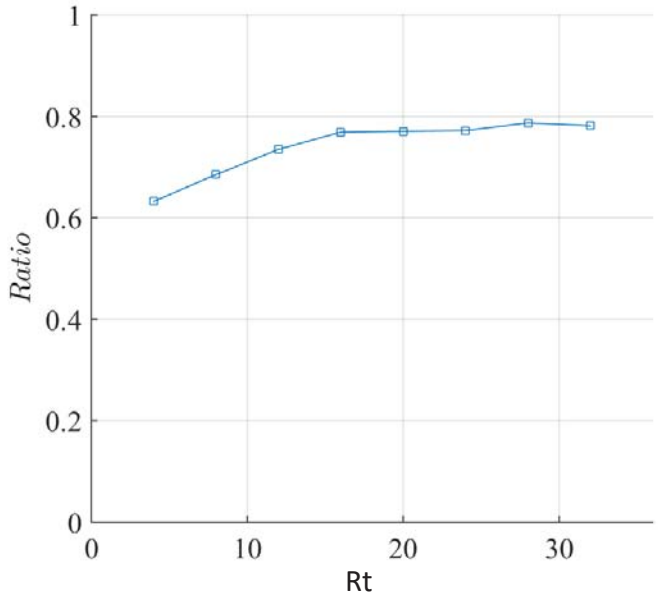


Figure 11.14: The percentage reduction in execution time using binary RLNC in comparison to the conventional integer RLNC

words, $(Rt - ht)$ redundant encoded segments (rows) are produced and transmitted to prevent data loss due to damage or alteration. When the value of ht is close to Rt, the communication overhead and availability level, both, decrease. This means that a trade-off exists between communication overhead and availability level. In contrast, whenever ht is less than Rt, high data availability is achieved, and high communication overhead is also introduced. The choice of ht depends on the communication technology, where a good balance between data availability and communication overhead should always be ensured.

In the case of the second binary RLNC scheme, this overhead will be only introduced if users choose to send redundant data to ensure data availability $(Rt > ht)$. A subset $(Rt - ht)$ of the Rt available sub-channels will be utilized to send redundant data. In this case, the encoding process will generate a data communication overhead of $((Rt - ht) \times \frac{NB_M}{ht})$ bytes. However, this overhead will increase the availability degree and the resistance against link failure and channel errors. These $(Rt - ht)$ redundant rows, each having a dimension equal to $1 \times ht$, are transmitted to overcome data loss, damage or alteration. Specifically, source data is encoded at the sub-matrix level. The dimension of each sub-matrix, before and after the RLNC binary encoding/decoding process, will be $(ht \times ht)$. A set of dynamic binary encoding matrices (in this case having a size equal to $ht \times ht$) is used instead of a static one to enable the parallelism of the encoding/decoding process and to complicate the crypatanalysis process. The remaining $(Rt - ht)$

sub-channels are used for the transmission of redundant rows to surpass lossy sub-channels. Whenever ht is close to Rt, the communication overhead and availability level would decrease. In contrast, when Rt is greater than ht, high data availability is attained and high communication overhead is also introduced.

11.4.4 Efficiency

The proposed schemes utilize simple and hardware-efficient operations, mainly byte substitution, byte permutation, block permutation and matrix multiplication, to secure resource-limited devices in multi-homed IoT systems. These operations are executed at the source and destination, where no additional operations are required at intermediate nodes such as aggregators, if they exist. Consequently, the proposed solutions provide multiple security services, such as data confidentiality and data availability, in a simple and efficient manner and with acceptable overhead and complexity.

11.4.5 Transparency

The proposed solutions preserve the homomorphic properties of encrypted/encoded data due to the nature of byte permutation and matrix encoding operations. Hence, this proves that the proposed schemes are transparent to any intermediate coding process (if it exists), done by intermediate nodes such as aggregators.

11.4.6 Flexibility and Scalability

Another important property of the proposed solutions is that both respond to any change in the number of communication channels, Rt, as well as the message length, in a fluent and smooth manner.

11.4.7 Error Propagation

In principle, all cryptograhic schemes should ensure low error propagation so that these schemes are considered efficient and deploy-able. This is a hard challenge since these errors mainly result from interference and noise in transmission channels, leading to the destruction of transmitted data. Consequently, there exists a trade-off between the avalanche effect (security) and error propagation [267]. In the proposed schemes, byte-error is limited to the erroneous row/segment in the received matrix, and which corresponds to the noisy sub-channel/RAT. In order to overcome this issue, only a subset, ht, of the available Rt rows ($ht \in Rt$ rows), that corresponds to the channels having the best performance are chosen in the decryption process of the integer RLNC scheme (first approach). Similarly, users utilizing the second RLNC approach (binary RLNC) can transmit data on a subset ($Rt - ht$) of the available Rt rows, which corresponds to the channels

having the best performance. The remaining ht rows would carry redundant data to overcome any link error or failure. As a result, the probability of recovering the original message with fewer errors, increases.

Chapter 12

Conclusions and Future Work

This PhD dissertation mainly focuses on:

Designing and evaluating efficient PLS solutions for emerging communication systems in 5G networks.

Over the past few years, communication security and research related to this field have witnessed the proliferation of PLS; a novel security method for emerging communication systems in 5G networks such as IoT, D2D, M2M and many others. Extensive research and experimentation have shown that PLS is superior over conventional security in terms of robustness and efficiency. One main advantage of PLS is that using random and dynamic physical layer parameters reduces the computational complexity and execution time of traditional security schemes. Another benefit is that the physical layer is common to all heterogeneous devices, hence, security schemes at this layer are considered generic. Consequently, PLS has been explored in detail and various PLS schemes have been presented in the literature.

In this dissertation, the schemes presented in the literature have been summarized and compared to each other. As a result, several limitations and challenges have been analyzed and highlighted on. It has been shown that most techniques depend on the randomness of the channel only, which is not optimal in terms of system robustness and security. For this purpose, several PLS schemes that overcome the drawbacks of existing PLS methods have been designed and evaluated. Each of these schemes targets a specific security service: device authentication, key generation and distribution, data confidentiality, source authentication and message integrity, and data availability. All of the proposed schemes, combined, represent in a complete security framework for emerging systems.

This dissertation has valuable contribution in the PLS field, since it introduces novel solutions that are secure, efficient and deploy-able. It also paves the way for further development and optimization of PLS solutions in 5G 3GPP standards.

Currently, the security of PowerLine Communication (PLC) in Smart Grids is

being explored, in collaboration with Iberdrola Innovation Middle East (Qatar). In particular, it has been shown that one can benefit from the randomness and dynamicity of PLC channels to design PLS schemes that ensure data confidentiality, authentication and availability, at the physical and data link layers. This research work is still in progress. The contribution in this area will have a huge impact on Smart Grids since the proposed solutions will have a direct application in real life scenarios.

On-going and future work will focus on multiple aspects that compliment the work presented in this dissertation, and which are:

- Implementing and assessing the proposed solutions in a hardware environment.
- Proposing new PLS solutions based on machine learning and analyzing them in terms of performance and robustness.
- Designing an efficient intrusion detection system at the network and physical layers.
- Channel modelling; that is studying and comparing the common channel features/parameters that can be extracted by communicating entities from the shared wireless channel. Identifying the most secure and efficient features to be used in PLS.

Appendix A

Abbreviations

3GPP	Third Generation Partnership Project
PLS	Physical Layer Security
OFDM	Orthogonal Frequency Division Multiplexing
AES	Advanced Encryption Standard
HMAC	Hash-based Message Authentication Code
CMAC	Cipher-based Message Authentication Code
GMAC	Galois Message Authentication Code
CPC	Channel Pre-coding
BER	Bit Error Rate
MD	Message Digest
MIM	Man-In-the-Middle
MIMO	Multiple-Input Multiple-Output
NOMA	Non-Orthogonal Multiple Access
PD-NOMA	Power-Domain Non-Orthogonal Multiple Access
WSN	Wireless Sensor Network
CD-NOMA	Code-Domain Non-Orthogonal Multiple Access
MUSA	Multi-User Shared Access
SCMA	Sparse Code Multiple Access
LDS	Low-Density Spreading
PDMA	Pattern Division Multiple Access
BDM	Bit Division Multiplexing
DOS	Denial-of-Service
DDOS	Distributed Denial-of-Service
CSI	Channel State Information
SNR	Signal-to-Noise Ratio
SINR	Signal-to-Interference-plus-Noise Ratio
HTTPS	Hypertext Transfer Protocol-Secure
TLS	Transport Layer Security
IPsec	Internet Protocol Security
IP	Internet Protocol

