Authentication Enhancement in Command and Control Networks

(A Study in Vehicular Ad-Hoc Networks)

Mahmoud Ahmed Shawky Ahmed

Submitted in fulfilment of the requirements for the
Degree of Doctor of Philosophy

School of Engineering
College of Science and Engineering
University of Glasgow

December 2023

Abstract

Intelligent transportation systems contribute to improved traffic safety by facilitating real-time communication between vehicles. By using wireless channels for communication, vehicular networks are susceptible to a wide range of attacks, such as impersonation, modification, and replay. In this context, securing data exchange between intercommunicating terminals, e.g., vehicle-to-everything (V2X) communication, constitutes a technological challenge that needs to be addressed. Hence, message authentication is crucial to safeguard vehicular ad-hoc networks (VANETs) from malicious attacks. The current state-of-the-art for authentication in VANETs relies on conventional cryptographic primitives, introducing significant computation and communication overheads. In this challenging scenario, physical (PHY)-layer authentication has gained popularity, which involves leveraging the inherent characteristics of wireless channels and the hardware imperfections to discriminate between wireless devices. However, PHY-layer-based authentication cannot be an alternative to crypto-based methods as the initial legitimacy detection must be conducted using cryptographic methods to extract the communicating terminal secret features. Nevertheless, it can be a promising complementary solution for the re-authentication problem in VANETs, introducing what is known as "cross-layer authentication." This thesis focuses on designing efficient cross-layer authentication schemes for VANETs, reducing the communication and computation overheads associated with transmitting and verifying a crypto-based signature for each transmission. The following provides an overview of the proposed methodologies employed in various contributions presented in this thesis.

1. **The first cross-layer authentication scheme**: A four-step process represents this approach: initial crypto-based authentication, shared key extraction, re-authentication via a PHY challenge-response algorithm, and adaptive adjustments based on channel conditions. Simulation results validate its efficacy, especially in low signal-to-noise ratio (SNR) scenarios while proving its resilience against active and passive attacks.

2. **The second cross-layer authentication scheme**: Leveraging the spatially and temporally correlated wireless channel features, this scheme extracts high entropy shared keys that can be used to create dynamic PHY-layer signatures for authentication. A 3-Dimensional (3D) scattering Doppler emulator is designed to investigate the scheme's performance at different speeds of a moving vehicle and SNRs. Theoretical and hardware implementation analyses prove the scheme's capability to support high detection probability for an

acceptable false alarm value ≤ 0.1 at SNR ≥ 0 dB and speed ≤ 45 m/s.

3. **The third proposal: Reconfigurable intelligent surfaces (RIS) integration for improved authentication**: Focusing on enhancing PHY-layer re-authentication, this proposal explores integrating RIS technology to improve SNR directed at designated vehicles. Theoretical analysis and practical implementation of the proposed scheme are conducted using a 1-bit RIS, consisting of 64×64 reflective units. Experimental results show a significant improvement in the P_d, increasing from 0.82 to 0.96 at SNR $= -6$ dB for multicarrier communications.

4. **The fourth proposal: RIS-enhanced vehicular communication security**: Tailored for challenging SNR in non-line-of-sight (NLoS) scenarios, this proposal optimises key extraction and defends against denial-of-service (DoS) attacks through selective signal strengthening. Hardware implementation studies prove its effectiveness, showcasing improved key extraction performance and resilience against potential threats.

5. **The fifth cross-layer authentication scheme**: Integrating PKI-based initial legitimacy detection and blockchain-based reconciliation techniques, this scheme ensures secure data exchange. Rigorous security analyses and performance evaluations using network simulators and computation metrics showcase its effectiveness, ensuring its resistance against common attacks and time efficiency in message verification.

6. **The final proposal: Group key distribution**: Employing smart contract-based blockchain technology alongside PKI-based authentication, this proposal distributes group session keys securely. Its lightweight symmetric key cryptography-based method maintains privacy in VANETs, validated via Ethereum's main network (*MainNet*) and comprehensive computation and communication evaluations.

The analysis shows that the proposed methods yield a noteworthy reduction, approximately ranging from 70% to 99%, in both computation and communication overheads, as compared to the conventional approaches. This reduction pertains to the verification and transmission of 1000 messages in total.

Keywords: Conditional privacy-preservation, Cross-layer authentication, PHY-layer authentication, PHY-layer security, Reconfigurable intelligent surfaces, Vehicular ad-hoc networks.

Contents

List of Tables

List of Figures

List of Acronyms

AAD average authentication delay

ACPPA A conditional privacy preservation authentication algorithm

AES advanced encryption standard

AFE analogue front-end

AGC automatic gain control

AN artificial noise

APLR average packet loss ratio

ARQ automatic repeat request

AVISPA automated validation of internet security protocols and applications

AWGN additive white Gaussian noise

BAN Burrows–Abadi–Needham

BCSKE blockchain-based secret key extraction

BER bit error rate

BGR bit generation rate

BMR bit mismatch rate

BP bilinear pairing

BW bandwidth

CA certificate authority

CCH control channel

CDF cumulative distribution function

CFO carrier frequency offset

CFR channel frequency response

CIR channel impulse response

CLT central limit theorem

CL-AtSe constraint logic-based attack searcher

CNN convolution neural network

CP cyclic prefix

CPAS clustering-based PHY-layer authentication scheme

CPNN convolution pre-processing neural network

CPPA conditional privacy-preserving authentication

CRC cyclic redundancy check

CRL certificate revocation list

CSI channel state information

CSR certificate signing request

DHP Diffie-Hellman problem

DL deep learning

DLP discrete logarithm problem

DNN deep neural network

DoS Denial-of-service

DPSK differential phase shift keying

DSRC dedicated short-range communication

ECC elliptic curve cryptosystem

ECDLP elliptic curve discrete logarithm problem

ECDSA elliptic curve digital signature algorithm

E2E end-to-end

FCC Federal Communications Commission

FFT fast Fourier transform

FPGA field-programmable gate array

FR freshness rule

GLRT generalized likelihood ratio test

GS group signature

GTEA Gaussian tag-embedded authentication

HB-PL Hopper-Blum-based PHY-layer

HLPSL high-level protocol specification language

HMAC hash message authentication code

ICs integrated circuits

ID identity

ID-MAP identity-based message authentication scheme using proxy vehicles

IF intermediate format

IoT internet-of-things

IP internet protocol

IQI I/Q amplitude and phase shift imbalance

I/Q in-phase/quadrature

JR jurisdiction rule

LDPC low-density parity-check

LIAP local identity-based anonymous message authentication protocol

LoS line-of-sight

LRT likelihood ratio test

LS least square

LTS long training sequence

L-RSU leader RSU

MAC media access control

MAMO multi-attributes multi-observation

MBER message bit error rate

MIMO multiple-input multiple-output

MITM Man-in-the-Middle

MMR message meaning rule

MMSE minimum mean-square error

MRC maximal ratio combining

MSE mean square error

M-RSUs member RSUs

NERA new and efficient RSU based authentication

NIST national institute of standards and technology

NLoS non-line-of-sight

NVR nonce verification rule

OBU onboard unit

OFDM orthogonal frequency-division multiplexing

OFMC on-the-fly model checker

PDF probability density function

PDP power delay profile

PHY physical

PII personally identifiable information

PIN positive-intrinsic-negative

PKCs public key certificates

PKI public key infrastructure

PoS proof-of-stack

PoW proof-of-work

PSD power spectral density

PUF physically unclonable function

PHY-CRAM physical layer challenge-response authentication mechanism

PHY-FTM PHY-layer feature tracking mechanism

PHY-PCRAS PHY-layer challenge-response authentication scheme

PHY-SIAM PHY-layer signature-based identity authentication mechanism

RCA region certification authority

RFF radio frequency fingerprinting

RIS reconfigurable intelligent surface

ROC receiver operating characteristic

ROM random oracle model

RSA Rivest-Shamir-Adleman

RSS received signal strength

RSU roadside unit

RTA region trust authority

RUs reflecting units

Rx receiver

SATMC SAT-based model checker

SBGR secret bit generation rate

SDR software-defined radio

SNR signal-to-noise ratio

SPRT sequential probability ratio test

STR signal-to-tag power allocation ratio

STS short training sequence

TA trusted authority

TA4SP tree automata-based on automatic approximations for analysis of security protocol

TBER tag bit error rate

TCP transmission control protocol

TDD time division duplex

TESLA timed efficient stream loss-tolerant authentication

TPDs tamper-proof devices

TTP trusted third party

Tx transmitter

USRP universal software radio peripheral

VANET vehicular ad-hoc network

V2I vehicle-to-infrastructure

V2V vehicle-to-vehicle

V2X vehicle-to-everything

WFRFT weighted fractional Fourier transform

WSN wireless sensor network

2D two-dimensional

3D three-dimensional

5G fifth-generation

6G sixth-generation

List of Publications

During my PhD studies, I have been involved in several publications as the first author and co-author. While some of these publications are directly related to my thesis topic, others are not. The following is a comprehensive list of all my publications during this period.

A. *Articles*

(1) **M. A. Shawky**, M. Bottarelli, G. Epiphaniou, and P. Karadimas, "An Efficient Cross-Layer Authentication Scheme for Secure Communication in Vehicular Ad-hoc Networks", *IEEE Transactions on Vehicular Technology*, vol. 72, no. 7, 2023.

(2) **M. A. Shawky**, M. Usman, M. A. Imran, Q. H. Abbasi, S. Ansari, and A. Taha, "Adaptive Chaotic Map-based Key Extraction for Efficient Cross-Layer Authentication in VANETs", *Vehicular Communications*, vol. 39, 2023.

(3) **M. A. Shawky**, M. Usman, D. Flynn, M. A. Imran, Q. H. Abbasi, S. Ansari, and A. Taha, "Blockchain-based Secret Key Extraction for Efficient and Secure Authentication in VANETs", *Journal of Information Security and Applications*, vol. 74, 2023.

(4) **M. A. Shawky**, A. Jabbar, M. Usman, M. A. Imran, Q. H. Abbasi, S. Ansari, and A. Taha, "Efficient Blockchain-Based Group Key Distribution for Secure Authentication in VANET", *IEEE Networking Letters*, vol. 5, no. 1, pp. 64-68, 2023.

(5) **M. A. Shawky**, S. T. Shah, M. S. Mollel, J. U. Kazim, Q. H. Abbasi, M. A. Imran, S. Ansari, and A. Taha, "Reconfigurable Intelligent Surface-Assisted Cross-Layer Authentication for Secure and Efficient Vehicular Communications", *IEEE Transactions on Wireless Communications*, Under Review.

(6) **M. A. Shawky**, S. T. Shah, Q. H. Abbasi, M. A. Imran, S. F. Hasan, S. Ansari, and A. Taha, "RIS Enabled Secret Key Generation for Secured Vehicular Communication in the presence of Denial-of-Service Attacks", *MDPI Sensors*, vol. 23, 2023.

(7) **M. A. Shawky**, S. T. Shah, Q. H. Abbasi, M. A. Imran, S. Ansari, and A. Taha, "How Secure Are Our Roads? An In-Depth Review of Authentication in Vehicular Communications", *Vehicular Communications*, Under Review.

(8) A. Taha, B. Barakat, M. M. A. Taha, **M. A. Shawky**, C. S. Lai, S. Hussain, M. Z. Abdeen and Q. H. Abbasi, "A Comparative Study of Single and Multi-Stage Forecasting

Algorithms for the Prediction of Electricity Consumption Using a UK-National Health Service (NHS) Hospital Dataset", *Future Internet*, vol. 15, no. 4, 2023.

(9) D. Mitchell, P. D. E. Baniqued, et al., **M. A. Shawky**, et al., "Lessons Learned: Symbiotic Autonomous Robot Ecosystem for Nuclear Environments", *IET Cyber-Systems and Robotics*, 2023.

(10) S. T. Shah, **M. A. Shawky**, J. U. Kazim, S. F. Hasan, S. Ansari, A. Taha, T. J. Cui, M. A. Imran, and Q. H. Abbasi, "Reconfigurable Intelligent Surface-Enabled Indoor Localisation: An Experimental Analysis", *Nature Electronics*, Under Review.

(11) A. G. Abdellatifa, A. A. Salama, H. S. Zied, A. A. Elmahallawy, and **M. A. Shawky**, "An Improved Indoor Positioning based on Crowd-Sensing Data Fusion and Particle Filter", *Journal of Physical Communication*, vol. 61, 2023.

B. *Conference Proceedings*

(1) **M. A. Shawky**, Q. H. Abbasi, M. A. Imran, S. Ansari, and A. Taha, "Cross-Layer Authentication based on Physical-Layer Signatures for Secure Vehicular Communication", IEEE Intelligent Vehicles Symposium (IV), Aachen, Germany, 2022.

(2) **M. A. Shawky**, M. Usman, M. A. Imran, Q. H. Abbasi, S. Ansari, and A. Taha, "Adaptive and Efficient Key Extraction for Fast and Slow Fading Channels in V2V Communications", IEEE 96^{th} Vehicular Technology Conference (VTC2022-Fall), London, United Kingdom, 2022.

(3) J. Kaur, **M. A. Shawky**, M. S. Mollel, O. Popoola, M. A. Imran, Q. H. Abbasi, and H. T. Abbas, "AI-Enabled CSI Fingerprinting for Indoor Localisation towards Context-Aware Networking in 6G", IEEE Wireless Communications and Networking Conference (WCNC), Glasgow, Scotland, UK, 2023.

(4) B. A. Zaidi, **M. A. Shawky**, A. Taha, Q. H. Abbasi, M. A. Imran, and S. Ansari, "An Efficient Deep Learning-based Spectrum Awareness Approach for Vehicular Communication", IEEE Wireless Communications and Networking Conference (WCNC), Glasgow, Scotland, UK, 2023.

(5) A. Jabbar, M. A. Jamshed, **M. A. Shawky**, Q. H. Abbasi, M. A. Imran, and M. Ur Rehman, "Multi-Gigabit Millimeter-Wave Industrial Communication: a Solution for Industry 4.0 and Beyond", 2022 IEEE Global Communications Conference (GLOBECOM), Rio de Janeiro, Brazil, 2022.

(6) S. Hafeez, **M. A. Shawky**, M. Al-Quraan, L. Mohjazi, M. A. Imran, and Y. Sun, "BETA-UAV: Blockchain-based Efficient and Trusted Authentication for UAV Communication", 2022 IEEE 22nd International Conference on Communication Technology (ICCT), Nanjing, China, 2022.