M2M	Machine-to-Machine
D2D	Device-to-Device
IoT	Internet-of-Things
WLAN	Wireless Local Area Network
WiMAX	Worldwide Interoperability for Microwave Access
LTE	Long-Term Evolution
VLC	Visible Light Communication
BAN	Body Area Network
PLC	Power Line Communication
RFID	Radio Frequency Identification
RF	Radio Frequency
VANET	Vehicular Ad-Hoc Network
UWB	Ultra-Wide-Band
UAV	Unmanned Aerial Vehicle
BW	Bandwidth
CP	Cyclic Prefix
ISI	Inter-Symbol Interference
ICI	Inter-Carrier Interference
IFFT	Inverse Fast Fourier Transform
FFT	Fast Fourier Transform
STS	Short Training Sequence
LTS	Long Training Sequence
AGC	Automatic Gain Control
PAPR	Peak-to-Average Power Ratio
OOB	Out-of-Band
MCM	Multi-Carrier Modulation
FBMC	Filter Bank Multi-Carrier
UF-OFDM	Universal Filter OFDM
GFDM	Generalized Frequency Division Multiplexing
QAM	Quadrature Amplitude Modulation
OQAM	Offset Quadrature Amplitude Modulation
TDMA	Time-Division Multiple Access
CDMA	Code Division Multiple Access
Wi-Fi	Wireless Fidelity
QPSK	Quadrature Phase Shift Keying
BS	Base Station
UE	User Equipment
FU	Far User
NU	Near User
SIC	Successive Interference Cancellation
AWGN	Additive White Gaussian Noise
CFO	Carrier Frequency Offset
XOR	Exclusive OR

SHA	Secure Hash Algorithm
MAC	Message Authentication Code
DSSS	Direct Sequence Spread Spectrum
FHSS	Frequency Hopping Spread Spectrum
PUF	Physical Unclonable Function
FPGA	Field-Programmable Gate Array
NIST	National Institute of Standards and Technology
KGR	Key Generation Rate
ROR	Real-Or-Random
KDR	Key Disagreement Rate
RSS	Received Signal Strength
SSL	Secure Sockets Layer
AoA	Angle of Arrival
MSE	Mean Square Error
MISO	Multiple-Input Single-Output
SISO	Single-Input Single-Output
AN	Artificial Noise
AFF	Artificial Fast Fading
SVD	Singular Value Decomposition
QOS	Quality-of-Service
LoRaWAN	Log Range Wide Area Network
CDF	Cumulative Distribution Function

Appendix B

Notations and Symbols

DS	Delay spread
DS_{avg}	average delay spread
π	Mathematical constant approximately equal to 3.14159
$x_D(t)$	Superimposed downlink NOMA signal
$x_{D,k}(t)$	Individual downlink NOMA signal of the k^{th} user
$x_{U,k}(t)$	Individual uplink NOMA signal of the k^{th} user
$x'_{D,k}(t)$	Received downlink NOMA signal at the k^{th} user
K	Number of users in the network
α_k	Power coefficient of the k^{th} user
P_{BS}	Transmission power at the base station
$y_U(t)$	Superimposed uplink NOMA signal
h_k	Channel coefficient of the k^{th} user
H	Channel matrix
$no_k(t)$	Addition white Gaussian noise at the k^{th} user
λ	Wavelength
U	Orthogonal matrix resulting from the SVD of H
V	Orthogonal matrix resulting from the SVD of H
Λ	Diagonal matrix resulting from the SVD of H
y	Receive vector at BS
x	Transmit vector from UE
c	Ciphered/encrypted signal
X_{ID}	Unique user-identification
X_{PUF}	PUF output based on channel
SL	Link selection table
SG	Matrix selection table
$Gen(\cdot)$	A probabilistic fuzzy function (generation)
ΔT	Transmission delay
M	Message
Rt	Number of RATs
G	Invertible matrix (RLNC)

$Rep(\cdot)$	A probabilistic fuzzy function (recovery)
σ	Output of the $Gen(\cdot)$ function
τ	Output of the $Gen(\cdot)$ function
N_0	Channel-based nonce
SK	Secret session key
DK	Dynamic key
M_O	Modulation order
m_b	Number of bits in one constellation point
DSK	Dynamic sub-key
CS	Complex symbol
F	Frame
FS	Frame symbol
E_b/N_0	Signal-to-noise power ratio
EFS	Encrypted frame symbol
NB_F	Size of one frame
NB_M	Number of bytes in a message
$\wedge$	AND operation
NB_{FS}	Size of one frame symbol
qf	Galois field size
$Pbox$	Permutation table
$Sbox$	Substitution table
ID_S	Secret session identifier
PRS	Pseudo-random sequence
BKS	Binary key-stream
R	Random number
TS	Time stamp
j^2	-1
I	Identity matrix
$\odot$	Matrix multiplication
$(.)^T$	Transpose
$(.)^H$	Hermitian
$(.)^{-1}$	Inverse operation
$h(.)$	Hash function
$\|\|$	Concatenation operation
$\oplus$	Exclusive-OR (XOR)
nt	Number of antennas at the transmitter
nr	Number of antennas at the receiver
NB_{BL}	Number of blocks in one frame
Φ	Number of bytes in one frame
ϕ	Number of bytes in one block

Bibliography

[1] V. Poor and et. al, "Wireless physical layer security," *Proc. Nat. Acad. Sci. U. S. A.*, vol. 114, no. 1, pp. 19–26, 2017.

[2] A. Goldsmith, *Wireless communications*. Cambridge university press, 2005.

[3] T. Akitaya and T. Saba, "Energy efficient artificial fast fading for MISO-OFDM systems," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, pp. 1–6, Dec 2015.

[4] J. Zhang and et. al, "Design of an OFDM physical layer encryption scheme," *IEEE Trans. Veh. Technol.*, vol. 66, no. 3, pp. 2114–2127, 2017.

[5] R. Fielding and et. al, "Hypertext transfer protocol HTTP/1.1," *RFC 2616, june*, 1999.

[6] T. Dierks and E. Rescorla, "The transport layer security (TLS) protocol version 1.2," *RFC 5246, August*, 2008.

[7] B. Forouzan, *TCP/IP protocol suite*. McGraw-Hill, Inc., 2002.

[8] J. Hamamreh and H. Arslan, "Secure orthogonal transform division multiplexing (OTDM) waveform for 5g and beyond," *IEEE Commun. Lett.*, vol. 21, pp. 1191–1194, May 2017.

[9] J. Harshan and et. al, "Insider-attacks on physical-layer group secret-key generation in wireless networks," in *IEEE Proc. Wireless Commun. Netw. Conf. (WCNC)*, pp. 1–6, IEEE, 2017.

[10] J. Zhang and et. al, "Verification of key generation from individual OFDM subcarrier's channel response," in *Proc. IEEE Global Commun. Conf. Workshops (GC Wkshps)*, pp. 1–6, Dec 2015.

[11] F. Huo and G. Gong, "A new efficient physical layer OFDM encryption scheme," in *Proc. IEEE Int. Conf. Computer Commun. (INFOCOM)*, pp. 1024–1032, IEEE, 2014.

[12] T. Akitaya and et. al, "Time-domain artificial noise generation technique using time-domain and frequency-domain processing for physical layer security in MIMO-OFDM systems," in *Proc. IEEE Int. Conf. Commun. (ICC)*, pp. 807–812, June 2014.

[13] X. Wu and et. al, "Artificial-noise-aided physical layer phase challenge-response authentication for practical OFDM transmission," *IEEE Trans. Wireless Commun.*, vol. 15, pp. 6611–6625, Oct 2016.

[14] K. Umebayashi *et al.*, "A study on secure pilot signal design for OFDM systems," in *Asia-Pacific Signal and Inform. Process. Association Annual Summit and Conf. (APSIPA)*, pp. 1–5, Dec 2014.

[15] J. Massey, "Cryptography - a selective survey," *Digit. Commun.*, vol. 85, pp. 3–25, 1986.

[16] M. Bloch and et. al, "Wireless information-theoretic security," *IEEE Trans. Inf. Theory*, vol. 54, no. 6, pp. 2515–2534, 2008.

[17] C. Shannon, "Communication theory of secrecy systems," *Bell Labs Technical Journal*, vol. 28, no. 4, pp. 656–715, 1949.

[18] A. Wyner, "The wire-tap channel," *Bell Labs Technical Journal*, vol. 54, no. 8, pp. 1355–1387, 1975.

[19] A. Mukherjee and et. al, "Principles of physical layer security in multiuser wireless networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 3, pp. 1550–1573, 2014.

[20] M. Bloch and J. Barros, "Physical-layer security cambridge univ," 2011.

[21] R. Ahlswede and et. al, "Common randomness in information theory and cryptography. I. Secret sharing," *IEEE Trans. Inf. Theory*, vol. 39, no. 4, pp. 1121–1132, 1993.

[22] I. Csiszar and et. al, "Broadcast channels with confidential messages," *IEEE Trans. Inf. Theory*, vol. 24, no. 3, pp. 339–348, 1978.

[23] A. Khisti and G. Wornell, "Secure transmission with multiple antennas i: The MISOME wiretap channel," *IEEE Trans. Inf. Theory*, vol. 56, no. 7, pp. 3088–3104, 2010.

[24] J. Liu and et. al, "Optimal power allocation for achieving perfect secrecy capacity in mimo wire-tap channels," in *IEEE proc. Inf. Sci. and Syst.*, pp. 606–611, IEEE, 2009.

[25] Y. Cao and et. al, "A survey of emerging m2m systems: Context, task, and objective," *IEEE Internet Things J.*, vol. 3, no. 6, pp. 1246–1258, 2016.

[26] J. M. Hamamreh, H. M. Furqan, and H. Arslan, "Classifications and applications of physical layer security techniques for confidentiality: A comprehensive survey," *IEEE Communications Surveys Tutorials*, vol. 21, pp. 1773–1828, Secondquarter 2019.

[27] R. Melki, H. Noura, M. Mansour, and A. Chehab, "A survey on OFDM physical layer security," *Physical Communication*, vol. 32, pp. 1–30, 2019.

[28] D. Swift, "A practical application of SIM/SEM/SIEM automating threat identification," *Paper, SANS Infosec Reading Room, The SANS*, 2006.

[29] M. Dworkin, "Recommendation for block cipher modes of operation: methods and techniques," tech. rep., DTIC Document, 2001.

[30] C. Shannon, "Communication Theory of Secrecy Systems," *Bell Systems Technical Journal*, vol. 28, pp. 656–715, 1949.

[31] C. Paar and J. Pelzl, *Understanding cryptography: a textbook for students and practitioners*. Springer Science & Business Media, 2009.

[32] T. Kwon and et. al, "Design and implementation of a simulator based on a cross-layer protocol between MAC and PHY layers in a WiBro compatible. IEEE 802.16 e OFDMA system," *IEEE Communications Magazine*, vol. 43, no. 12, pp. 136–146, 2005.

[33] B. Saltzberg, "Performance of an efficient parallel data transmission system," *IEEE Trans. on Commun. Technol.*, vol. 15, no. 6, pp. 805–811, 1967.

[34] R. Chang, "Synthesis of band-limited orthogonal signals for multichannel data transmission," *Bell Labs Technical Journal*, vol. 45, no. 10, pp. 1775–1796, 1966.

[35] J. Bingham, "Multicarrier modulation for data transmission: An idea whose time has come," *IEEE Commun. Mag.*, vol. 28, no. 5, pp. 5–14, 1990.

[36] M. Schwartz, *Mobile wireless communications*. Cambridge University Press, 2004.

[37] R. Prasad, *OFDM for wireless communications systems*. Artech House, 2004.

[38] "Concepts of orthogonal frequency division multiplexing (OFDM) and 802.11 wlan." http://rfmw.em.keysight.com/wireless/helpfiles/89600b/webhelp/subsystems/wlan-ofdm/content/ofdm_basicprinciplesoverview.htm. (Accessed on 01/04/2018).

[39] "802.11 OFDM WLAN overview." http://rfmw.em.keysight.com/wireless/helpfiles/89600b/webhelp/subsystems/wlan-ofdm/content/ofdm_80211-overview.htm. (Accessed on 01/04/2018).

[40] I. C. S. L. S. Committee *et al.*, "Wireless LAN medium access control (MAC) and physical layer (PHY) specifications," *IEEE Standard 802.11-1997*, 1997.

[41] H. Rahbari and M. Krunz, "Exploiting frame preamble waveforms to support new physical-layer functions in OFDM-based 802.11 systems," *IEEE Trans. Wireless Commun.*, vol. 16, pp. 3775–3786, June 2017.

[42] R. Franzin and et. al, "A performance comparison between OFDM and FBMC in PLC applications," in *IEEE International Conference on Ecuador Technical Chapters Meeting (ETCM)*, IEEE, 2017.

[43] V. Moles-Cases and et. al, "A comparison of OFDM, QAM-FBMC, and OQAM-FBMC waveforms subject to phase noise," in *Proc. IEEE Int. Conf. Commun. (ICC)*, pp. 1–6, IEEE, 2017.

[44] Q. He and A. Schmeink, "Comparison and evaluation between FBMC and OFDM systems," in *Proc. Int. ITG Workshop on Smart Antennas (WSA)*, pp. 1–7, VDE, 2015.

[45] A. Viholainen *et al.*, "Prototype filter design for filter bank based multicarrier transmission," in *Proc. IEEE European Signal Process. Conference*, pp. 1359–1363, IEEE, 2009.

[46] Q. He and A. Schmeink, "Comparison and evaluation between FBMC and OFDM systems," in *Proc. workshop on smart antennas (WSA)*, pp. 1–7, VDE, 2015.

[47] R. Franzin and P. Lopes, "A performance comparison between OFDM and FBMC in PLC applications," in *Proc. IEEE Ecuador Technical Chapters Meeting (ETCM)*, IEEE, 2017.

[48] A. Roessler, "5G waveform candidates application note," *Rohde&Schwarz, Munich, Germany, Tech. Rep. 1MA271*, 2016.

[49] M. Aldababsa *et al.*, "A tutorial on nonorthogonal multiple access for 5G and beyond," *Wireless Communications and Mobile Computing*, 2018.

[50] H. Furqan *et al.*, "Physical layer security for NOMA: Requirements, merits, challenges, and recommendations," *arXiv preprint arXiv:1905.05064*, 2019.

[51] R. Kizilirmak and H. Bizaki, "Non-orthogonal multiple access (NOMA) for 5G networks," *Towards 5G Wireless Networks-A Physical Layer Perspective*, pp. 83–98, 2016.

[52] M. Taherzadeh, H. Nikopour, A. Bayesteh, and H. Baligh, "Scma codebook design," in *Proc. IEEE Vehicular Technology Conference (VTC2014-Fall)*, pp. 1–5, IEEE, 2014.

[53] "MIMO antenna beamforming: Radio-electronics.com." `http://www.radio-electronics.com/info/antennas/mimo/antenna-beamforming.php`. (Accessed on 01/04/2018).

[54] W. Shehab and Z. Al-qudah, "Singular value decomposition: Principles and applications in multiple input multiple output communication system," *Intl. J. Comput. Netwo. Commun*, vol. 9, no. 1, pp. 13–21, 2017.

[55] X. Wu and et. al, "Physical-layer authentication for multi-carrier transmission," *IEEE Commun. Lett.*, vol. 19, no. 1, pp. 74–77, 2015.

[56] F. Liu and et. al, "A two dimensional quantization algorithm for CIR-based physical layer authentication," in *Proc. IEEE Int. Conf. Commun. (ICC)*, pp. 4724–4728, June 2013.

[57] M. Pospl and R. Mark, "Experimental study of wireless transceiver authentication using carrier frequency offset monitoring," in *International Conference Radioelektronika (RADIOELEKTRONIKA)*, pp. 335–338, April 2015.

[58] M. Liu and et. al, "TBAS: Enhancing wi-fi authentication by actively eliciting channel state information," in *IEEE Int. Conf. Sensing, Commun. and Netw. (SECON)*, pp. 1–9, June 2016.

[59] C. Dai and et. al, "Physical layer authentication algorithm based on SVM," in *IEEE proc. Int. Conf. Commun. China (ICCC)*, pp. 1597–1601, IEEE, 2016.

[60] H. Wen and et. al, "A novel framework for message authentication in vehicular communication networks," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, pp. 1–6, IEEE, 2009.

[61] G. Caparra and et. al, "Energy-based anchor node selection for IoT physical layer authentication," in *Proc. IEEE Int. Conf. Commun. (ICC)*, pp. 1–6, IEEE, 2016.

[62] J. Zhang and et. al, "Using basis expansion model for physical layer authentication in time-variant system," in *IEEE Proc. Commun. Netw. Security (CNS)*, pp. 348–349, IEEE, 2016.

[63] A. Mahmood and et. al, "Channel impulse response-based distributed physical layer authentication," *arXiv preprint arXiv:1703.08559*, 2017.

[64] W. Wang and et. al, "Privacy-preserving location authentication in Wi-Fi networks using fine-grained physical layer signatures," *IEEE Trans. Wireless Commun*, vol. 15, pp. 1218–1225, Feb 2016.

[65] X. Du and et. al, "Physical layer challenge-response authentication in wireless networks with relay," in *Proc. IEEE Int. Conf. Computer Commun. (INFOCOM)*, pp. 1276–1284, April 2014.

[66] G. Verma and et. al, "Physical layer authentication via fingerprint embedding using software-defined radios," *IEEE Access*, vol. 3, pp. 81–88, 2015.

[67] X. Wu and et. al, "A channel coding approach for physical-layer authentication," in *IEEE Proc. Wireless Commun. Sig. Process. (WCSP)*, pp. 1–5, IEEE, 2016.

[68] A. Amanna and et. al, "Realizing physical layer authentication using constellation perturbation on a software-defined radio testbed," in *IEEE Proc. Military Commun. Conf. (MILCOM)*, pp. 1207–1212, IEEE, 2016.

[69] X. Fang and et. al, "Towards PHY-aided authentication via weighted fractional fourier transform," in *IEEE proc. Veh. Technol. Conf. (VTC-Fall)*, pp. 1–5, IEEE, 2016.

[70] J. Yang and et. al, "A physical-layer authentication scheme based on hash method," in *IEEE proc. Int. Conf. Commun. China (ICCC)*, pp. 99–104, IEEE, 2015.

[71] D. Xu, P. Ren, J. Ritcey, and Y. Wang, "Code-frequency block group coding for anti-spoofing pilot authentication in multi-antenna OFDM systems," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 7, pp. 1778–1793, 2018.

[72] K. Das, M. Wazid, N. Kumar, M. Khan, K. Choo, and Y. Park, "Design of secure and lightweight authentication protocol for wearable devices environment," *IEEE journal of biomedical and health informatics*, vol. 22, no. 4, pp. 1310–1322, 2018.

[73] X. Li *et al.*, "A three-factor anonymous authentication scheme for wireless sensor networks in internet of things environments," *Journal of Network and Computer Applications*, vol. 103, pp. 194–204, 2018.

[74] A. Das *et al.*, "Provably secure user authentication and key agreement scheme for wireless sensor networks," *Security and Communication Networks*, vol. 9, no. 16, pp. 3670–3687, 2016.