(7) M. Z. Khan, A. Taha, M. Farooq, **M. A. Shawky**, M. Imran, and Q. H. Abbasi, "Comparative Analysis of Artificial Intelligence on Contactless Human Activity Localization", International Telecommunications Conference (ITC-Egypt'2022), Alexandria, Egypt, 2022.

(8) C. Yun, **M. A. Shawky**, and S. Ansari, "An Optimized Digital Signature Algorithm for Efficient and Secure Authentication in VANETs", International Telecommunications Conference (ITC-Egypt'2023), Alexandria, Egypt, 2023.

(9) A. A. Farid, A. T. Khalil, and **M. A. Shawky**, "Optimized Performance of Attacks Detection in WSN based on Machine Learning Algorithms", International Telecommunications Conference (ITC-Egypt'2023), Alexandria, Egypt, 2023.

(10) M. Farooq, **M. A. Shawky**, A. Fatima, A. Tahir, M. Z. Khan, H. Abbas, M. Imran, Q. H. Abbasi, and A. Taha, "Room-Level Activity Classification from Contextual Electricity Usage Data in a Residential Home", International Telecommunications Conference (ITC-Egypt'2023), Alexandria, Egypt, 2023.

(11) A. Jabbar, J. U. Kazim, **M. A. Shawky**, Q. Abbasi, M. A. Imran, and M. Ur-Rehman, "Over-The-Air (OTA) Measurements and Characterization of Millimeter-Wave Antennas Using Benchtop Compact Antenna Test Range (CATR)", In 2024 18^{th} European Conference on Antennas and Propagation (EuCAP), Glasgow, UK, Under Review.

C. *Book chapters*

(1) **M. A. Shawky**, R. M. Sohaib, M. Usman, Q. H. Abbasi, M. A. Imran, S. Ansari, and A. Taha, "Cooperative Intelligent Transport Systems for Net-Zero", in The Role of 6G and Beyond on the Road to Net-Zero Carbon, IET.

Acknowledgements

I would like to express my deepest gratitude to the divine Almighty God for the unwavering sustenance of my health and well-being throughout the duration of my PhD. I am eternally grateful for the wisdom, inspiration, strength, and divine favour that facilitated my acquisition of the scholarship that made this academic pursuit possible. In addition, completing this thesis would not have been possible without the support and guidance of several individuals, and I would like to take this opportunity to express my sincere gratitude.

First and foremost, I want to thank Professor Muhammad Ali Imran for his unwavering support and motivation throughout my PhD. His encouragement and mentorship have been instrumental in shaping my research and personal growth. I am also grateful to Dr Ahmad Taha and Dr Shuja Ansari for their supervision and for providing me with the opportunity to pursue my PhD under their guidance. Their continuous support and valuable insights have been invaluable to my research.

I would also like to extend my gratitude to Professor Qammer H. Abbasi, Dr Muhammad Usman, Dr Syed Tariq Shah, and Dr Michael S. Mollel for their contributions, feedback, and insightful discussions during my PhD. Furthermore, I wish to acknowledge the Egyptian Armed Forces for providing me with the funding to pursue my PhD at James Watt School of Engineering, University of Glasgow, UK.

Finally, I would like to express my deepest appreciation to my parents and my wife, Wessam E. Mansour. Their unwavering support, encouragement, and sacrifices have been essential to my success. Wessam has been my pillar of strength throughout this journey, and I am forever grateful for her love and unwavering commitment to our family.

Declaration

I hereby declare that this thesis, consisting of nine chapters, represents my original work. All sources used for this research have been duly acknowledged in the reference section. Furthermore, the research conducted in this thesis adheres to the guidelines and regulations set forth by the University of Glasgow.

Chapter 1

Introduction

This chapter introduces the background and research motivation underlying this thesis, discusses the objective and main contributions, and outlines the thesis and included publications.

1.1 Background

Globally, road traffic injuries and fatalities reach about 1.3 million annually, with more than 3000 fatalities daily. According to the "2^{nd} global status report on road safety", it is expected to become the fifth leading cause of death by 2030, resulting in around 2.4 million deaths annually [1]. Moreover, between twenty to fifty million people suffer from non-fatal injuries annually, often resulting in disabilities. Economically, these accidents cost countries between 1% to 3% of their gross national product, totalling over 500 billion globally. This rise is due in part to rapid motorisation without adequate safety improvements. In 2020, the European Commission reported a decrease in fatal road crashes by about 23% compared to 2010, aiming for zero fatalities by 2050 [2]. For the next decade, a safety framework plan is published in [3] to enhance safety and efficiency in transportation, adapting technology to develop and implement intelligent road systems based on sensors' data distributed via vehicular ad-hoc network (VANET).

VANET is a form of a mobile ad-hoc network in the vehicle domain that enables vehicle-to-everything (V2X) communication (e.g., vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I)), see Fig. 1.1 [4]. This significantly enhances the performance of many traffic-related applications, including safety, mobility, and autonomy. Moreover, it reduces the carbon footprint and facilitates green transportation by providing vehicles with the ability to optimise their routes and avoid traffic congestion en route to their intended destinations. VANETs generally consist of three primary terminals: a trusted authority (TA), roadside unit (RSU), and a wireless communication device located on the vehicle, also known as an onboard unit (OBU) [5]. The following describes the role of each terminal in the network.

1. *The Trusted authority*: The TA is a trusted terminal for all vehicles and RSUs. The TA plays a critical role in facilitating secure and trustworthy communication between net-

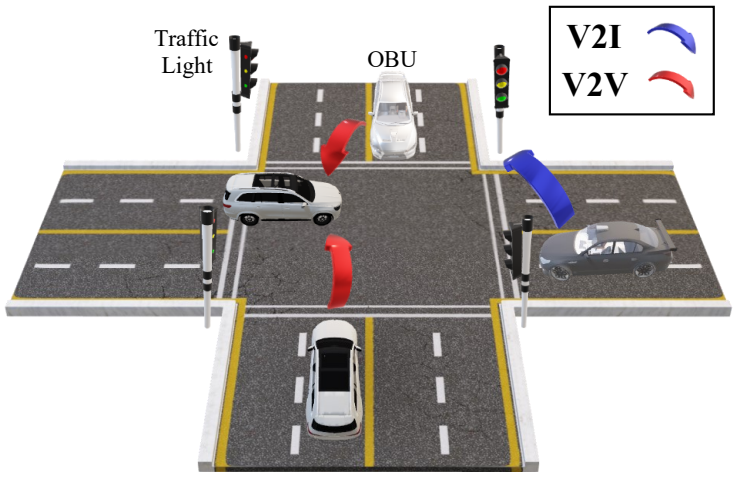


Figure 1.1: Common types of vehicular communication.

work participants, ensuring that messages transmitted between vehicles are authentic and have not been tampered with. In addition, the TA is also responsible for managing the network's membership by registering vehicles and RSUs before they can participate in network activities. In addition, the TA is responsible for revoking vehicles engaging in malicious activity, such as launching an attack or violating traffic laws [5, 6].

2. *Roadside units*: RSUs are stationary wireless devices that are deployed along the roadside infrastructure. The primary role of RSUs in VANETs is to enhance the reliability and efficiency of communication between vehicles and between vehicles and the TA. In addition, RSUs can also be used for various other applications. These include providing information about traffic conditions, road hazards, and emergency situations to vehicles in the network. They can also be used for collecting traffic data and performing traffic management tasks, such as traffic flow optimisation and congestion control [5, 6].

3. *Vehicles' OBUs*: OBUs are electronic wireless devices installed on vehicles that enable communication with other vehicles and RSUs. The primary role of OBUs in VANETs is to support the exchange of information among vehicles and between vehicles and the network infrastructure [5, 6].

In VANETs, each vehicle transmits a safety-related message containing information on location, speed, and heading at a transmission rate of $100-300$ *msec* using the dedicated short-range communication (DSRC) protocol [6]. In the DSRC protocol, a 75 MHz bandwidth (BW) has been allocated by the U.S. Federal Communications Commission (FCC) for VANET applications over the frequency band 5.85 GHz to 5.925 GHz, as shown in Fig. 1.2 [6].

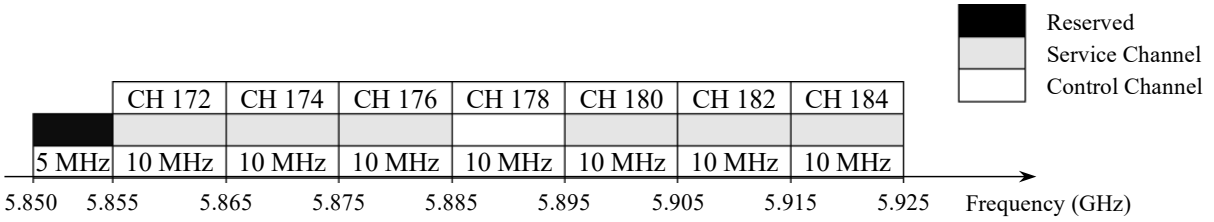


Figure 1.2: The U.S. dedicated spectrum for the IEEE 802.11p [6].

1.2 Research motivation

The open-access nature of wireless communication makes it vulnerable to typical attacks [7]. For instance, a malicious vehicle can frame an emergency to mislead other drivers into slowing down and braking; impersonate a legitimate vehicle; replay a significant number of bogus messages, which creates an unrealistic traffic situation. These attacks can cause serious problems, e.g., traffic jams or accidents. Therefore, message authentication must be established to identify the sender's legitimacy. Generally, there are three common types of authentication in VANETs: public key infrastructure (PKI)-based, identity (ID)-based, and group signature (GS)-based [5]. In PKI-based authentication, each vehicle has a pair of keys, private and public keys [8]. The private key is kept secret and is used to generate digital signatures on messages. For verification, the public key is attached to the transmitted message in the form of a digital certificate signed by the TA. In ID-based authentication, the vehicle's identifier, such as the vehicle identification number, is used as its public key, which can be used to verify signatures generated by the vehicle's private key. This approach eliminates the need for a separate public key infrastructure, as the identifier itself serves as the public key [9]. In GS-based authentication, group members generate the signature (σ) on behalf of the group using their secret keys, while the recipient verifies σ using the group's public key [10]. The signature is generated in such a way that it cannot be traced back to the specific member who generated it, offering anonymity and privacy preservation. However, these methods require complex cryptographic operations, leading to high computation and communication costs for transmitting and verifying messages.

To overcome this limitation, PHY-layer authentication techniques have emerged as a promising solution to reduce the overheads associated with upper-layer cryptographic approaches [11]. This technique employs the unique features of wireless channels, such as channel amplitude and phase responses [12], and the hardware impairments, such as analogue front-end imperfections and carrier frequency offset [13], to discriminate between terminals. Nevertheless, PHY-layer-based authentication cannot provide a completely alternative solution since an initial identity verification of the corresponding terminal is still needed using upper-layer-based authentication. Fig. 1.3 shows the different layers of the network protocol stack [14].

Cross-layer authentication is an emerging research area that focuses on integrating authentication mechanisms across different layers of the communication protocol stack in wireless

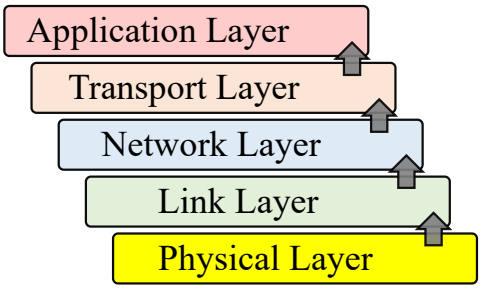


Figure 1.3: Layered protocol architecture [14].

networks. Unlike traditional authentication methods that operate independently at different layers, cross-layer authentication techniques leverage the interdependence between different layers to enhance the security and efficiency of the authentication process. The idea behind cross-layer authentication is to combine the strengths of different layers to create a more robust authentication mechanism. By sharing information and resources between layers, cross-layer authentication can improve the accuracy and reliability of authentication, while reducing the overhead and latency associated with traditional authentication methods.

Cross-layer authentication can be applied to different layers of the communication protocol stack, including the physical layer, the data link layer, and the network layer [15]. At the physical layer, cross-layer authentication can use techniques such as PHY-layer key generation and channel-based authentication to establish a shared secret key between communicating parties. This key can then be used to encrypt subsequent communications and prevent unauthorised access. At the data link layer, cross-layer authentication can use techniques such as media access control (MAC)-layer authentication to verify the identity of communicating parties and prevent spoofing attacks. At the network layer, cross-layer authentication can use techniques such as network-layer authentication to authenticate the routing information and prevent attacks such as packet injection. The existing cross-layer authentication schemes in VANETs are developed by integrating the physical layer with the upper layer (cryptographic) operations [16]. This integration should be rational and practical to support the application nature in terms of dynamicity, resource availability, and channel conditions. Consequently, selecting the appropriate PHY-layer-based technique for re-authentication is essential to provide reliable communication.

In addition, the current state-of-the-art of PHY-layer authentication relies on common assumptions and possesses inherent limitations for secure implementation within V2X communication scenarios. In this challenging scenario, it becomes imperative to investigate the integration of cryptographic operations based on number theory and smart contract-based blockchain technology. These approaches aim to mitigate the constraints associated with PHY-layer methods, thus offering potential solutions to their limitations. Moreover, the detection probability of PHY-layer-based techniques is influenced by signal-to-noise ratio (SNR) values [16]; a higher SNR correlates with increased detection probability. Thus, exploring the integration of advanced sixth-generation (6G) technologies such as reconfigurable intelligent surface (RIS) technology

emerges as a pivotal aspect for the evolution of VANET networks. RIS technology plays a crucial role by strengthening signals directed toward network terminals, thereby improving the detection probability of the PHY-layer authentication process.

1.3 Research scope

The research scope of this thesis is to investigate and develop novel cross-layer authentication methodologies for enhancing the security and efficiency of VANETs. The focus is on mitigating the computation and communication overheads associated with conventional cryptographic approaches while ensuring the security and privacy of the exchanged data in vehicular communication networks. In addition, this thesis emphasises the significance of integrating cutting-edge 6G technology, specifically RIS to enhance the performance of PHY-layer authentication.

1.4 Aims, research questions, and objectives

This section summarises the aims, research questions, and objectives of this thesis, as follows.

1.4.1 Aims

This thesis aims to advance vehicular communication security by introducing efficient cross-layer authentication techniques that combine cryptographic primitives with PHY-layer attributes. The outcomes include reduced computation and communication overheads, improved security resilience against security threats, and enhanced authentication performance for VANETs.

1.4.2 Research questions

The following research questions have been formulated to guide this thesis:

1. Q_1: Can PHY-layer authentication be considered as an alternative to existing cryptography-based authentication methods for VANET applications?

2. Q_2: Which PHY-layer authentication methods can be employed as an effective technique for re-authenticating communicating terminals in VANETs?

3. Q_3: What can be done to overcome some of the limitations of existing PHY-layer authentication and secret key extraction using current crypto-based methods?

4. Q_4: How can recent advancements in wireless communication, specifically the RIS technology, be harnessed to improve authentication and key extraction performance in specific scenarios of vehicular communications?

These research questions will guide the investigation and analysis of the current state-of-the-art techniques and technologies in the field of VANET authentication, leading to new insights and recommendations for improving the security and reliability of vehicular communications.

1.4.3 Objectives

The primary research objectives encompass the following:

- *Objective 1*: This thesis delves into the design and implementation of innovative cross-layer authentication strategies that leverage both cryptographic and PHY-layer techniques. The aim is to minimise the complexities and resource requirements while maintaining robust security measures for VANETs.

- *Objective 2*: Investigating the potential of PHY-layer authentication to complement traditional cryptographic mechanisms for VANET security. This includes exploring methods to extract terminal secret features through cryptographic means while utilising PHY-layer attributes for re-authentication purposes.

- *Objective 3*: This thesis explores the applicability of advanced cryptographic methods derived from number theory and emerging technologies such as blockchain to address some of the performance limitations of PHY-layer authentication methods and enhance the security performance of VANETs.

- *Objective 4*: This thesis explores the integration of RIS technology to improve the PHY-layer authentication performance. The investigation encompasses enhancing the SNR directed toward designated vehicles, thereby improving detection probabilities and the system's resistance against active attackers.

- *Objective 5*: This thesis aims to conduct comprehensive performance evaluations to quantify the efficiency, and overhead reductions attained through the proposed schemes. This assessment encompasses simulations, analytical models, and practical experiments, measuring the improvements in efficiency and overhead reduction achieved by these schemes.

- *Objective 6*: This thesis involves a thorough security analysis to validate the resilience of the proposed methodologies against diverse attacks. Employing formal verification techniques, including logic-based analysis, aiming to rigorously evaluate the security robustness of the proposed authentication schemes.

- *Objective 7*: This thesis explores the practicality of the proposed methods through real-world implementations and experiments. By emphasising practical implementation, it validates the performance and security enhancements of the proposed approaches.

1.5 Thesis outline and research publications

This thesis includes a compilation of publications, and the following list introduces these works and their corresponding chapters, as listed in Table 1.1. The present research work is structured into nine chapters, with each chapter delving into specific aspects of the research inquiry. As per the referencing conventions adopted in this study, Chapter 2 follows the published work in [17]. Chapters 3, and 4 have been supported by published works cited in [18, 19] and [20, 21], respectively. Furthermore, Chapters 5, 6, 7, and 8 incorporate references to published works, which are presented in [22], [23], [24], and [25], respectively. Finally, Chapter 9 provides a comprehensive summary of the thesis, elucidating key insights from the research findings for future research. The objectives of each individual chapter are depicted in Fig. 1.4.

Table 1.1: Included papers within the thesis's chapters

Chapter	Included papers (article/conference)	Status	Cited in	Chapter objective
2	**M. A. Shawky**, S. T. Shah, M. Usman, Q. H. Abbasi, M. A. Imran, S. Ansari, and A. Taha "How Secure Are Our Roads? An In-Depth Review of Authentication in Vehicular Communications", *Vehicular Communications*, Under Review.	Under Review	Reference [17]	Exploring the current state-of-the-art
3	**M. A. Shawky**, M. Bottarelli, G. Epiphaniou, and P. Karadimas, "An Efficient Cross-Layer Authentication Scheme for Secure Communication in Vehicular Ad-hoc Networks", *IEEE Transactions on Vehicular Technology*, vol. 72, 2023.	Published	Reference [18]	Reducing the computation and communication costs of traditional authentication schemes
	M. A. Shawky, Q. H. Abbasi, M. A. Imran, S. Ansari, and A. Taha, "Cross-Layer Authentication based on Physical-Layer Signatures for Secure Vehicular Communication", IEEE Intelligent Vehicles Symposium (IV), Aachen, Germany, 2022.	Presented and published	Reference [19]	
4	**M. A. Shawky**, M. Usman, M. A. Imran, Q. H. Abbasi, S. Ansari, and A. Taha, "Adaptive Chaotic Map-based Key Extraction for Efficient Cross-Layer Authentication in VANETs", *Vehicular Communications*, vol. 39, 2023.	Published	Reference [20]	Overcoming the PHY-Layer limitations arising from inter-vehicle spacing of $\lambda/2$
	M. A. Shawky, M. Usman, M. A. Imran, Q. H. Abbasi, S. Ansari, and A. Taha, "Adaptive and Efficient Key Extraction for Fast and Slow Fading Channels in V2V Communications", VTC2022-Fall, London, United Kingdom, 2022.	Presented and published	Reference [21]	
5	**M. A. Shawky**, S. T. Shah, M. S. Mollel, J. U. Kazim, Q. H. Abbasi, M. A. Imran, S. Ansari, and A. Taha, "Reconfigurable Intelligent Surface-Assisted Cross-Layer Authentication for Secure and Efficient Vehicular Communications", *IEEE Transactions on Wireless Communications*, Under Review.	Under review	Reference [22]	Enhancing the detection probability of the PHY-layer re-authentication
6	**M. A. Shawky**, S. T. Shah, Q. H. Abbasi, M. A. Imran, S. F. Hasan, S. Ansari, and A. Taha, "RIS Enabled Secret Key Generation for Secured Vehicular Communication in the presence of Denial-of-Service Attacks", *MDPI Sensors*, vol 23, 2023.	Published	Reference [23]	Enhancing the secret key extraction performance in the presence of DoS attack
7	**M. A. Shawky**, M. Usman, D. Flynn, M. A. Imran, Q. H. Abbasi, S. Ansari, and A. Taha, "Blockchain-based Secret Key Extraction for Efficient and Secure Authentication in VANETs", *Journal of Information Security and Apps*, 2023.	Published	Reference [24]	Reconciling the mismatched bits arising from channel non-reciprocity components
8	**M. A. Shawky**, A. Jabbar, M. Usman, M. A. Imran, Q. H. Abbasi, S. Ansari, and A. Taha, "Efficient Blockchain-Based Group Key Distribution for Secure Authentication in VANET", *IEEE Networking Letters*, vol. 5, no. 1, pp. 64-68, 2023.	Published	Reference [25]	Establishing an efficient group key distribution process

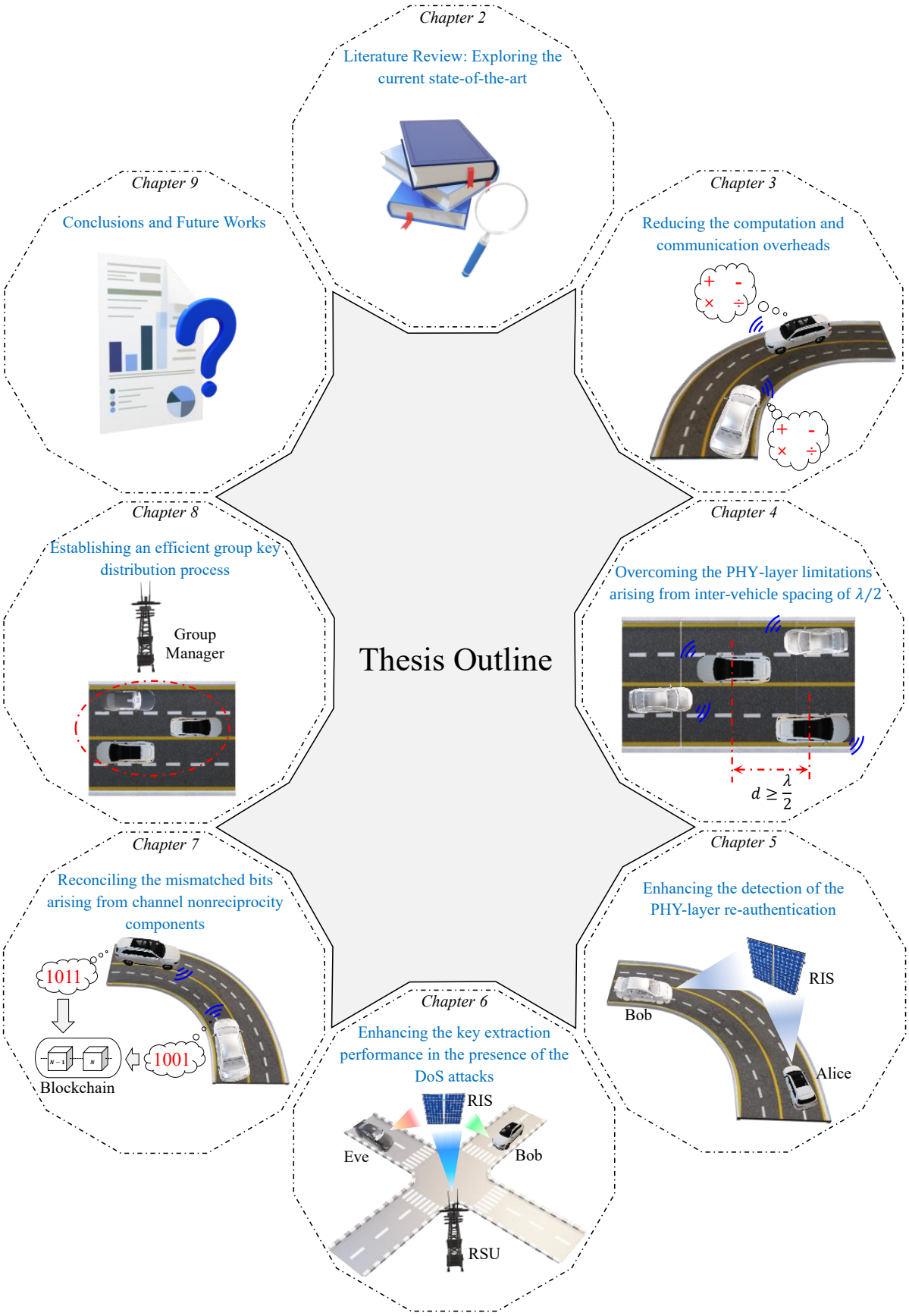


Figure 1.4: A summary of the thesis structure and each chapter's objective.

Chapter 2

Related Works

This chapter presents a systematic classification of existing authentication techniques in wireless communications. By examining the strengths and limitations of each approach, this chapter aims to provide a comprehensive overview of the current state-of-the-art of authentication in wireless communications. The effectiveness of any authentication scheme is determined by several factors, including latency, complexity, security, privacy preservation, and power consumption. In this context, the aim is to introduce the performance evaluation metrics of authentication for VANET applications in Section 2.1, followed by a discussion of the classification presented in Fig. 2.1. The classification outlined in Fig. 2.1 organises current authentication methods of ad-hoc networks into three main categories: cryptographic-based, PHY-layer-based, and cross-layer-based authentication presented in Sections 2.2, 2.3, and 2.4, respectively. The insights gained from this classification can inform the development of new and improved authentication techniques for ad-hoc wireless networks. Furthermore, this chapter includes a taxonomy of techniques for extracting secret keys at the physical layer (see Fig. 2.14), presented in Section 2.5. Finally, the outcomes of this chapter are summarised in Section 2.6.

2.1 Performance evaluation metrics

The security and privacy requirements, as well as the computation and communication overheads, constitute the primary evaluation metrics that an authentication scheme must satisfy to be deemed effective in VANETs [26, 27]. These metrics are defined as follows.

2.1.1 Security and privacy requirements

An effective authentication scheme must satisfy the following security and privacy requirements.

1. *Privacy preservation*: Privacy preservation refers to the protection of users' personally identifiable information (PII) from unauthorised access or disclosure. In VANETs, privacy preservation aims to safeguard the identity, location, and travel pattern information

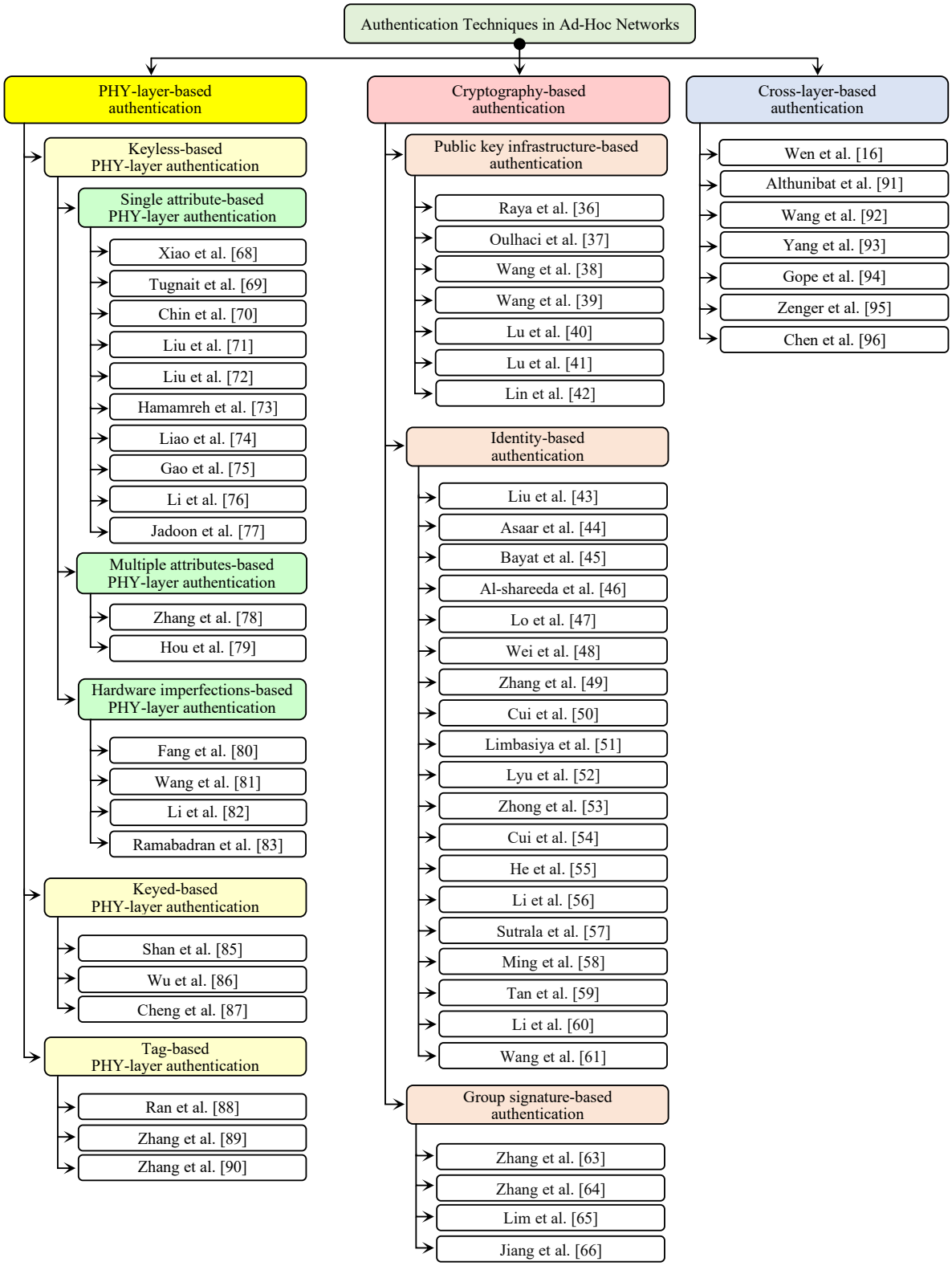


Figure 2.1: Classification of authentication schemes in wireless communications [5, 14].

of vehicles and their passengers from being revealed to unauthorised entities or attackers. The objective is to prevent the misuse of this information, such as for surveillance, tracking, or profiling purposes [5].