[75] C. Chang and H. Le, "A provably secure, efficient, and flexible authentication scheme for ad hoc wireless sensor networks," *IEEE Transactions on Wireless Communications*, vol. 15, no. 1, pp. 357–366, 2016.

[76] J. Zhang and et. al, "Key generation from wireless channels: A review," *IEEE Access*, vol. 4, pp. 614–626, 2016.

[77] Z. Rezki and et. al, "Secret key agreement: Fundamental limits and practical challenges," *IEEE Wireless Commun.*, 2017.

[78] J. Zhang and et. al, "Efficient key generation by exploiting randomness from channel responses of individual OFDM subcarriers," *IEEE Trans. Commun.*, vol. 64, pp. 2578–2588, June 2016.

[79] C. Sahin and et. al, "Secure and robust symmetric key generation using physical layer techniques under various wireless environments," in *IEEE Radio and Wireless Symposium (RWS)*, pp. 211–214, IEEE, 2016.

[80] Y. Peng and et. al, "Secret key generation based on estimated channel state information for TDD-OFDM systems over fading channels," *IEEE Trans. Wireless Commun.*, vol. 16, pp. 5176–5186, Aug 2017.

[81] A. Mazin and et. al, "Secure key management for 5G physical layer security," in *IEEE proc. Wireless Microw. Technol. Conf. (WAMICON)*, pp. 1–5, IEEE, 2017.

[82] Y. Al-Moliki and et. al, "Robust key generation from optical OFDM signal in indoor VLC networks," *IEEE Photon. Technol. Lett.*, vol. 28, pp. 2629–2632, Nov 2016.

[83] R. Horstmeyer and et. al, "Physical key-protected one time pad," June 9 2015. US Patent 9,054,871.

[84] J. Guajardo and et. al, "Fpga intrinsic pufs and their use for IP protection," in *CHES*, vol. 4727, pp. 63–80, Springer, 2007.

[85] L. Bolotnyy and G. Robins, "Physically unclonable function-based security and privacy in RFID systems," in *IEEE Proc. Int. Conf. Pervasive Comput. Commun. (PerCom'07)*, pp. 211–220, IEEE, 2007.

[86] R. Pappu and et. al, "Physical one-way functions," *Science*, vol. 297, no. 5589, pp. 2026–2030, 2002.

[87] P. Tuyls and et. al, *Security with noisy data: on private biometrics, secure key storage and anti-counterfeiting*. Springer Science & Business Media, 2007.

[88] Y. Al-Moliki, M. Alresheedi, and Y. Al-Harthi, "Robust key generation from optical OFDM signal in indoor VLC networks," *IEEE Photonics Technology Letters*, vol. 28, no. 22, pp. 2629–2632, 2016.

[89] S. Gollakota and D. Katabi, "Physical layer wireless security made fast and channel independent," in *Proc. IEEE Int. Conf. Computer Commun. (INFOCOM)*, pp. 1125–1133, IEEE, 2011.

[90] J. Zhang and et. al, "Securing wireless communications of the internet of things from the physical layer, an overview," *Entropy*, vol. 19, no. 8, p. 420, 2017.

[91] H. Liu and et. al, "Fast and practical secret key extraction by exploiting channel response," in *Proc. IEEE Int. Conf. Computer Commun. (INFOCOM)*, pp. 3048–3056, April 2013.

[92] A. Badawy and et. al, "Channel secondary random process for robust secret key generation," in *IEEE proc. Int. Wireless Commun. and Mobile Comput. Conf. (IWCMC)*, pp. 114–119, Aug 2015.

[93] Y. Zhang and et. al, "An over-the-air key establishment protocol using keyless cryptography," *Future Generation Computer Systems*, 2016.

[94] A. Badawy and et. al, "Unleashing the secure potential of the wireless physical layer: Secret key generation methods," *Physical Communication*, vol. 19, pp. 1–10, 2016.

[95] R. Guillaume and et. al, "Fair comparison and evaluation of quantization schemes for PHY-based key generation," in *Proc. Int. OFDM Workshop (InOWo'14)*, pp. 1–5, Aug 2014.

[96] A. Saad and et. al, "Comparative simulation for physical layer key generation methods," in *IEEE proc. Int. Wireless Commun. and Mobile Comput. Conf. (IWCMC)*, pp. 120–125, IEEE, 2015.

[97] O. Gunlu and et. al, "Reliable secret-key binding for physical unclonable functions with transform coding," in *IEEE Proc. Global Cpnf. Signal Inf. Process. (GlobalSIP)*, pp. 986–991, IEEE, 2016.

[98] Z. Mahmood and et. al, "Lightweight two-level session key management for end user authentication in internet of things," in *Internet of Things (iThings) and IEEE Green Computing and Communications*

(GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), 2016 IEEE International Conference on, pp. 323–327, IEEE, 2016.

[99] W. Xi and et. al, "KEEP: Fast secret key extraction protocol for D2D communication," in *IEEE Int. Symp. of Quality of Service (IWQoS)*, pp. 350–359, May 2014.

[100] J. Li and et. al, "Analysis of non-reciprocity factors in extracting secret key from wireless channels for practical indoor scenarios," in *IEEE proc. Int. Conf. Commun. China (ICCC)*, pp. 227–231, Oct 2016.

[101] "WARP project." `http://warpproject.org/trac`. (Accessed on 01/04/2018).

[102] P. Luo and et. al, "Threat on physical layer security: Side channel vs. wiretap channel," in *IEEE Proc. Int. Conf. Comput. Science Eng. (CSE)*, pp. 295–300, IEEE, 2013.

[103] Y. Tsai and et. al, "Effective channel perturbation based on cyclic delay for physical layer security in OFDM systems," in *IEEE proc. Int. Conf. Inf. Sci. Electron. Electr. Eng.*, vol. 2, pp. 823–827, April 2014.

[104] A. Hajomer and et. al, "Secure OFDM transmission precoded by chaotic discrete hartley transform," *IEEE Photon. J*, vol. PP, no. 99, pp. 1–1, 2017.

[105] W. Zhang and et. al, "Hybrid time-frequency domain chaotic interleaving for physical-layer security enhancement in OFDM-PON systems," in *IEEE proc. Int. Conf. Commun. China (ICCC)*, pp. 1–4, IEEE, 2016.

[106] L. Zhang and et. al, "Theory and performance analyses in secure CO-OFDM transmission system based on two-dimensional permutation," *J. Lightw. Technol*, vol. 31, pp. 74–80, Jan 2013.

[107] H. Li and et. al, "Secure transmission in OFDM systems by using time domain scrambling," in *IEEE proc. Veh. Technol. Conf. (VTC Spring)*, pp. 1–5, June 2013.

[108] J. M. Hamamreh and et. al, "Secure pre-coding and post-coding for ofdm systems along with hardware implementation," in *IEEE proc. Int. Wireless Commun. and Mobile Comput. Conf. (IWCMC)*, pp. 1338–1343, June 2017.

[109] H. Li and et. al, "Dynamic subcarrier coordinate interleaving for eavesdropping prevention in OFDM systems," *IEEE Commun. Lett.*, vol. 18, pp. 1059–1062, June 2014.

[110] F. Huo and G. Gong, "A new efficient physical layer OFDM encryption scheme," in *Proc. IEEE Int. Conf. Computer Commun. (INFOCOM)*, (Toronto, ON, Canada), pp. 1024–1032, Apr. 2014.

[111] Y. Lee and et. al, "Secure index and data symbol modulation for OFDM-IM," *IEEE Access*, vol. PP, no. 99, pp. 1–1, 2017.

[112] J. Hamamreh and et. al, "OFDM-subcarrier index selection for enhancing security and reliability of 5G URLLC services," *IEEE Access*, vol. PP, no. 99, pp. 1–1, 2017.

[113] K. Naito and et. al, "Channel state based secure wireless communication," in *IEEE Proc. Int. Conf. Computer Commun. Workshops (INFOCOM WKSHPS)*, pp. 828–834, April 2016.

[114] T. Wang and et. al, "Security-coded OFDM system based on multiorder fractional fourier transform," *IEEE Commun. Lett.*, vol. 20, pp. 2474–2477, Dec 2016.

[115] M. Sakai and et. al, "Intrinsic interference based physical layer encryption for OFDM/OQAM," *IEEE Commun. Lett.*, vol. 21, no. 5, pp. 1059–1062, 2017.

[116] J. Xiong and Z. Wang, "Physical layer security OFDM communication using phased array antenna," in *IEEE proc. Int. Conf. Commun. China (ICCC)*, pp. 1–4, July 2016.

[117] F. Huo and G. Gong, "XOR encryption versus phase encryption, an in-depth analysis," *IEEE Trans. Electromagn. Compat.*, vol. 57, pp. 903–911, Aug 2015.

[118] W. Zhang and et. al, "Joint PAPR reduction and physical layer security enhancement in OFDMA-PON," *IEEE Photon. Technol. Lett.*, vol. 28, pp. 998–1001, May 2016.

[119] L. Deng and et. al, "Secure OFDM-PON system based on chaos and fractional fourier transform techniques," *J. Lightw. Technol.*, vol. 32, pp. 2629–2635, Aug 2014.