2. *Unlinkability*: Unlinkability ensures that distrusted terminals cannot track the transmitter behaviours by determining the origins of two different messages. It can be achieved through the use of techniques such as dynamically updated pseudonyms, which allow vehicles to hide their real identities while still being able to communicate without linking between different messages. By ensuring unlinkability, VANETs can protect the privacy of the users and prevent attackers from tracking and profiling users' activities [26].

3. *Message authentication*: Receiver's ability to authenticate every safety-related message sent from a specific terminal. The primary objective of message authentication is to prevent malicious entities from injecting false or modified messages into the network, which can compromise the safety and efficiency of vehicular communication [26].

4. *Message integrity*: Receiver's ability to detect any modification attempts on messages exchanged between vehicles. This can be ensured through the use of cryptographic techniques such as message authentication codes and digital signatures [27].

5. *Non-Repudiation*: Non-repudiation ensures that a sender cannot deny having sent a message or data to a receiver. It is a security requirement that provides proof of the origin of a message, its delivery to the intended recipient, and its contents. Non-repudiation helps to prevent disputes and false accusations that may arise in case of malicious attacks or errors. It is achieved through the use of digital signatures and certificates, which provide a unique and verifiable identification of the sender and ensure the authenticity, integrity, and confidentiality of the transmitted information [27].

6. *Traceability*: Traceability ensures the ability to track and identify the real identity of vehicles in the network, which can be used for various purposes such as traffic management, accident investigation, and law enforcement. However, traceability can also pose a threat to privacy as it may reveal sensitive information about the driver, such as their identity and travel patterns. Therefore, it is important to implement traceability in a way that balances the need for information with the privacy concerns of the users. This can be achieved through the use of anonymous identifiers, encryption, and access controls, which ensure that the data is only accessible to authorised parties and is protected from unauthorised disclosure or misuse [27].

7. *Resistance to attacks*: The primary objective of an attacker is to cause disruption to the network through the implementation of the following common attacks [28].

    - *Passive attack*: A passive attack involves eavesdropping on the communication between vehicles without altering or modifying the content of the messages. In this attack, the attacker monitors messages exchanged between vehicles to gain sensitive information, such as the vehicle's location, heading, and speed, without being detected or raising suspicions. This attack can be carried out using various techniques

such as radio frequency scanning, packet sniffing, or traffic analysis. Passive attacks are difficult to be detected as the attackers do not alter the original messages or cause any disruption in message contents. Therefore, it is important to implement security solutions such as encryption, authentication, and access control, which can prevent passive attacks by ensuring the confidentiality and integrity of messages' contents.

- *Active attacks*: An active attack involves modifying or manipulating the content of the messages exchanged between vehicles or between a vehicle and infrastructure. In this attack, the attacker aims to disrupt vehicular communication by performing impersonation, message replay, and modification. This can have severe consequences, causing accidents, traffic congestion, or misleading vehicles to the wrong destination. Active attacks can be carried out by malicious vehicles or infrastructure or by exploiting vulnerabilities in VANET protocols or applications. The following are the common types of active attacks [29, 30].
  (a) *Replay attack*: A replay attack involves retransmitting previously captured messages, with the aim of causing confusion or carrying out unauthorised actions. To avoid this attack, cryptographic techniques such as timestamps or sequence numbers ensure message freshness and uniqueness, and digital signatures, authentication codes, and secure communication channels guarantee message authenticity and integrity [29].
  (b) *Impersonation attack*: An impersonation attack involves a malicious entity acting as a legitimate entity to gain unauthorised access over the network. In this attack, the attacker pretends to be a trusted entity such as a vehicle, a traffic sign, or a roadside unit with the aim of either gaining access to sensitive information, disrupting communication, or carrying out unauthorised actions [29].
  (c) *Modification attack*: In this attack, the attacker tries to modify the data transmitted over the network to cause malicious effects or intercepts the data being transmitted and alters it in some way before forwarding it to the intended recipient. This modification can be done in various ways, such as changing the content of the message, the source or destination of the message, or the timing of the message. Some techniques that can be used to prevent such attacks include message authentication, encryption, and digital signatures [30].
  (d) *Man-in-the-Middle attack*: In this attack, the attacker may alter and relay broadcasted messages between terminals that believe they are in direct communication with each other [30].
  (e) *Sybil attack*: The attacker generates multiple fabricated identities and tries to masquerade multiple legitimate users to affect the functionality of the network. To prevent such attacks, robust security mechanisms are essential, such as authentication and verification mechanisms to ensure only legitimate vehicles are

allowed to join the network or reputation-based systems that can detect anomalies in each vehicle's behaviour.

(f) *Denial-of-service (DoS) attack*: In VANETs, the cumulative computational load increases with the number, N, of transmitted signatures. The sum of transmitted signatures represented as $\sum_{i=1}^{N} \sigma_i$, reflects the computational demands. As N grows, the network's computational resources might become insufficient, resulting in performance degradation. This potential strain on computational resources highlights the need to carefully consider the scalability of signature transmission within VANETs to maintain optimal network performance. In this attack, the attacker floods the network with broadcast messages, sends fake requests, or consumes network resources to disrupt the network's functionality.

2.1.2 Computation and communication overheads

In VANET applications, it is vital to consider the computation and communication overheads as they significantly impact the overall system performance. The term "computation overhead" pertains to the computational power and processing requirements necessary to execute intricate algorithms and protocols within the VANET environment [5]. This overhead is influenced by the complexity of the algorithms used for tasks such as route planning, data fusion, collision avoidance, and other intelligent decision-making processes. On the other hand, the term "communication overhead" refers to utilising channel bandwidth and network resources that are essential for exchanging information among vehicles and infrastructure components [31]. In VANETs, vehicles are limited in processing power and memory, and running complex algorithms can drain their battery quickly. This can be challenging for safety-critical applications, such as collision avoidance, which require real-time processing and low latency. As the number of vehicles on the network increases, computation and communication overheads increase, leading to congestion and message delivery delays. In the realm of authentication, ensuring reliability necessitates finding the ideal trade-off between minimal computational complexity and communication costs, thereby guaranteeing optimal network scalability.

Moreover, this balance enables the implementation of resource-efficient algorithms and communication protocols tailored to the specific constraints of vehicle environments. Optimal management of computation and communication overheads not only enhances system performance but also extends the operational capabilities of the vehicular communication network. Table 2.1 categorises the associated overheads necessary for transmitting and verifying a single authentication request in VANETs based on low, medium, and high categories [32]. The low, medium, and high categories for communication overhead are 1 to 50 *bytes*, 51 to 100 *bytes*, and 101 to 140 *bytes*, respectively, while those for computation overhead are 1 to 3 *msec*, 3.1 to 6 *msec*, and 6.1 to 10 *msec*, respectively [32].

Table 2.1: Classification of the computation and communication overheads [32]

Evaluation metric	**Classification category**		
	Low	**Medium**	**High**
Communication overhead (*bytes*)	1 : 50	51 : 100	101 : 140
Computation overhead (*msec*)	1 : 3	3.1 : 6	6.1 : 10

2.2 Cryptography-based authentication

The current state-of-the-art of authentication in VANETs has been enriched by numerous contributions from researchers who have employed various cryptographic techniques, including elliptic curve cryptosystem (ECC) [33], bilinear pairing (BP) [34], and hash functions [35], among others. In general, there are two main authentication objectives in VANETs: identity and message authentication. Identity authentication aims to verify the identity of the communicating entities in VANETs, ensuring that only authorised entities can participate in the network. Message authentication, on the other hand, ensures the authenticity and integrity of the messages exchanged between various entities. This section presents a comprehensive comparison of recently published PKI-based, ID-based, and GS-based authentication methods in VANETs.

2.2.1 Public key infrastructure-based authentication

PKI-based authentication is a type of authentication mechanism that relies on a trusted third-party known as the TA or the certificate authority (CA) to issue digital certificates to communicating entities. These digital certificates contain the public key of the entity, which is used by other entities to authenticate the entity's identity. Generally, PKI-based authentication involves the following steps:

1. **Setup**: The entity creates a key pair of private and public keys for authentication.
2. **Certificate issuance**: The entity sends a certificate signing request (CSR) to the CA, which includes the entity's public key and identifying information.
3. **Certificate verification**: The CA verifies the entity's identity and issues a digital certificate containing the entity's public key and identifying information.
4. **Message signing/verification**: The entity uses its private key to sign messages or generate digital signatures, which can be verified by other entities using the entity's public key.

Suppose an entity A wants to authenticate its identity to entity B using PKI-based authentication. Then, A generates a key pair $\{A_{priv}, A_{pub}\}$, where A_{priv} and A_{pub} are the private and public keys, respectively.

1. A sends a CSR to the CA, which includes its public key and identifying information: $CSR_A = \langle A_{pub}, ID_A \rangle$.

2. The CA verifies the identity of A and issues a digital certificate containing the entity's public key and identifying information: $Cert_A = \langle CSR_A, \sigma_{CA} \rangle$, where the CA's signature on the CSR is $\sigma_{CA} = Sign_{CA_{priv}}(CSR_A)$ and CA_{priv} is the CA's private key.

3. Next, A signs the packet payload $(m \| Cert_A)$ using its private key: $\sigma_A = Sign_{A_{priv}}(m \| Cert_A)$ and sends the tuple $\langle m, Cert_A, \sigma_A \rangle$ to B.

4. Finally, B checks $\sigma_{CA} \in Cert_A$ and verifies the received signature (σ_A) using A's public key: $Verify_{A_{pub}}(\sigma_A)$.

PKI-based authentication is a widely used technique that involves preloading a substantial number of anonymous certificates ($\sim 43,800$) and corresponding private keys [5]. These certificates are signed by the CA and do not contain any personally identifiable information. In this way, users remain anonymous. To ensure long-term security and privacy, it is necessary to preload a sufficient number of certificates onto each vehicle's OBU, typically enough to last for a certain period. These certificates can be updated during the annual registration process. When a safety-related message needs to be signed, an anonymous certificate and its associated private key are randomly selected. The private key is used to generate the signature, while the public key attached to the certificate is used for verification by the recipient. Only the CA has access to information linking the real identities of vehicles to their anonymous certificates. This mechanism allows the CA to trace misbehaviour and identify users if necessary. A selective list of articles in Fig. 2.1 is provided to offer a comprehensive comparison of various methodologies and their limitations in PKI-based authentication. These articles offer a detailed overview of the various approaches to PKI-based authentication and their respective advantages and limitations.

Revocation is a primary limitation of PKI-based authentication in vehicular networks, requiring the management of many certificates in the certificate revocation list (CRL). In practical terms, revoking a single vehicle requires adding all of its issued certificates to the CRL. With more revoked vehicles, the CRL size increases. This adversely affects the signature verification as the recipient checks the CRL for each received signature, posing a challenge. Hence, careful consideration of the CRL size and verification time is crucial for efficient and secure management of revoked certificates. In [36], Raya et al. proposed a modified PKI-based approach to distribute thousands of pseudonyms with corresponding private keys to vehicles. In this scheme, the sender of a message selects a pseudonym, and the corresponding private key is then used to sign messages. The receiver can verify the authentication by using the corresponding certificate.

In high-speed dynamic conditions, centralizing certification services on the servers could compromise access availability. To address this problem, the research conducted by Oulhaci et al. [37] discusses the design and implementation of a distributed certification system architecture that centralizes the certification services on the region certification authority (RCA) instead

of the CA. The authors emphasised the need for security in VANETs and proposed a distributed approach that allows for effective management of public key certificates (PKCs) using the RCA and sub-ordinate RSUs to issue and sign PKCs to the corresponding vehicles in the same region while improving the resistance to attacks with compromised RSUs. Wang et al. [38] proposed the local identity-based anonymous message authentication protocol (LIAP) for VANETs. The LIAP scheme uses a hybrid digital signature approach, where registered vehicles are issued long-term certificates for mutual authentication between RSUs and vehicles in the same geographic area. To verify the certificates, the validity of the vehicle's certificate is checked against the vehicle's CRL, while the RSU's certificate is checked against the RSU's CRL. After mutual authentication, the RSU issues a local master key, which is transmitted encrypted to the corresponding vehicle. By using the received local master key, the vehicle then generates its anonymous IDs and private keys, which are used for signing messages. To ensure secure communication, the local master key is periodically updated. When communicating with adjacent vehicles in another region with a different RSU, each signature must include the local public key of the regional RSU. Vehicles, in turn, identify the source region of messages by checking the received local public key and then verify the signatures based on bilinear pairing and map-to-point ($M \rightarrow P$) hash function.

In an attempt to reduce the high computational cost of checking the CRL, Wang et al. [39] proposed a hybrid authentication scheme that combines PKI-based with anonymous ID-based authentication. After registration with the CA, each vehicle is issued a unique long-term certificate. To obtain an anonymous ID, the vehicle sends a message request to the corresponding RSU, which checks the validity of the vehicle's certificate and the freshness of the request at a certain timestamp. If the request is valid, the RSU sends the vehicle's request to the CA, which issues the corresponding pseudo ID and sends it to the vehicle via the RSU as an encrypted message using the vehicle's public key. The vehicle decrypts the message and obtains its anonymous ID, which is subsequently utilised during the message signing phase.

This part presents an overview of prevalent blockchain-based authentication techniques to address some limitations of PKI-based authentication in VANETs. Lu et al. [40] employed the blockchain to design a proof of presence and absence of certificate issuance and revocation, respectively, offering conditional identity anonymity. In their scheme, a reputation score is sent with each transmission, indicating the degree of trustworthiness of the sender. Despite that, this solution cannot support unlinkability since the reputation scores are updated gradually. Thus, adversaries can trace broadcasted messages to build location-tracking attacks. Lu et al. [41] combined the Merkle Patricia Tree with the blockchain to enable monitoring of the authority's activities, thus promoting transparency. However, the process of generating anonymous certificates requires frequent interactions between vehicles and the CA. Lin et al. [42] integrated the blockchain "Ethereum" technology into the PKI-based approach to present a certificate distribution and revocation mechanism. In transactions, the CA updates the blockchain with users'

public key certificates' blocks, allowing network terminals to securely verify the received signatures via transactions' addresses as proof of activities.

2.2.2 Identity-based authentication

In ID-based authentication, the user's identity information is used to derive the public key, while the private key is computed and distributed by the key generation center (i.e., TA) based on the given identity information [9]. By doing so, the receiver verifies messages using the sender's public key while signing it using its private key. Generally, ID-based authentication involves the following steps:

1. **Setup**: This step involves generating the system public parameters (PP), the master key (msk), and its associated public key (mpk), so that $Setup \rightarrow \langle PP, msk, mpk \rangle$, where the master secret key (msk) is kept secret.

2. **Key Extraction**: Given an identity (ID) for a specific terminal, the TA extracts the ID's relevant secret key (x) based on $\langle PP, msk, mpk \rangle$, so that $Key_{Extract}(PP, msk, mpk) \rightarrow x$.

3. **Signing**: In this step, the communicating terminal computes the signature (σ) based on $\langle PP, mpk, x \rangle$, so that $Sign(PP, mpk, x) \rightarrow \sigma$.

4. **Verifying**: In this step, the recipient makes the decision on the received signature $Dec = \{accepted, rejected\}$ based on $\langle PP, mpk, ID, \sigma \rangle$, so that $Verify(PP, mpk, ID, \sigma) \rightarrow Dec$.

However, such a scheme suffers from high computation and communication overheads of the large-scale mathematical cryptographic operations executed at the protocol stack's upper layers (link and application layers) that cannot support high scalability and low latency. Scalable networks can add extra terminals without degradation in performance, which is the main objective of many studies [43–61]. Liu et al. [43] proposed the first proxy-based authentication scheme in which proxy vehicles employ their computation availabilities to verify signatures in favour of the RSUs and broadcast their verification results. In fact, this work is limited to V2I communication without considering the scenario of V2V communication. In [44], Asaar et al. revealed that the scheme presented in [43] is vulnerable to impersonation and modification attacks, then presented a modified proxy-based scheme, offering superior computational performance. In this scheme, the n number of received signatures are distributed between $\lceil \frac{n}{d} \rceil$ proxy vehicles for the signature verification process, where $d \simeq 0.1n$. However, the improved scheme preloads the TA's master key into vehicles' tamper-proof devices (TPDs), which is insecure due to the high vulnerability to side-channel attacks for imperfect TPDs. Resisting this type of attack, Bayat et al. [45] make use of the secure communication link between the TA and RSUs to store a dynamically updated master key into the RSUs' TPDs. Based on bilinear pairing properties and $M \rightarrow P$ hashing function, they developed an ID-based scheme that supports batch verification. However,

its significant computation complexity motivated Al-shareeda et al. [46] to design a free pairing conditional privacy-preserving authentication scheme, employing the online mode for updating TPDs' secret parameters to avoid potential side-channel attacks. However, the communication cost remains high. Lo et al. [47] proposed a solution to address the high computational overhead of bilinear pairing operations by utilising the computational Diffie-Hellman problem of the ECC for singular verification. Batch verification is another way of identifying a set of received signatures simultaneously. In a lightweight ID-based solution, Wei et al. [48] employed the factorization problem of the Rivest-Shamir-Adleman (RSA) cryptosystem for identity verification.

What's more, a recent study by Zhang et al. [49] demonstrated that the proposed scheme in [48] is vulnerable to the common modulus attack, which can expose the vehicles' secret parameters. Mitigating the computation load on the vehicles' side, reference [50] suggested a technique based on edge computing where the RSUs verify the received messages from adjacent vehicles and broadcast their verification results to surrounding vehicles. In [51], Limbasiya et al. demonstrated that [50] had security weaknesses related to impersonation attacks, then developed a message authentication approach based on symmetric key cryptography. In reference [52], Lyu et al. employed the timed efficient stream loss-tolerant authentication (TESLA) method along with the elliptic curve-based digital signatures to design a scheme that forecasts the vehicle's future position for immediate message authentication. Despite this, the high communication cost associated with the Merkle Hash Tree's added leaf values continues to pose a challenging issue. In order to reduce the cost of communication and maintain privacy, Zhong et al. [53] implemented a certificateless aggregation signature scheme that reduces the signature size. However, the authors neglected to consider V2V applications, which is an important aspect given that vehicles have a lower processing power than RSUs. In [54], Cui et al. developed an ECC-based content-sharing scheme tailored for fifth-generation (5G)-enabled vehicular networks. The authors' approach enables vehicles with content downloading requests to efficiently filter their nearby vehicles to select competent and suitable proxy vehicles. These selected proxy vehicles are then requested to provide content services.

Many studies have been presented to the research community to support the security and privacy requirements of VANETs. In [55–58], the authors proposed conditional privacy-preserving authentication (CPPA) schemes in which signatures are generated and verified using ECC-based scalar multiplication and addition operations. According to [56], a pseudo-ID-based scheme is proposed in which pseudo-identities are exchanged between terminals to offer conditional privacy. In [59–61], the authors proposed certificate-less authentication schemes to reduce authentication overheads and promote privacy. By adopting these schemes, vehicles are not required to store any certificates for authentication, and the TA is also relieved of the need to retrieve the real identity of malicious vehicles from certificates.

2.2.3 Group signature-based authentication

In GS-based authentication, the group is made up of a manager and members. Each group member (V_i) signs the message (m) on behalf of the group to generate the signature (σ) using V_i's secret key (gsk_i), offering privacy preservation. While the recipient verifies σ using the group public key (gpk). In general, GS-based authentication comprises four steps [62]:

1. **Key generation** Key_{gen}: This phase is executed to generate the essential secret and private keys, s.t. $Key_{gen} \rightarrow \{gsk_i, gpk, gmsk\}$ and other parameters, where $gmsk$ is the group manager's secret key.

2. **Message signing** $Sign(m, gsk_i, gpk)$: This phase is run by V_i to sign m using gsk_i related to gpk, s.t. $Sign(gsk_i, gpk, m) \rightarrow \sigma$.

3. **Message verification** $Verify(\sigma, gpk)$: The recipient checks the validity of σ using gpk without disclosing V_i's real identity (RID_i), s.t. $Verify(\sigma, gpk) \rightarrow m$.

4. **Traceability** $Open(\sigma, gpk, gmsk)$: The group manager can reveal RID_i using $gmsk$, s.t. $Open(\sigma, gpk, gmsk) \rightarrow RID_i$ in case of misbehaving.

However, a major limitation of this scheme is the requirement for the group key to be updated and distributed by the TA for each vehicle getting in/out from the group region which makes such a scheme hard to support forward and backward secrecy, especially in the case of high-speed group members. In [63], RSUs are assigned as group managers to improve the communication and computation overheads. However, compromised RSU makes vehicles' private information vulnerable to exposure. Zhang et al. [64] proposed a group key distribution algorithm used for GS-based batch verification. This algorithm uses a bivariate polynomial-based mechanism to ensure that only unrevoked vehicles can access the group session key. Reference [65] decreases the insignificant overhead on the TA by dividing the RSUs/domain into leader RSU (L-RSU) and member RSUs (M-RSUs). The L-RSU is responsible for generating group keys and tracing the real identities of misbehaving vehicles with the help of the TA. Increasing the number of M-RSUs/domain enhances the system's performance because the vehicle remains in the same domain for a longer period, decreasing the number of domains/region. Using a region trust authority, reference [66] provides vehicles with efficient authentication services and reduces the computational overhead on the TA and RSUs. Fig. 2.2 summarises the performance limitations associated with different types of cryptography-based authentication in VANETs.

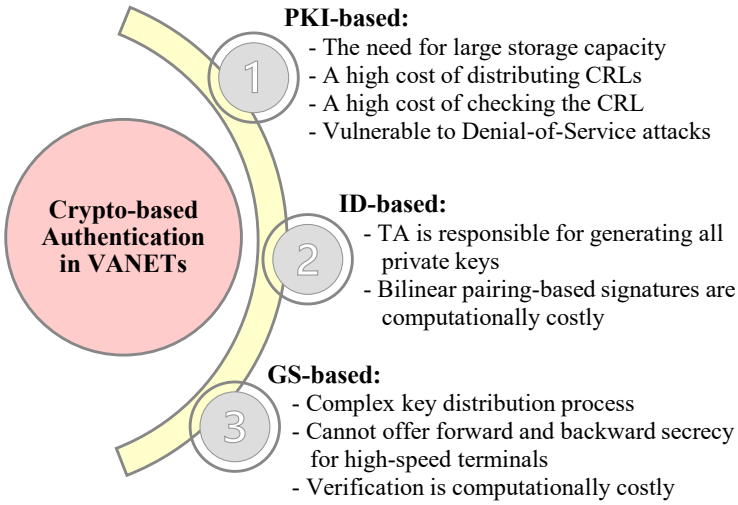


Figure 2.2: Performance limitations of cryptography-based authentication in VANETs.

2.3 PHY-layer authentication

PHY-layer authentication is a security mechanism that aims to establish the authenticity of wireless communication devices by exploiting the unique physical characteristics of their wireless transmissions. This authentication process is performed at the physical layer of the protocol stack and is effective in preventing various types of attacks, including spoofing and impersonation. In this context, existing PHY-layer authentication methods can be classified into three main categories: keyless-based, keyed-based, and tag-based, which are presented in Subsections 2.3.1, 2.3.2, and 2.3.3, respectively (see Fig. 2.1). These subsections provide a review of selected articles that discuss different PHY-layer authentication methods in the context of wireless communication. Finally, the challenges associated with implementing each method in vehicular communication networks are outlined.

2.3.1 Keyless-based PHY-layer-authentication

Keyless-based PHY-layer authentication is a technique used to verify the authenticity of wireless communication devices without relying on any pre-shared keys or secret information. Instead, this approach exploits the inherent properties of the hardware manufacturing process of different wireless devices, such as the carrier frequency offset (CFO) and analogue front-end (AFE) imperfections, and the unique attributes of wireless communication channels, such as the channel state information (CSI), received signal strength (RSS), power delay profile (PDP), channel frequency response (CFR), channel impulse response (CIR) and power spectral density (PSD), to discriminate between different wireless devices. Keyless-based authentication methods typically use statistical analysis to compare the characteristics of the received signal with those of a known reference signal to determine whether the sender is authentic or not. This approach is particularly useful in situations where pre-shared keys are not available or impractical to use. Keyless-based

Table 2.2: Classification of keyless-based PHY-layer-authentication

Ref.	Authors	Year	Authentication feature
Single attribute-based PHY-layer authentication			
[68]	Xiao et al.	2008	CFR
[69]	Tugnait et al.	2013	PSD
[70]	Chin et al.	2015	PDP
[71]	Liu et al.	2016	CIR
[72]	Liu et al.	2017	CIR
[73]	Hamamreh et al.	2018	CSI
[74]	Liao et al.	2019	CSI
[75]	Gao et al.	2019	RSS
[76]	Li et al.	2020	RSS
[77]	Jadoon et al.	2021	RSS
Hardware imperfections-based PHY-layer authentication			
[78]	Zhang et al.	2019	AFE imperfections
[79]	Hou et al.	2014	CFO
Multiple attributes-based PHY-layer authentication			
[80]	Fang et al.	2018	CFO & CIR & RSS
[81]	Wang et al.	2016	RSS & AFE imperfections (in-phase/quadrature (I/Q) imbalance)
[82]	Li et al.	2019	CSI & RSS & CFO
[83]	Ramabadran et al.	2020	CIR & AFE imperfections

authentication can generally be classified into three categories: single attribute-based, multiple attributes-based, and hardware imperfections-based methods. Table 2.2 categorises the literature on keyless-based authentication according to the distinctive discrimination features employed in the selected studies.

Single attribute-based PHY-layer authentication

The channel attributes-based method is founded on the principle of leveraging the short-term spatial and temporal correlations in channel characteristics between two wireless communication devices, which can be specified by a zero-order Bessel function, where the first zero occurred at a $\lambda/2$ distance between the legitimate user and the adversary [67], see Fig. 2.3. Thus allowing for location decorrelation between legitimate and wiretapped channel responses. This approach involves utilising a range of channel features including the CFR [68], PSD [69], PDP [70], CIR [71, 72], CSI [73, 74], and RSS [75–77]. By tracking these features, this method ensures that the received signals, $R_X(t)$ and $R_X(t+\Delta t)$, originate from the same source, where the receiving time interval Δt is less than or equal to the coherence time T_c. This approach is commonly known as the "feature tracking" mechanism.