[120] Y. Al-Moliki, M. Alresheedi, and Y. Al-Harthi, "Physical-layer security against known/chosen plaintext attacks for OFDM-based VLC system," *IEEE Communications Letters*, vol. 21, no. 12, pp. 2606–2609, 2017.

[121] H. Furqan, J. Hamamreh, and H. Arslan, "Enhancing physical layer security of OFDM systems using channel shortening," in *IEEE Proc. International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, pp. 1–5, IEEE, 2017.

[122] D. Cheng and et. al, "A general time-domain artificial noise design for OFDM AF relay systems," in *IEEE proc. Int. Conf. Commun. China (ICCC)*, pp. 1–6, Nov 2015.

[123] M. Yusuf and H. Arslan, "Controlled inter-carrier interference for physical layer security in OFDM systems," in *IEEE proc. Veh. Technol. Conf. (VTC-Fall)*, pp. 1–5, Sept 2016.

[124] M. Soltani and et. al, "Achieving secure communication through pilot manipulation," in *IEEE Proc. Annual Int. Symp. on Personal, Indoor, and Mobile Radio Commun. (PIMRC)*, pp. 527–531, Aug 2015.

[125] J. Lin and et. al, "Frequency diverse array beamforming for physical-layer security with directionally-aligned legitimate user and eavesdropper," in *European Signal Processing Conference (EUSIPCO)*, pp. 2166–2170, Aug 2017.

[126] J. Lin and et. al, "Physical-layer security for proximal legitimate user and eavesdropper: A frequency diverse array beamforming approach," *IEEE Trans. Inf. Forensics Security*, vol. PP, no. 99, pp. 1–1, 2017.

[127] L. Chen and et. al, "Fast power allocation for secure communication with full-duplex radio," *IEEE Trans. Signal Process.*, vol. 65, pp. 3846–3861, July 2017.

[128] M. Zhang and Y. Liu, "Energy harvesting for physical-layer security in OFDMA networks," *IEEE Trans. Inf. Forensics Security*, vol. 11, pp. 154–162, Jan 2016.

[129] L. Deng *et al.*, "Joint power and subcarrier allocation using auction games for secure multiuser OFDMA networks," in *IEEE proc. Int. Conf. Commun. China (ICCC)*, pp. 153–158, Nov 2015.

[130] H. Qin *et al.*, "Power allocation and time-domain artificial noise design for wiretap OFDM with discrete inputs," *IEEE Transactions on Wireless Communications*, vol. 12, no. 6, pp. 2717–2729, 2013.

[131] S. Sheikhzadeh and et. al, "Radio resource allocation for physical-layer security in OFDMA based hetnets with unknown mode of adversary," in *Iran Workshop on Communication and Information Theory (IWCIT)*, pp. 1–6, May 2017.

[132] S. Karachontzitis and et. al, "Security-aware max-min resource allocation in multiuser OFDMA downlink," *IEEETrans. Inf. Forensics Security*, vol. 10, pp. 529–542, March 2015.

[133] R. Saini and et. al, "Jammer-assisted resource allocation in secure OFDMA with untrusted users," *IEEETrans. Inf. Forensics Security*, vol. 11, pp. 1055–1070, May 2016.

[134] N. Mokari and et. al, "Secure robust ergodic uplink resource allocation in relay-assisted cognitive radio networks," *IEEE Trans. Signal Process.*, vol. 63, pp. 291–304, Jan 2015.

[135] G. Zhang and et. al, "Wireless powered cooperative jamming for secure ofdm system," *IEEE Trans. Veh. Technol*, vol. PP, no. 99, pp. 1–1, 2017.

[136] B. Zhang and et. al, "Priwhisper: Enabling keyless secure acoustic communication for smartphones," *IEEE Internet Things J.*, vol. 1, no. 1, pp. 33–45, 2014.

[137] Y. Xiao and et. al, "PAPR reduction based on chaos combined with SLM technique in optical OFDM IM/DD system," *Opt. Fiber Technol.*, vol. 21, pp. 81–86, 2015.

[138] P. Ponniah, *Database design and development: an essential guide for IT professionals*. Wiley Online Library, 2003.

[139] T. Krovetz *et al.*, "UMAC: Message authentication code using universal hashing," *The Internet Society, RFC4418*, 2006.

[140] Y. Chen *et al.*, "A publicly verifiable network coding scheme with null-space HMAC," *International Journal of Intelligent Information and Database Systems*, vol. 11, no. 2-3, pp. 117–131, 2018.

[141] B. Echandouri, F. Omary, F. Ziani, and A. Sadak, "SEC-CMAC a new message authentication code based on the symmetrical evolutionist ciphering algorithm," *International Journal of Information Security and Privacy (IJISP)*, vol. 12, no. 3, pp. 16–26, 2018.

[142] B. Sung, K. Kim, and K. Shin, "An AES-GCM authenticated encryption crypto-core for iot security," in *International Conference on Electronics, Information, and Communication (ICEIC)*, pp. 1–3, IEEE, 2018.

[143] N. Kumar and P. Chaudhary, "Password security using bcrypt with AES encryption algorithm," in *Smart Computing and Informatics*, pp. 385–392, Springer, 2018.

[144] D. Nairn *et al.*, "Authenticating messages sent over a vehicle bus that include message authentication codes," 2019. US Patent App. 10/211,990.

[145] I. Gribanova and A. Semenov, "Using automatic generation of relaxation constraints to improve the preimage attack on 39-step MD4," in *International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, pp. 1174–1179, IEEE, 2018.

[146] Y. Tian, K. Zhang, P. Wang, Y. Zhang, and J. Yang, "Add "salt" MD5 algorithm's FPGA implementation," *Procedia computer science*, vol. 131, pp. 255–260, 2018.

[147] A. Visconti and F. Gorla, "Exploiting an HMAC-SHA-1 optimization to speed up PBKDF2," *IEEE Transactions on Dependable and Secure Computing*, 2018.

[148] K. Yap *et al.*, "Method and apparatus to process SHA-2 secure hashing algorithm," Dec. 4 2018. US Patent App. 10/146,544.

[149] P. Luo, K. Athanasiou, Y. Fei, and T. Wahl, "Algebraic fault analysis of SHA-3 under relaxed fault models," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 7, pp. 1752–1761, 2018.

[150] D. Chen, N. Zhang, R. Lu, N. Cheng, K. Zhang, and Z. Qin, "Channel precoding based message authentication in wireless networks: Challenges and solutions," *IEEE Network*, vol. 33, pp. 99–105, January 2019.

[151] D. Chen, N. Zhang, N. Cheng, K. Zhang, Z. Qin, and X. S. Shen, "Physical layer based message authentication with secure channel codes," *IEEE Transactions on Dependable and Secure Computing*, pp. 1–1, 2018.

[152] H. Noura, *Conception et simulation des générateurs, crypto-systèmes et fonctions de hachage basés chaos performants*. PhD thesis, université de Nantes, 2012.

[153] Q. Yan and et. al, "Jamming resilient communication using MIMO interference cancellation," *IEEE Trans. Inf. Forensics Security*, vol. 11, pp. 1486–1499, July 2016.

[154] T. Do and et. al, "Jamming-resistant receivers for the massive MIMO uplink," *IEEE Trans. Inf. Forensics Security*, vol. PP, no. 99, pp. 1–1, 2017.

[155] W. Shen and et. al, "No time to demodulate-fast physical layer verification of friendly jamming," in *IEEE Proc. Military Commun. Conf. (MILCOM)*, pp. 653–658, IEEE, 2015.

[156] G. Satrya and Y. Shin, "Enhancing security of SIC algorithm on non-orthogonal multiple access (NOMA) based systems," *Physical Communication*, vol. 33, pp. 16–25, 2019.

[157] G. Satrya and Y. Shin, "Security enhancement to successive interference cancellation algorithm for non-orthogonal multiple access (NOMA)," in *Proc. IEEE Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, pp. 1–5, IEEE, 2017.

[158] M. Abolpour, M. Mirmohseni, and M. Aref, "Outage performance in secure cooperative NOMA," in *Proc. IEEE Iran Workshop on Communication and Information Theory (IWCIT)*, pp. 1–6, IEEE, 2019.

[159] Y. Feng, S. Yan, C. Liu, Z. Yang, and N. Yang, "Two-stage relay selection for enhancing physical layer security in non-orthogonal multiple access," *IEEE Transactions on Information Forensics and Security*, vol. 14, pp. 1670–1683, June 2019.

[160] Z. Xiang *et al.*, "Physical layer security in cognitive radio inspired NOMA network," *IEEE Journal of Selected Topics in Signal Processing*, vol. 13, no. 3, pp. 700–714, 2019.

[161] F. Zhou *et al.*, "Enhancing PHY security of MISO NOMA SWIPT systems with a practical non-linear EH model," in *Proc. IEEE International Conference on Communications Workshops (ICC Workshops)*, pp. 1–6, IEEE, 2018.

[162] N. Nandan, S. Majhi, and H. Wu, "Secure beamforming for MIMO-NOMA-based cognitive radio network," *IEEE Communications Letters*, vol. 22, no. 8, pp. 1708–1711, 2018.

[163] B. Zheng *et al.*, "Secure NOMA based two-way relay networks using artificial noise and full duplex," *IEEE Journal on Selected Areas in Communications*, vol. 36, pp. 1426–1440, July 2018.