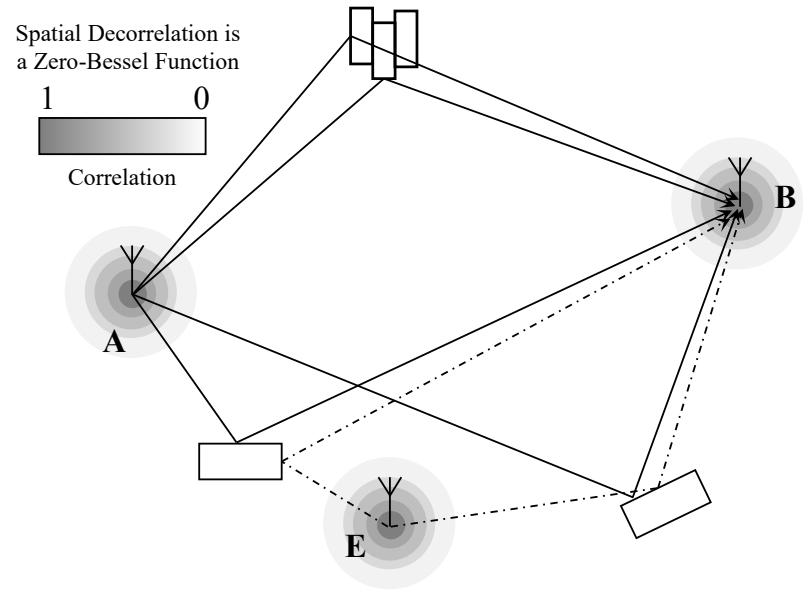


Figure 2.3: Spatial decorrelation representation between legitimate and wiretap channel.

In [68], Xiao et al. proposed an authentication method that enables a receiver (B) to authenticate a legitimate terminal (A) based on the channel frequency response in a time-variant environment. The proposed method involves comparing the estimated channel response $H_{A\to B}(K)$ with previously recorded responses $H_t(K)$. If the channel responses are highly correlated over time, the terminal is considered trustworthy. On the other hand, if the channel responses are not correlated, the terminal is deemed untrustworthy, depending on the threshold value (τ). The decision to trust or not trust the terminal is made based on a binary hypothesis testing problem. Specifically, the null hypothesis is defined as $H_0 : H_t(K) = H_{A\to B}(K)$, while the alternative hypothesis is defined as $H_1 : H_t(K) \neq H_{A\to B}(K)$. Tugnait et al. [69] introduced an authentication technique that leverages the correlation among power spectral density estimates, $S(f)$, obtained from a particular terminal at different time series. The method involves a binary hypothesis testing process represented by $H_0 : S_t(f) = S_{A\to B}(f)$ and $H_1 : S_t(f) \neq S_{A\to B}(f)$. The authors estimated the power spectral density using the Daniell method [84] and optimised the threshold value and the probability of false alarm (P_{fA}) for a generalized likelihood ratio test (GLRT), where P_{fA} is the probability that a third party is authenticated as an authorised terminal. Chin et al. [70] have utilised the concept of employing the power delay profile as a proficient PHY-layer authentication mechanism to differentiate between two consecutive bursts $\{m-1, m\}$ in mobile orthogonal frequency-division multiplexing (OFDM) system. The study formulates the PDP $\{\widehat{P}_{m-1}, \widehat{P}_m\}$ of different bursts based on the cyclic prefix (CP) length and the number of subcarriers (N). Fig. 2.4 depicts the flowchart of the method proposed in [70], which employs the statistical parameter S to make a trustworthy decision. The parameter S is determined by calculating $\{\widehat{P}_{m-1}, \widehat{P}_m\}$, and then compared to τ. The results indicated that as the value of τ increases, the false alarm probability decreases and vice versa.

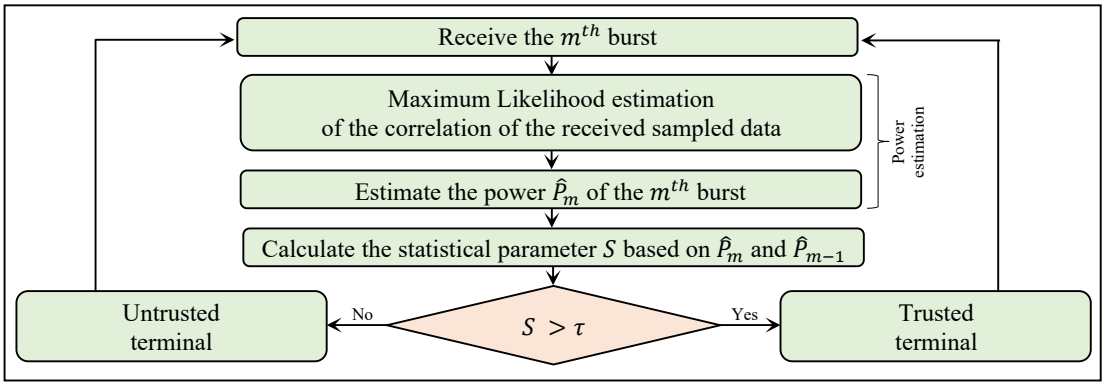


Figure 2.4: Flowchart of the power delay profile-based implementation of the PHY-layer authentication method in [70].

For improved performance, Liu et al. [71] presented a novel two-dimensional (2D) quantisation method $\{Q_h, Q_d\}$ that employs the channel amplitude ($\hat{h}_l(n)$) and path delay ($\hat{d}_l(n)$) estimates of the l^{th} multipath component to distinguish between trusted and untrusted terminals based on the spatial and temporal variations in transmission across different time intervals $\{n, n+1\}$. Fig. 2.5 illustrates the changes in the $\hat{h}_l(n)$ and $\hat{d}_l(n)$ components across various time slots. The decision rule is determined based on the following hypothesis:

$$\begin{aligned} H_0 &: S = S_h + S_d \leq \tau, \\ H_1 &: S = S_h + S_d > \tau \end{aligned} \tag{2.1}$$

where S_h and S_d are given by

$$\begin{aligned} S_h &= \sum_{l=0}^{L-1} Q_h \left[\left| \hat{h}_{X,l}(n+1) - \hat{h}_{A,l}(n) \right|^2 \right], \\ S_d &= \sum_{l=1}^{L-1} Q_d \left[\left| \hat{d}_{X,l}(n+1) - \hat{d}_{A,l}(n) \right| \right] \end{aligned} \tag{2.2}$$

where L is the total number of channel paths, and $\{\hat{h}_X, \hat{d}_X\}$ are the channel amplitude and path delay estimates from a specific terminal (X) at time $(n+1)$ while $\{\hat{h}_A, \hat{d}_A\}$ are that of a pre-authenticated terminal (A) at time n. The decision rules of S_h and S_d depend on the threshold values δ_h and δ_d, respectively. The null hypothesis H_0 means $x = A$ while H_1 means $x \neq A$.

In [72], Liu et al. proposed an amplify-and-forward cooperative relaying technique for end-to-end (E2E) transmission aimed at enhancing the wireless communication range, as shown in Fig. 2.6. The proposed method optimises the selection of the best cooperative relay (R_i) from a set of M relays, by maximising the SNR for E2E communication ($Alice \rightarrow R_i \rightarrow Bob$) through each of the relays $\forall R_i \in \{R_1, \ldots, R_M\}$. Furthermore, the authors presented a PHY-layer authentication mechanism that utilises the short-term channel correlation between consecutive E2E transmissions that pass through the selected best relay.

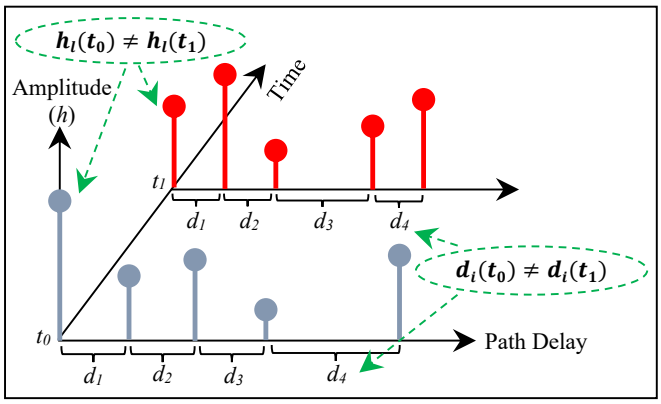


Figure 2.5: Channel variations (amplitude & phase) over time [71].

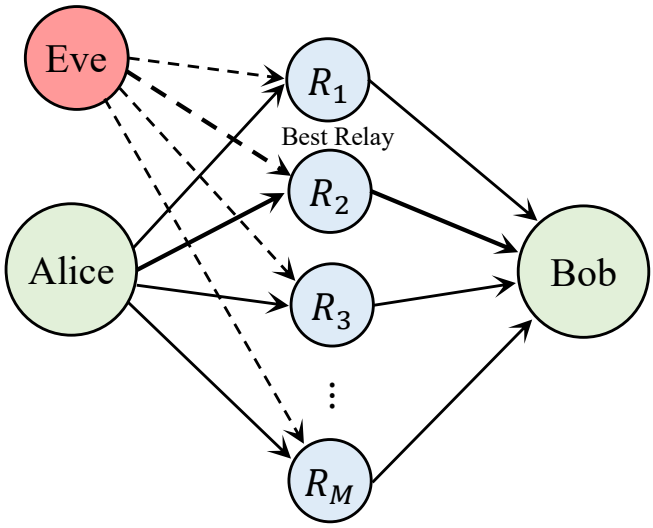
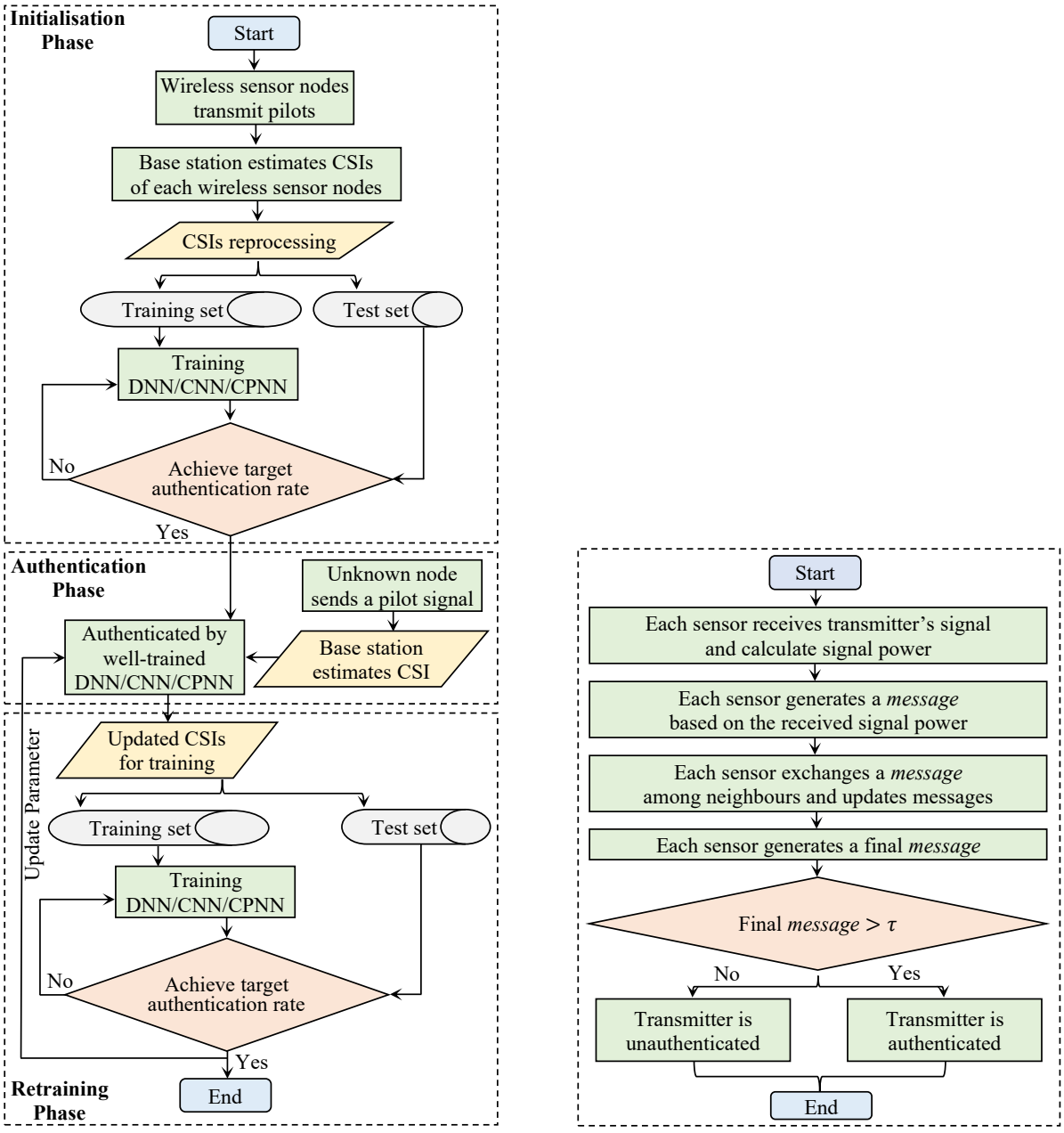


Figure 2.6: Relay selection for cooperative relaying using amplify and forward method [72].

In the context where a source node (Alice) communicates with a legitimate user (Bob) in the presence of passive eavesdropping (Eve), a technique known as cyclic redundancy check (CRC) is employed to encode information data bits. Upon transmission of the encoded data to Bob, the latter carries out a CRC decoding process and sends a retransmission request indicating whether the decoding was successful or not. Alice may repeat the transmission process until Bob exhausts the maximum number of retransmissions (L) or receives the transmitted packet successfully. This technique is commonly referred to as automatic repeat request (ARQ). To enhance the security of this method, Hamamreh et al. presented a technique in [73] that combines the artificial noise (AN) cancellation process with the use of maximal ratio combining (MRC) at the receiver terminal. The MRC process merges two consecutive retransmitted data packets (i.e., $L = 2$) to eliminate the effect of the AN on the intended receiver's side. The security strength of this approach relies on the complexity involved in Eve's ability to eliminate the added AN. This is because the AN is produced at the transmitter's side based on the channel reciprocity between legitimate terminals (i.e., $Alice \leftrightarrow Bob$).

In [74], Liao et al. proposed a deep learning (DL)-based authentication method for industrial wireless sensor nodes that adopts three alternative algorithms, deep neural network (DNN), convolution neural network (CNN), and improved convolution pre-processing neural network (CPNN). Fig. 2.7(a) presents the flowchart of this technique comprising three phases, initialisation, authentication, and retraining. During the initialisation phase, the kernels are trained using CSI data obtained from different nodes. In the subsequent authentication phase, the upcoming CSI is verified, and the dataset is retrained for the next iteration.



(a) Flowchart of the deep learning-based authentication.

(b) Flowchart of the cooperative PHY-layer authentication.

Figure 2.7: Flowchart of the PHY-layer authentication methods in [74, 75].

The wireless sensor network (WSN) is vulnerable to intelligent attackers who can imitate legitimate channel information through beamforming techniques. These attackers leverage machine learning, such as Q-learning, to select attack actions based on the communication channel and maximise long-term cumulative rewards. To mitigate such attacks on a network comprising M nodes, $\{n_1, \cdots, n_M\}$, Gao et al. [75] proposed a cooperative PHY-layer authentication approach. In this method, an intelligent attacker attempts to mimic the channel and transmit a signal to a sensor $n_i \in \{n_1, \cdots, n_M\}$. Each sensor in the network measures the power of the received signal $\psi(y_i)$, for $i = \{1, \cdots, M\}$. In the presence of an attack, both the attacker and the sink node coexist, resulting in a larger received power than usual. Notably, the higher the power of the deceptive signal, the greater the risk of the attack being detected. Therefore, there exists a tradeoff between transmit power and the detection probability of local observation. Finally, each sensor transmits messages to its neighbours based on $\psi(y_i)$, updates the final message (belief), and tests whether it exceeds the threshold value (belief threshold). The flowchart illustrating this method is presented in Fig. 2.7(b).

Li et al. [76] introduced a PHY-layer authentication framework that employs an area-based approach for detecting spoofing attacks. The proposed framework classifies the area surrounding the destination terminal, represented by the symbol ★ in Fig. 2.8. The legitimate area is defined as the region bounded by distances $[d_i, d_0]$, where the legitimate terminal (•) is positioned far from the destination point with distance d_{LR}, and the spoofing attacker (■) is located at a distance d_{AR}. This method involves the definition of a silent probability. Specifically, when the spoofer is located far away from ★ or the spoofing power is small, the attacker is more likely to keep silent. Hence, the surrounding area of ★ is categorised into three areas:

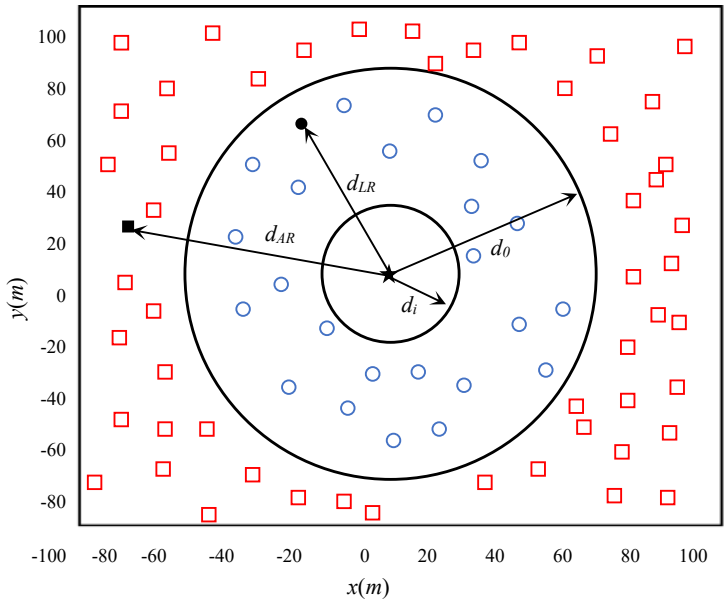


Figure 2.8: Area authentication model [76].

1. The clear area, where the spoofer has no opportunity to construct an attack due to the high probability of detection (i.e., the silent probability is larger than a threshold ε_1; e.g. 90%).

2. The danger area consists of locations where the spoofer can achieve a relatively higher successful spoofing probability larger than a threshold ε_A (e.g., 10%).

3. The warning area, where a legitimate terminal should avoid being located, as it can be classified as a spoofer (The area where users are at a significantly high risk of being identified as spoofers).

The boundaries of each region are allocated based on the RSS while the threshold values are chosen to meet practical specifications and demands. The probability of a successful attack increases as the legitimate user approaches the inner and outer boundaries. This framework offers an effective solution for detecting spoofing attacks and enhances network security.

The channel randomness and the short-term reciprocal features can be exploited to extract a high entropy secret key. By exchanging probing packets and quantising the resulting channel estimates, two communicating terminals can establish a symmetric shared key. However, the imperfect channel reciprocity can introduce discrepancies in the extracted key, requiring subsequent information reconciliation and privacy amplification stages. A comprehensive discussion of this approach is presented in Section 2.5. Jadoon et al. [77] proposed a Hopper-Blum-based PHY-layer (HB-PL) authentication scheme that excludes the information reconciliation and privacy amplification stages. The extracted information is used as an input secret key for the HB protocol used for authentication. The authors estimated the percentage of successful authentication for varying numbers of exchange probing packets and compared the results with the traditional cascade scheme using MATLAB simulations. The simulations showed that the HB-PL scheme achieves a 95% authentication rate with 55 exchange probing packets, while the cascade scheme required 65 exchange probing packets to achieve a 90% authentication rate. The proposed HB-PL scheme offers a more efficient and simplified approach for PHY-layer authentication in wireless communication systems.

Hardware imperfections-based PHY-layer authentication

Hardware imperfections-based authentication is an innovative technique for enhancing wireless communication security. This approach leverages the intrinsic imperfections inherited in the hardware components of wireless devices for authentication purposes. These imperfections can be a result of manufacturing variations, such as AFE imperfections [78] and CFO [79]. The AFE imperfections encompass deviations in amplification, filtering, or other circuitry characteristics, contributing to a unique signature for individual devices [78]. Concurrently, CFO imperfections result in discrepancies in the oscillation frequencies of devices, further enriching the distinguishing features [79]. These subtle diversities, typically unnoticed during regular device operation, serve as an underlying basis for robust authentication frameworks. By harnessing these inherent

imperfections, authentication protocols can verify the legitimacy of devices within a network, mitigating the risk of unauthorised access or malicious intrusions. Furthermore, this approach not only fortifies security but also exhibits potential for cost-effective implementation, leveraging existing hardware attributes without necessitating additional overhead or resource-intensive procedures. This method involves the measurement and characterisation of these imperfections using signal processing techniques and machine learning algorithms to establish unique hardware signatures, which can then be used to authenticate legitimate users and devices. In the work introduced by Zhang et al. [78], radio frequency fingerprinting (RFF) is employed to differentiate between various terminals in internet-of-things (IoT) devices. The received signal is partitioned into samples, and the primary features can be extracted as follows:

1. *Transient part*: The turn-on transient part of the signal can be identified by the authenticator during frequency synthesis when the frequency synthesizer synchronises with the user's assigned transmission frequency.

2. *Near transient part*: This part includes both the turning on transient and stable segments of the signal.

3. *Preamble part*: The unique hardware features can be extracted from the preamble segment of the signal by calculating its power spectral density and analysing its frequency and phase attributes to generate the RFF.

4. *The entire signal*: The frequency, phase, amplitude, and I/Q samples can be analysed across the entire signal to extract RFF features.

5. *RF burst*: Radio-frequency identification-based systems can use the out-of-band emissions from a sinusoidal carrier outside its intended frequency to obtain a distinct hardware fingerprint.

The feature vector of N devices is represented by $V = \{v_1, v_2, \ldots, v_N\}$. The classifier is trained to obtain the function (ϕ), which represents the feature space of N users, such that $C = \phi(V)$, where $C = \{c_1, c_2, \ldots, c_N\}$ is the class space. The decision rule is taken based on the scoring function $s : V \times C$. If the scoring value of (v_i, c_i) is greater than τ, then the person is considered an authorised terminal. Otherwise, the terminal is unauthorised.

The CFO is another distinctive PHY-layer attribute, in which, RF oscillators of each pair of communication terminals can be relatively biased to the central oscillating frequency in addition to the Doppler shift induced by the mobility terminals. In a study conducted by Hou et al. [79], a Kalman filter is employed to predict the current CFO value using previous CFO variations and a training sequence. The decision rule involves a binary hypothesis testing problem, where the current CFO estimate is compared to the predicted Kalman CFO.

Multiple attributes-based PHY-layer authentication

For improved authentication performance, a combination of multiple PHY-layer features can be used for discriminating between different geographically located terminals. Fang et al. [80] proposed an authentication technique that leverages machine learning and N number of multiple channel attributes. This technique utilises different attributes, such as the RSS, CFO, and CSI, among others. The normalisation process for each channel attribute is done using the maximum and minimum range of each attribute's observations. A training set of observations is utilised to train the authentication step. The Gaussian kernel function $f(.)$ is employed to model the authentication process, as illustrated in Fig. 2.9. The authors measure the mean square error (MSE) versus the iteration index for different combinations of attributes {CFO, CIR, RSS, CFO & CIR, CFO & RSS, CIR & RSS, CFO & CIR & RSS}. The results show that the best authentication performance is achieved using all three attributes (CFO & CIR & RSS). Moreover, the evaluation of the MSE with respect to different numbers of attributes $(N = 1,2,3)$ indicates that increasing the number of attributes leads to a decrease in the error rate. The security strength of this technique lies in the difficulty of an adversary to forge the correct observations of all three attributes, which enhances the security of the authentication process.

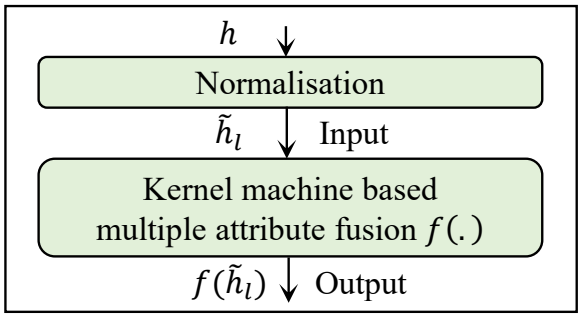


Figure 2.9: Kernel machine-based multiple PHY-layer channel attributes [80].

Wang et al. [81] proposed a multi-attributes multi-observation (MAMO) authentication mechanism that combines the RSS estimation and I/Q amplitude and phase shift imbalance (IQI) for both stationary and mobile device scenarios. The proposed method employs three antennas to obtain multiple observations, which enhances the reliability of the authentication mechanism by estimating the MRC for multiple channels. To evaluate the performance of the authentication techniques, the receiver operating characteristic (ROC) curves (probability of detection P_d versus probability of false alarm P_{fa}) are evaluated using MATLAB simulations for three methods, namely, IQI&RSS-MRC, IQI-MRC, and IQI-Non-MRC. The results indicated that the MAMO technique achieves superior performance, 50.48% and 9.28%, compared to the IQI-Non-MRC and IQI-MRC, respectively.

Li et al. [82] developed a software-defined radio (SDR) platform-based PHY-layer authentication mechanism for 802.11 a/g networks. The channel characteristics are extracted from the

802.11 frames using the following methods:

1. The CSI is calculated from the long training sequence (LTS). A vector of 52 samples is generated to estimate the CSI of the current frame, and changes are estimated by comparing the pre-stored and received values of the LTS.

2. The RSS is obtained from the power values of the short training sequence (STS) as it is considered to be fixed across different frames.

3. The frequency offset is measured from the last 64 samples in the STS, and the previous samples are discarded to avoid the effect of automatic gain control (AGC) adjustment.

4. The timestamp is estimated using a 64-bit counter that is incremented every clock cycle.

The TickSEC platform, which includes an embedded MicroBlaze processor is used to evaluate the detection rates of Wi-Fi devices using different machine learning models, such as linear regression analysis, decision tree, and support vector machines. The detection rates ranged from 96.8% to 98.48%.

Ramabadran et al. [83] propose a scheme to generate a shared key between two nodes $(1,2)$ for phase encryption of modulated signals, taking into account circuit impairments. The key generation process is based on the CIR and is explained as follows.

1. **Step 1**: Node-1 sends a predefined probe signal $y(n)$ to Node-2, and the received signal $r_2(n)$ combines the hardware frequency response $h_{11}(n)$ of Node-1 with the CIR $h_{12}(n)$. The received signal in the frequency domain is given by

$$R_2(j\omega) = Y(j\omega)H_{11}(j\omega)H_{12}(j\omega) \tag{2.3}$$

   The obtained $R_2(j\omega)$ is then multiplied by its hardware frequency response $h_{22}(n)$, resulting in the final estimate $E_2(j\omega)$, denoted by

$$E_2(j\omega) = \frac{R_2(j\omega)}{Y(j\omega)}H_{22}(j\omega) = H_{11}(j\omega)H_{12}(j\omega)H_{22}(j\omega) \tag{2.4}$$

   This process is repeated to estimate $E_1(j\omega)$ of Node-1 within T_c, such that $H_{12}(j\omega) \simeq H_{21}(j\omega)$. Despite the added noise and hardware performance variations, the estimated $E_1(j\omega)$ is highly correlated with $E_2(j\omega)$, which is quantised to extract the shared key.

2. **Step 2**: The transmission of modulated signals involves the encryption of the phase component at the transmitter side and its subsequent decryption at the receiver side. The encryption and decryption processes rely on the use of a non-linear function, which is predetermined and applied on both sides. The non-linear function is a critical component

to generate the encryption and decryption phase values, given by

$$y = x^3 + f_1 x + f_2 \bmod f_3 \tag{2.5}$$

where the prime coefficients (f_1, f_2, f_3) of the non-linear function are selected randomly from the extracted shared key, which is periodically updated to avoid brute force attacks. Finally, the encryption phase values are added to the modulated signals' phases.

The extracted key is tested for symmetry. In addition, the proposed scheme is evaluated through experiments on an SDR testbed. The results demonstrate the effectiveness of the proposed scheme in detecting unauthorised devices.