[164] B. Zheng, F. Chen, M. Wen, Q. Li, Y. Liu, and F. Ji, "Secure NOMA based cooperative networks with rate-splitting source and full-duplex relay," in *Proc. IEEE International Symposium on Wireless Communication Systems (ISWCS)*, pp. 1–5, Aug 2018.

[165] M. Zeng, N. Nguyen, O. Dobre, and H. Poor, "Securing downlink massive MIMO-NOMA networks with artificial noise," *IEEE Journal of Selected Topics in Signal Processing*, vol. 13, pp. 685–699, June 2019.

[166] Y. Feng *et al.*, "Beamforming design and power allocation for secure transmission with NOMA," *IEEE Transactions on Wireless Communications*, vol. 18, pp. 2639–2651, May 2019.

[167] K. Xiao, L. Gong, and M. Kadoch, "Opportunistic multicast NOMA with security concerns in a 5G massive MIMO system," *IEEE Communications Magazine*, vol. 56, pp. 91–95, March 2018.

[168] Y. Alsaba, C. Leow, and S. A. Rahim, "Null-steering beamforming for enhancing the physical layer security of non-orthogonal multiple access system," *IEEE Access*, vol. 7, pp. 11397–11409, 2019.

[169] J. Chen, L. Yang, and M. Alouini, "Physical layer security for cooperative NOMA systems," *IEEE Transactions on Vehicular Technology*, vol. 67, pp. 4645–4649, May 2018.

[170] Y. Wu, A. Khisti, C. Xiao, G. Caire, K. Wong, and X. Gao, "A survey of physical layer security techniques for 5G wireless networks and challenges ahead," *IEEE Journal on Selected Areas in Communications*, vol. 36, pp. 679–695, April 2018.

[171] F. Jiang, G. Huang, W. Liu, and C. Sun, "Adaptive power allocation for D2D assisted cooperative relaying system with NOMA," in *Proc. IEEE/CIC International Conference on Communications in China (ICCC)*, pp. 676–681, Aug 2018.

[172] L. Lei *et al.*, "Power and load optimization in interference-coupled non-orthogonal multiple access networks," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, pp. 1–6, Dec 2018.

[173] L. You *et al.*, "Resource optimization with load coupling in multi-cell NOMA," *IEEE Transactions on Wireless Communications*, vol. 17, pp. 4735–4749, July 2018.

[174] B. Su, Q. Ni, and B. He, "Robust transmit designs for secrecy rate constrained MISO NOMA system," in *Proc. IEEE Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, pp. 1–5, Sep. 2018.

[175] H. Wang, X. Zhang, Q. Yang, and T. Tsiftsis, "Secure users oriented downlink MISO NOMA," *IEEE Journal of Selected Topics in Signal Processing*, vol. 13, pp. 671–684, June 2019.

[176] B. ElHalawany, R. Ruby, T. Riihonen, and K. Wu, "Performance of cooperative NOMA systems under passive eavesdropping," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, pp. 1–6, Dec 2018.

[177] A. Islam, M. Uddin, M. Kader, and S. Shin, "Blockchain based secure data handover scheme in non-orthogonal multiple access," in *Proc. IEEE International Conference on Wireless and Telematics (ICWT)*, pp. 1–5, July 2018.

[178] D. Xu, P. Ren, and H. Lin, "Combat hybrid eavesdropping in power-domain NOMA: Joint design of timing channel and symbol transformation," *IEEE Transactions on Vehicular Technology*, vol. 67, pp. 4998–5012, June 2018.

[179] B. Chen *et al.*, "Secure primary transmission assisted by a secondary full-duplex NOMA relay," *IEEE Transactions on Vehicular Technology*, vol. 68, pp. 7214–7219, July 2019.

[180] X. Shao and Z. Wu, "A novel multiple access scheme with physical layer security," in *Proc. IEEE International Conference on Communication Technology (ICCT)*, pp. 36–40, Oct 2018.

[181] K. Lai, J. Lei, L. Wen, G. Chen, W. Li, and P. Xiao, "Secure transmission with randomized constellation rotation for downlink sparse code multiple access system," *IEEE Access*, vol. 6, pp. 5049–5063, 2018.

[182] H. Noura, R. Melki, A. Chehab, and M. Mansour, "A physical encryption scheme for low-power wireless M2M devices: a dynamic key approach," *Mobile Networks and Applications*, pp. 1–17, 2018.

[183] R. Melki, H. Noura, M. Mansour, and A. Chehab, "An efficient OFDM-based encryption scheme using a dynamic key approach," *IEEE Internet of Things Journal*, 2018.

[184] F. Wang and et. al, "Poster abstract: Security in uplink MU-MIMO networks," in *IEEE/ACM Second International Conference on Internet-of-Things Design and Implementation (IoTDI)*, pp. 351–352, April 2017.

[185] S. Tomasin, "Comparison between asymmetric and symmetric channel-based authentication for MIMO systems," in *International ITG Workshop on Smart Antennas (WSA)*, pp. 1–5, March 2017.

[186] L. Xiao and et. al, "Game theoretic study on channel-based authentication in MIMO systems," *IEEE Trans. Veh. Technol.*, vol. 66, pp. 7474–7484, Aug 2017.

[187] L. Xiao and et. al, "Channel-based authentication game in MIMO systems," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, pp. 1–6, Dec 2016.

[188] O. Topal and et. al, "Space-frequency grouping based key extraction for MIMO-OFDM systems," in *International Symposium on Wireless Communication Systems (ISWCS)*, pp. 320–324, Aug 2017.

[189] K. Chen and B. Natarajan, "Evaluating node reliability in cooperative MIMO networks," *IEEE Trans. Inf. Forensics Security*, vol. 11, pp. 1453–1460, July 2016.

[190] L. Cheng and et. al, "Secret key generation via random beamforming in stationary environment," in *Proc. Int. Conf. Wireless Commun. Signal Process. (WCSP)*, pp. 1–5, Oct 2015.

[191] V. Yakovlev and et. al, "Secret key agreement based on a communication through wireless MIMO fading channels," in *Federated Conference on Computer Science and Information Systems (FedCSIS)*, pp. 823–830, Sept 2016.

[192] D. Qin and Z. Ding, "Exploiting multi-antenna non-reciprocal channels for shared secret key generation," *IEEE Trans. Inf. Forensics Security*, vol. 11, pp. 2693–2705, Dec 2016.

[193] K. Chen and et. al, "Secret key generation rate with power allocation in relay-based LTE-A networks," *IEEE Trans. Inf. Forensics Security*, vol. 10, pp. 2424–2434, Nov 2015.

[194] H. Taha and E. Alsusa, "Secret key establishment technique using channel state information driven phase randomisation in multiple-input multiple-output orthogonal frequency division multiplexing," *IET Information Security*, vol. 11, no. 1, pp. 1–7, 2017.

[195] J. Choi, "Secret key transmission for OFDM based machine type communications," *Journal of Communications and Networks*, vol. 19, pp. 363–370, August 2017.

[196] J. Choi and J. Ha, "Secret key transmission based on channel reciprocity for secure IoT," in *European Conference on Networks and Communications (EuCNC)*, pp. 388–392, June 2016.

[197] H. Furqan and et. al, "Secret key generation using channel quantization with SVD for reciprocal MIMO channels," in *International Symposium on Wireless Communication Systems (ISWCS)*, pp. 597–602, Sept 2016.

[198] H. Taha and E. Alsusa, "Secret key exchange under physical layer security using MIMO private random precoding in fdd systems," in *Proc. IEEE Int. Conf. Commun. (ICC)*, pp. 1–6, May 2016.

[199] H. Taha and E. Alsusa, "A MIMO precoding based physical layer security technique for key exchange encryption," in *IEEE Proc. Veh. Techn. Conf. (VTC Spring)*, pp. 1–5, May 2015.

[200] H. Taha and E. Alsusa, "Secret key exchange using private random precoding in MIMO FDD and TDD systems," *IEEE Trans. Veh. Techn.*, vol. 66, pp. 4823–4833, June 2017.

[201] E. Yaacoub, "On secret key generation with massive MIMO antennas using time-frequency-space dimensions," in *IEEE Middle East Conference on Antennas and Propagation (MECAP)*, pp. 1–4, Sept 2016.

[202] H. Taha and E. Alsusa, "Secret key exchange and authentication via randomized spatial modulation and phase shifting," *IEEE Trans. Veh. Techn.*, vol. PP, no. 99, pp. 1–1, 2017.

[203] H. Taha and E. Alsusa, "PHY-SEC: Secret key exchange and authentication via random spatial modulation and phase shifting," in *IEEE Proc. Int. Wireless Commun. Mobile Comput. Conf. (IWCMC)*, pp. 1327–1332, June 2017.

[204] Y. Wang and L. Zhang, "High security orthogonal factorized channel scrambling scheme with location information embedded for MIMO-based VLC system," in *IEEE proc. Veh. Technol. Conf. (VTC Spring)*, pp. 1–5, June 2017.

[205] K. Guan and et. al, "A computationally efficient shift-register based information scrambling approach to physical layer security in MIMO-SDM systems," in *Optical Fiber Communications Conference and Exhibition (OFC)*, pp. 1–3, March 2016.

[206] K. Guan and et. al, "Enhanced physical layer security of MIMO-SDM systems through information scrambling," in *European Conference on Optical Communication (ECOC)*, pp. 1–3, Sept 2015.

[207] Y. Tanigawa and et. al, "A physical layer security scheme employing imaginary receiver for multiuser MIMO-OFDM systems," in *Proc. IEEE Int. Conf. Commun. (ICC)*, pp. 1–6, May 2017.

[208] M. Ahmed and L. Bai, "Space time block coding aided physical layer security in gaussian MIMO channels," in *International Bhurban Conference on Applied Sciences and Technology (IBCAST)*, pp. 805–808, Jan 2017.

[209] Y. Liu and et. al, "Secrecy capacity analysis of artificial noisy MIMO channels #x2014; an approach based on ordered eigenvalues of wishart matrices," *IEEE Trans. Inf. Forensics Security*, vol. 12, pp. 617–630, March 2017.

[210] X. Chen and et. al, "Security in MIMO wireless hybrid channel with artificial noise," in *International Conference on Cyber Security of Smart Cities, Industrial Control System and Communications (SSIC)*, pp. 1–4, Aug 2015.

[211] A. Shafie and et. al, "Hybrid spatio-temporal artificial noise design for secure MIMOME-OFDM systems," *IEEE Trans. Veh. Technol.*, vol. 66, pp. 3871–3886, May 2017.

[212] Y. Kozai and T. Saba, "An artificial fast fading generation scheme for physical layer security of MIMO-OFDM systems," in *International Conference*

on Signal Processing and Communication Systems (ICSPCS), pp. 1–5, Dec 2015.

[213] X. Li and et. al, "Hybrid massive MIMO for secure transmissions against stealthy eavesdroppers," *IEEE Commun. Lett.*, vol. PP, no. 99, pp. 1–1, 2017.

[214] M. Khandaker and et. al, "Constructive interference based secure precoding," in *IEEE Proc. Int. Symp. Inf. Theory (ISIT)*, pp. 2875–2879, June 2017.

[215] X. Chen and Y. Zhang, "Mode selection in MU-MIMO downlink networks: A physical-layer security perspective," *IEEE Syst. J*, vol. 11, pp. 1128–1136, June 2017.

[216] A. E. Shafie and et. al, "Enhancing the PHY-layer security of MIMO buffer-aided relay networks," *IEEE Wireless Commun. Lett.*, vol. 5, pp. 400–403, Aug 2016.

[217] L. Zhang and et. al, "The performance of the MIMO physical layer security system with imperfect CSI," in *IEEE Conf. Commun. Netw. Security (CNS)*, pp. 346–347, Oct 2016.

[218] B. Chen and et. al, "Original symbol phase rotated secure transmission against powerful massive MIMO eavesdropper," *IEEE Access*, vol. 4, pp. 3016–3025, 2016.

[219] B. Chen and et. al, "Securing uplink transmission for lightweight single-antenna UEs in the presence of a massive MIMO eavesdropper," *IEEE Access*, vol. 4, pp. 5374–5384, 2016.

[220] L. Zhang and et. al, "Non-linear transceiver design for secure communications with artificial noise-assisted MIMO relay," *IET Communications*, vol. 11, no. 6, pp. 930–935, 2017.

[221] G. Li and A. Hu, "Virtual MIMO-based cooperative beamforming and jamming scheme for the clustered wireless sensor network security," in *IEEE proc. Int. Conf. Commun. China (ICCC)*, pp. 2246–2250, Oct 2016.

[222] E. Yaacoub and M. Al-Husseini, "Achieving physical layer security with massive MIMO beamforming," in *European Conference on Antennas and Propagation (EUCAP)*, pp. 1753–1757, March 2017.

[223] J. Tang and et. al, "Combining MIMO beamforming with security codes to achieve unconditional communication security," in *IEEE proc. Int. Conf. Commun. China (ICCC)*, pp. 105–109, Nov 2015.

[224] J. Tang, H. Wen, *et al.*, "Combining MIMO beamforming with security codes to achieve unconditional communication security," in *IEEE proc. Int. Conf. Commun. China (ICCC)*, pp. 105–109, Nov 2015.

[225] Y. Zhang and et. al, "Joint transmit antenna selection and jamming for security enhancement in MIMO wiretap channels," in *IEEE proc. Int. Conf. Commun. China (ICCC)*, pp. 1–6, Nov 2015.

[226] Y. Fan and et. al, "Physical layer security based on interference alignment in K-User MIMO Y wiretap channels," *IEEE Access*, vol. 5, pp. 5747–5759, 2017.

[227] S. Gong *et al.*, "Millimeter-wave secrecy beamforming designs for Two-Way Amplify-and-Forward MIMO relaying networks," *IEEE Trans. Veh. Technol*, vol. 66, pp. 2059–2071, March 2017.

[228] Y. Qassim and et. al, "Post-quantum hybrid security mechanism for MIMO systems," in *Int. Conf. Comput. Netw. Commun. (ICNC)*, pp. 684–689, Jan 2017.

[229] K. Jayasinghe and et. al, "Physical layer security for relay assisted MIMO D2D communication," in *IEEE Proc. Int. Conf. Commun. Workshop (ICCW)*, pp. 651–656, June 2015.

[230] H. Lei and et. al, "On secure underlay MIMO cognitive radio networks with energy harvesting and transmit antenna selection," *IEEE Trans. Green Commun. Netw.*, vol. 1, pp. 192–203, June 2017.

[231] A. Kalantari and et. al, "Directional modulation via symbol-level precoding: A way to enhance security," *IEEE J. Sel. Topics Signal Process.*, vol. 10, pp. 1478–1493, Dec 2016.

[232] M. Hafez and et. al, "Secure spatial multiple access using directional modulation," *IEEE Trans. Wireless Commun.*, 2017.

[233] Z. Li and et. al, "Cooperative jamming for secure communications in MIMO cooperative cognitive radio networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, pp. 7609–7614, June 2015.

[234] L. Li and et. al, "Improving wireless physical layer security via exploiting co-channel interference," *IEEE J. Sel. Topics Signal Process.*, vol. 10, pp. 1433–1448, Dec 2016.

[235] S. Ahn and et. al, "Enhancing physical-layer security in MISO wiretap channel with pilot-assisted channel estimation: Beamforming design for pilot jamming," in *IEEE Proc. Int. Conf. Signal Process. Commun. Systems (ICSPCS)*, pp. 1–5, Dec 2016.

[236] E. McCune, "DSSS vs. FHSS narrowband interference performance issues," *RF Signal Processing Magazine*, 2000.

[237] I. Javed and et. al, "Novel schemes for interference-resilient OFDM wireless communication," *International Journal of Communication Systems*, vol. 30, no. 6, 2017.

[238] Y. Basciftci and et. al, "Securing massive MIMO at the physical layer," in *IEEE Conf. Commun. Netw. Security (CNS)*, pp. 272–280, Sept 2015.

[239] S. Sodagari and T. Clancy, "Efficient jamming attacks on MIMO channels," in *Proc. IEEE Int. Conf. Commun. (ICC)*, pp. 852–856, IEEE, 2012.

[240] L. Li and C. Chigan, "A virtual MIMO based anti-jamming strategy for cognitive radio networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, pp. 1–6, IEEE, 2016.

[241] W. Shen and et. al, "MCR decoding: A MIMO approach for defending against wireless jamming attacks," in *IEEE Proc. Conf. Commun. Netw. Security (CNS)*, pp. 133–138, IEEE, 2014.

[242] P. Chaturvedi and K. Gupta, "Detection and prevention of various types of jamming attacks in wireless networks," *IRACST-International Journal of Computer Networks and Wireless Communications (IJCNWC), ISSN: 2250-3501*, vol. 3, no. 2, 2013.

[243] L. Atzori and et. al, "The internet of things: A survey," *Computer networks*, vol. 54, no. 15, pp. 2787–2805, 2010.

[244] F. V. den Abeele *et al.*, "Scalability analysis of large-scale LoRaWAN networks in ns-3," *IEEE Internet of Things Journal*, vol. 4, no. 6, pp. 2186–2198, 2017.

[245] D. Dolev and A. Yao, "On the security of public key protocols," *IEEE Transactions on information theory*, vol. 29, no. 2, pp. 198–208, 1983.

[246] A. Delignat-Lavaud, *On the security of authentication protocols on the web*. PhD thesis, Paris Sciences et Lettres, 2016.

[247] R. Amin, R. S. Sherratt, D. Giri, S. H. Islam, and M. K. Khan, "A software agent enabled biometric security algorithm for secure file access in consumer storage devices," *IEEE Transactions on Consumer Electronics*, vol. 63, no. 1, pp. 53–61, 2017.

[248] A. Babaei and G. Schiele, "Physical unclonable functions in the internet of things: State of the art and open challenges," *Sensors*, vol. 19, no. 14, p. 3208, 2019.

[249] C. Mesaritakis *et al.*, "Physical unclonable function based on a multi-mode optical waveguide," *Scientific reports*, vol. 8, no. 1, pp. 1–12, 2018.

[250] R. Melki, H. Noura, and A. Chehab, "Lightweight multi-factor mutual authentication protocol for IoT devices," *International Journal of Information Security*, pp. 1–16, 2019.

[251] H. Noura, R. Melki, and A. Chehab, "Secure and lightweight mutual multi-factor authentication for IoT communication systems," in *Proc. IEEE Vehicular Technology Conference (VTC2019-Fall)*, pp. 1–7, IEEE, 2019.

[252] M. Aman, K. Chua, and B. Sikdar, "Secure data provenance for the internet of things," in *Proceedings of the 3rd ACM International Workshop on IoT Privacy, Trust, and Security*, pp. 11–14, ACM, 2017.

[253] M. Wazid *et al.*, "A novel authentication and key agreement scheme for implantable medical devices deployment," *IEEE journal of biomedical and health informatics*, vol. 22, no. 4, pp. 1299–1309, 2018.

[254] M. Wazid *et al.*, "Secure authentication scheme for medicine anti-counterfeiting system in IoT environment," *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1634–1646, 2017.

[255] S. Challa *et al.*, "Secure signature-based authenticated key establishment scheme for future IoT applications," *IEEE Access*, vol. 5, pp. 3028–3043, 2017.

[256] S. Chatterjee *et al.*, "Secure biometric-based authentication scheme using chebyshev chaotic map for multi-server environment," *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 5, pp. 824–839, 2016.

[257] R. Amin and G. Biswas, "Design and analysis of bilinear pairing based mutual authentication and key agreement protocol usable in multi-server environment," *Wireless Personal Communications*, vol. 84, no. 1, pp. 439–462, 2015.

[258] M. Aman, K. Chua, and B. Sikdar, "Mutual authentication in IoT systems using physical unclonable functions," *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1327–1340, 2017.

[259] M. Abdalla, P. Fouque, and D. Pointcheval, "Password-based authenticated key exchange in the three-party setting," in *International Workshop on Public Key Cryptography*, pp. 65–84, Springer, 2005.

[260] R. Melki, H. Noura, M. Mansour, and A. Chehab, "An efficient OFDM-based encryption scheme using a dynamic key approach," *IEEE Internet of Things Journal*, vol. 6, no. 1, pp. 361–378, 2018.

[261] H. Noura, R. Melki, A. Chehab, M. Mansour, and S. Martin, "Efficient and secure physical encryption scheme for low-power wireless M2M devices," in *proc. IEEE International Wireless Communications & Mobile Computing Conference (IWCMC)*, pp. 1267–1272, IEEE, 2018.

[262] H. Noura, A. Chehab, L. Sleem, M. Noura, R. Couturier, and M. Mansour, "One round cipher algorithm for multimedia IoT devices," *Multimedia Tools and Applications*, pp. 1–31, Jan. 2018.

[263] J. Zhang, T. Duong, A. Marshall, and R. Woods, "Key generation from wireless channels: A review," *IEEE Access*, vol. 4, pp. 614–626, 2016.

[264] H. Noura, A. Chehab, L. Sleem, M. Noura, R. Couturier, and M. Mansour, "One round cipher algorithm for multimedia iot devices," *Multimedia Tools and Applications*, pp. 1–31, 2018.

[265] L. Bassham *et al.*, "A statistical test suite for random and pseudorandom number generators for cryptographic applications," NIST Pubs 800-22 Rev 1a, NIST, Sept. 2010.

[266] J. L. Rodgers and W. Nicewander, "Thirteen ways to look at the correlation coefficient," *The American Statistician*, vol. 42, no. 1, pp. 59–66, 1988.

[267] A. Massoudi *et al.*, "Overview on selective encryption of image and video: challenges and perspectives," *EURASIP Journal on Information Security*, vol. 2008, Dec. 2008.

[268] N. Maharaja, B. Mishra, and R. Bansode, "Performance evaluation of spatial multiplexing MIMO-OFDM system using MMSE detection under frequency selective rayleigh channel," *Global Journal of Computer Science and Technology*, 2016.

[269] A. Omri and R. Bouallegue, "New transmission scheme for MIMO-OFDM," *International Journal of Next Generation Network*, vol. 3, no. 1, pp. 11–19, 2011.

[270] G. Tsoulos, *MIMO system technology for wireless communications*. CRC press, 2006.

[271] "Spatial multiplexing - matlab & simulink - mathworks france." `https://fr.mathworks.com/help/comm/examples/spatial-multiplexing.html`, Janvier 2019.

[272] X. Li, H. C. Huang, A. Lozano, and G. J. Foschini, "Reduced-complexity detection algorithms for systems using multi-element arrays," in *Globecom'00-IEEE. Global Telecommunications Conference. Conference Record (Cat. No. 00CH37137)*, vol. 2, pp. 1072–1076, IEEE, 2000.

[273] N. Tippenhauer, K. Rasmussen, and S. Capkun, "Physical-layer integrity for wireless messages," *Computer Networks*, vol. 109, pp. 31–38, 2016.

[274] I. Damgård, "A design principle for hash functions," in *Proceedings of the 9th Annual International Cryptology Conference on Advances in Cryptology*, CRYPTO '89, (London, UK, UK), pp. 416–427, Springer-Verlag, 1990.

[275] R. C. Merkle, "A certified digital signature," in *Proceedings on Advances in cryptology*, CRYPTO '89, (New York, NY, USA), pp. 218–238, Springer-Verlag New York, Inc., 1989.

[276] J. Amigó, L. Kocarev, and J. Szczepanski, "Theory and practice of chaotic cryptography," *Physics Letters A*, vol. 366, no. 3, pp. 211–216, 2007.

[277] N. Masuda, G. Jakimoski, K. Aihara, and L. Kocarev, "Chaotic block ciphers: from theory to practical algorithms," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 53, no. 6, pp. 1341–1352, 2006.

[278] H. Noura, M. Noura, A. Chehab, M. Mansour, and R. Couturier, "Efficient and secure cipher scheme for multimedia contents," *Multimedia Tools and Applications*, pp. 1–30, 2018.

[279] H. Guo, C. Chen, Y. Gao, X. Li, and J. Jin, "A secure three-factor multiserver authentication protocol against the honest-but-curious servers," *Wireless Communications and Mobile Computing*, vol. 2018, 2018.

[280] G. Ohtake, R. Safavi-Naini, and L. Zhang, "Outsourcing scheme of ABE encryption secure against malicious adversary," *Computers & Security*, 2019.

[281] J. Li, T. Isobe, and K. Shibutani, "Converting meet-in-the-middle preimage attack into pseudo collision attack: Application to SHA-2," in *International Workshop on Fast Software Encryption*, pp. 264–286, Springer, 2012.

[282] M. Bellare and T. Kohno, "Hash function balance and its impact on birthday attacks," in *International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 401–418, Springer, 2004.

[283] S. Rupanagudi *et al.*, "A further optimized mix column architecture design for the advanced encryption standard," in *Proc. IEEE International Conference on Knowledge and Smart Technology (KST)*, pp. 181–185, IEEE, 2019.

[284] Y. Wang, Q. Chen, and G. Yu, "Multi-homing in unlicensed LTE networks," in *IEEE proc. International Conference on Communications (ICC)*, pp. 1–6, IEEE, 2018.

[285] R. Melki, M. Mansour, and A. Chehab, "A fairness-based congestion control algorithm for multipath TCP," in *IEEE Proc. Wireless Commun. Net. Conf. (WCNC)*, IEEE, 2018.

[286] R. Naves *et al.*, "A framework for evaluating physical-layer network coding gains in multi-hop wireless networks," *IEEE Transactions on Mobile Computing*, 2018.

[287] K. Fouli *et al.*, "Random linear network coding (RLNC)-based symbol representation," in *draft-heide-nwcrg-rlnc-00*, 2019.

[288] P. Chou, Y. Wu, and K. Jain, "Practical network coding," in *Proceedings of the annual Allerton conference on communication control and computing*, vol. 41, pp. 40–49, The University; 1998, 2003.

[289] P. Zhang *et al.*, "A lightweight encryption scheme for network-coded mobile ad hoc networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 9, pp. 2211–2221, 2013.

[290] H. Noura *et al.*, "ERSS-RLNC: Efficient and robust secure scheme for random linear network coding," *Computer Networks*, vol. 75, pp. 99 – 112, 2014.

[291] D. Gomez *et al.*, "Enhanced opportunistic random linear source/network coding with cross-layer techniques over wireless mesh networks," in *Wireless Days (WD), IFIP*, pp. 1–4, IEEE, 2014.

[292] C. de Alwis *et al.*, "Towards minimising the coefficient vector overhead in random linear network coding," in *proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5127–5131, IEEE, 2013.

[293] T. Ho *et al.*, "A random linear network coding approach to multicast," *IEEE TRANS. INFORM. THEORY*, vol. 52, no. 10, pp. 4413–4430, 2006,.

[294] J. Ying *et al.*, "Vandermonde factorization of hankel matrix for complex exponential signal recovery—application in fast NMR spectroscopy," *IEEE Transactions on Signal Processing*, vol. 66, no. 21, pp. 5520–5533, 2018.

[295] P. Jindal and S. Makkar, "Modified RC4 variants and their performance analysis," in *Microelectronics, Electromagnetics and Telecommunications*, pp. 367–374, Springer, 2019.

[296] S. Chhabra and K. Lata, "Enhancing data security using obfuscated 128-bit AES algorithm-an active hardware obfuscation approach at RTL level," in *IEEE proc. International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pp. 401–406, IEEE, 2018.

[297] P. Prasada *et al.*, "FPGA implementation of parallel transformative approach in AES algorithm," in *Information and Communication Technology for Competitive Strategies*, pp. 333–340, Springer, 2019.