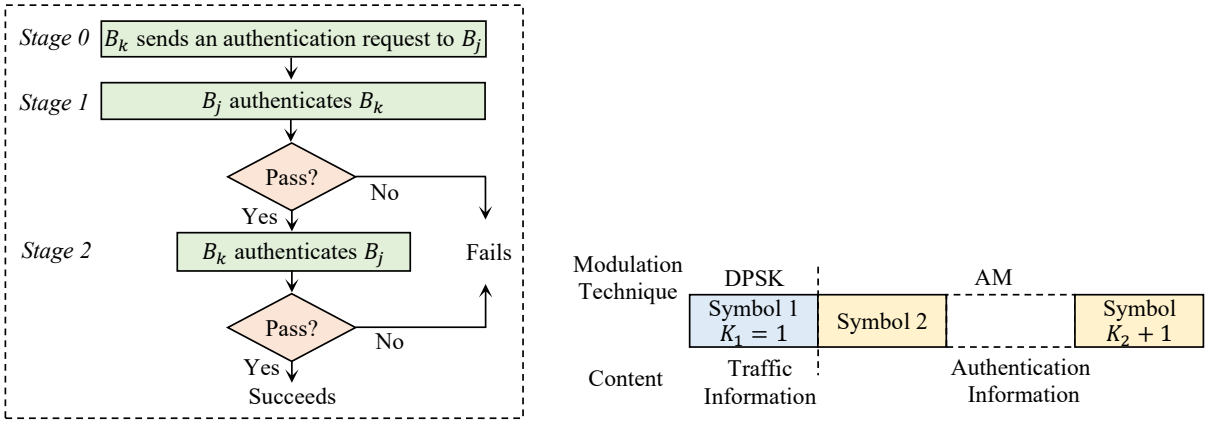
2.3.2 Keyed-based PHY-layer-authentication

This authentication method works by using a secret key that is shared between the communicating wireless devices and the physical characteristics of the wireless channel. The key is used to authenticate the identity of the wireless device during the initial connection setup. Shan et al. [85] proposed a physical layer challenge-response authentication mechanism (PHY-CRAM) for wireless networks, which is a one-way or mutual authentication scheme for an OFDM system with N subcarriers. The scheme utilises either stage 1 alone or stages 1 and 2 in combination, as illustrated in the flowchart presented in Fig. 2.10(a). The frame structure consists of k_1 and k_2 symbols, as shown in Fig. 2.10(b). The first k_1 symbols contain traffic information, which is modulated using differential phase shift keying (DPSK), while the following k_2 symbols are used for authentication based on the symmetric key pair $\{X_j,\ X_k\}$ and the random values D_n. The authentication process is carried out between two legitimate terminals, B_j and B_k, and can be summarised as follows.

1. **Stage 0**: To maintain the secrecy of the CSI, B_k sends an authentication request to B_j using random amplitude subcarriers.

2. **Stage 1**: B_j generates random values D_n within the range of $[k_3, k_4]$, where $0 < k_3 < 1 < k_4$. These random values are used to amplitude modulate the last k_2 symbols, which are then sent to B_k. Based on the received signal $R^{(1)} = D_n H_{jk} + W_n^{(1)}$, B_k calculates and sends $T^{(1)} = \frac{\mathscr{M}(X_k)}{D_n H_{jk} + W_n^{(1)}}$ to B_j, where $\mathscr{M}(.)$ represents the mapping function, H_{jk} is the wireless channel response from B_j to B_k, and $W_n^{(1)}$ is the added noise at the receiver side.

3. **Stage 2**: B_j receives $R^{(2)} = \frac{\mathscr{M}(X_k) H_{kj}}{D_n H_{jk} + W_n^{(1)}} + W_n^{(2)}$, where H_{kj} represents the wireless channel response from B_k to B_j. Due to channel reciprocity ($H_{kj} \approx H_{jk}$) within T_c and neglecting the noise, it follows that $R^{(2)} = \frac{\mathscr{M}(X_k)}{D_n}$. Finally, B_j performs binary hypothesis testing based on the pre-agreed shared key X_k and D_n to authenticate B_k.



(a) Flowchart of the mutual authentication. (b) Frame structure of the authentication process.

Figure 2.10: Flowchart and frame structure of the keyed-based authentication method in [85].

This mechanism has been implemented and evaluated using a field-programmable gate array (FPGA) in both urban and rural channels. The ROC has been measured for various values of SNRs including 10, 15, and 20 dB, and key lengths of 40 and 60 bits. The evaluations have been carried out for line-of-sight (LoS) scenarios at different distances of 3, 6, and 28 meters, as well as for non-line-of-sight (NLoS) scenarios at a distance of 6 meters. The results show that increasing the key length and SNR leads to better ROC values. The worst performance is observed at an SNR of 10 dB for the NLoS scenario and LoS with a distance of 28 meters.

In [86], a novel PHY-layer challenge-response authentication scheme (PHY-PCRAS) is proposed for a multi-carrier communication system with N subcarriers. The scheme utilises the short-term reciprocal features of the channel phase response to generate the response signal corresponding to the received challenge. This scheme facilitates mutual authentication between two parties, Alice and Bob, who have previously agreed upon a shared key denoted by (k_A, k_B). For a one-way authentication, Alice sends an initial challenge-modulated sinusoidal signal to Bob. Subsequently, Bob computes the phase difference estimate of the i^{th} subcarrier denoted by $\Delta\theta_{i1} = \theta_i - \theta_1$, $\forall i \in \{1, \cdots, N\}$, where θ_i represents the estimated channel phase response of the i^{th} subcarrier. Then, Bob replies to Alice's challenge by encapsulating the mapped key k_B into the phase of the response signal, masked by the computed $\Delta\theta_{i1}$. Leveraging the channel reciprocity, the response signal is equalised at the side of Alice. Subsequently, Alice verifies the received signal by performing binary hypothesis testing, using the shared key k_B as a reference for authentication.

In a similar way, Cheng et al. [87] introduced a secret key extraction mechanism at the physical layer. This mechanism involves extracting a preliminary key, which is subsequently employed for authentication purposes based on the reciprocal features of the channel phase response. The study evaluates the ROC under varying numbers of subcarriers $N = \{16, 32, 64, 128\}$ at a SNR of 5 dB. The results demonstrate a positive correlation between ROC performance and

N, with near-ideal performance observed when $N = 32$ subcarriers and SNR $= 5$ dB. Furthermore, the scheme introduced in [87] demonstrates better performance than PHY-CRAM [85] and similar performance to PHY-PCRAS [86].

2.3.3 Tag-based PHY-layer-authentication

The principle behind tag-based PHY-layer authentication involves superimposing a secret modulated signal into the transmitted signal, which acts as a signal watermark. The study conducted by Ran et al. [88] proposes a method for implementing tag-based PHY-layer authentication by inserting a tag signal (t_i) into the transmitted signal (x_i). The tag is generated using a hash function $g(.)$, which takes into account both the channel gain (H_i) and the message contents (s_i) as inputs, resulting in $t_i = g(s_i,\ H_i)$. The tag is then combined with the transmitted message by padding it, such that $x_i = \rho_s s_i + \rho_t t_i$, where ρ_s and ρ_t represent the allocated energy to the message and tag, respectively. Additionally, $\rho_s^2 + \rho_t^2 = 1$, and $0 < \rho_s, \rho_t < 1$. At the receiver end, the estimated channel gain (H_i') is used to compute the receiver's tag value (t_i'), which is then compared with the transmitted tag using the cost function $f_c(t_i,\ t_i')$ to make a decision. Simulation results showed that increasing the SNR value led to a decrease in the bit error rate (BER) and an increase in the authentication rate, and vice versa. The simulation is conducted with varying SNR values (ranging from 0 to 12 dB) and with 40% of the transmitted signal being tagged. The results also showed that increasing the energy allocation of the tag led to a decrease in the BER.

The tag-based authentication method proposed by Zhang et al. [89] involves embedding an encrypted tag signal into the message signal using an asymmetric encryption scheme. The reference tag signal of each terminal is shared with the access point (Bob) through a secured channel. Public and private keys with low key space are computed for network terminals, after which the reference tag is encrypted using a hash function and Alice's private key to generate the cover tag. The cover tag is then embedded into the 16-QAM modulated signal. In the tag verification process, the estimated tagged signal is decrypted, allowing Bob to authenticate the sender under binary hypothesis testing.

Zhang et al. [90] proposed a novel Gaussian tag-embedded authentication (GTEA) scheme utilising the weighted fractional Fourier transform (WFRFT) to generate the tag signal. The Gaussian distribution of the embedded tag signal provides an additional layer of security as it appears as random noise to unauthorised terminals. The effectiveness of the GTEA scheme is evaluated through simulations where both the message bit error rate (MBER) and tag bit error rate (TBER) are measured across varying SNR values ($0 \rightarrow 40$) dB and signal-to-tag power allocation ratio (STR) values ($0 \rightarrow 40$). Results show that increasing the STR leads to a decrease in MBER and an increase in TBER for a fixed SNR value. Similarly, for a fixed STR, an increase in SNR value leads to a decrease in MBER and TBER, ultimately improving the authentication performance.

2.3.4 Challenges and limitations of PHY-layer authentication

This section discusses performance limitations associated with each type of PHY-layer authentication. The challenges related to single attribute-based authentication can be summarised as follows: Firstly, this technique is hindered by a low probability of detection when there are significant channel variations and low SNRs. As a result, it may not be practical for applications that have limited resources or are required to cover long distances. Secondly, observing the wireless channel attributes of all the relevant terminals within a limited time period T_c is a challenging task, especially for applications that are highly dynamic or densely populated. Finally, initial identity verification of the corresponding terminal is still necessary based on existing cryptographic protocols to identify its legitimacy and extract its unique features. Hardware imperfections-based authentication has a significant weakness as the features extracted from different devices vary slightly, leading to false decision-making. In addition, these features are also characterised by their instabilities due to voltage supply, temperature variations, and electromagnetic interference.

Multiple attributes-based authentication poses several challenges that require careful consideration. Firstly, machine and deep learning-based techniques encounter significant performance limitations due to the inherent complexity of these techniques. Specifically, the training of kernels and neurons requires the use of large datasets, which is not feasible in VANET applications. Secondly, each terminal in the network must be pre-registered to extract its distinctive features for supervised authentication approaches. Keyed-based authentication requires the presence of a pre-agreed shared key between the communicating terminals. Furthermore, the establishment of this shared key requires an initial identity verification of the corresponding terminal, which is accomplished through existing cryptography-based authentication schemes. Tag-based authentication introduces a significant tradeoff between decoding performance and security, particularly under varying signal-to-tag power allocation ratios. Moreover, to ensure the legitimacy of the communicating terminal and agree on a secret tag, initial identity verification is still necessary, which is performed through existing cryptographic protocols.

Table 2.3 presents a concise summary of the problem statement and limitations associated with various types of PHY-layer authentication. In summary, PHY-layer-based authentication cannot provide a comprehensive alternative solution, as initial identity verification of the corresponding terminal is still necessary to authenticate the legitimacy of the terminal and extract its unique features or establish a symmetric shared key using existing cryptographic protocols. However, these methods can serve as an integral component in the broader context of cross-layer authentication, as discussed in the subsequent section.

Table 2.3: Challenges and limitations of PHY-layer authentication

No.	Problem statement	Authentication method	Limitations
[68]	Crypto-based authentication ignores the channel's unique features, which hinders attack detection and key management at the physical layer.	Single attribute-based authentication	1- This technique suffers from a low probability of detection at significant channel variations and small SNRs, making it impractical in resource-constrained and long-range applications. 2- All the corresponding terminals must be extensively observed to capture their wireless channel attributes within T_c, which is not feasible for dynamic and high-density applications. 3- Initial identity verification of the corresponding terminal is still needed based on the existing cryptographic protocols to identify its legitimacy and extract its distinctive features.
[69]	Cryptographic techniques are inadequate in providing a comprehensive defense against malicious attacks, particularly in wireless communication channels.		
[70]	Current security techniques fail to account for the channel temporal variations, which can serve as a valuable security resource and are difficult to replicate.		
[71]	Crypto-based methods incur high computation and communication overheads.		
[72]	Spoofing attacks pose a significant threat and traditional methods are computationally expensive and resource-intensive.		
[73]	Traditional authentication schemes include complex key distribution and management processes, large key lengths, and high computation complexity.		
[74]	Deep neural and convolution neural networks cannot provide low latency and high authentication rates with few hidden layers.		
[75]	Intelligent attackers can mimic the CSI using beamforming. An attacker uses machine learning to maximise long-term reward accumulation by taking an attack action.		
[76]	Existing authentication schemes are an effective security solution, but detection accuracy decreases on high-mobility terminals and massive connections.		
[77]	PHY-layer secret key generation extracts a shared key between nodes. However, the information reconciliation corrects mismatched bits through public channels, posing an immense threat.		
[78]	Traditional authentication methods are not practical for very small, low-cost, and resource-constrained devices. Therefore, the need for a lightweight authentication scheme is high-priority to address these limitations.	Hardware imperfections based authentication	1- This approach has a significant weakness in that features extracted from different devices vary slightly, leading to false decision-making. 2- These features are also characterised by their instabilities due to voltage supply, temperature variations, and electromagnetic interference.
[79]	Cryptographic authentication is processed at the upper layers without configuring the physical layer attributes, which is an effective authentication resource for scalable networks.		
[80]	Multiple attributes-based PHY-layer authentication requires high computation complexity. Additionally, sophisticated adaptive techniques must be used to detect any disclosure within T_c.	Multiple attributes-based authentication	1- The high complexity of machine/deep learning-based schemes constitutes a significant performance limitation due to the need for large data sets for training kernels/neurons, which is not applicable in VANETs. 2- Each terminal in the network must be pre-registered to extract its distinctive features for supervised authentication approaches. 3- Initial identity verification of the corresponding terminal is still needed based on the existing cryptographic protocols to identify its legitimacy.
[81]	Low reliability of channel-based PHY-layer authentication due to signal quality fluctuation. In addition, frequent authentication handovers degrade 5G communication system performance.		
[82]	In Wi-Fi applications, extracting channel features without affecting communication performance, and providing lightweight authentication with low latency are considered challenging.		
[83]	The open nature of wireless networks leaves the nodes open to traffic analysis and interception by eavesdroppers and man-in-the-middle platforms.		
[85]	Conventional authentication requires high computation and storage capacity which is not suitable for resource-constrained applications.	Keyed-based authentication	1- A pre-agreed shared key is still needed between terminals. 2- Initial identity verification of the corresponding terminal is still needed based on the existing cryptographic protocols to identify its legitimacy and establish a symmetric shared key.
[86]	Wireless communication through open networks is vulnerable to spoofing attacks where an attacker impersonates a legitimate party.		
[87]	PHY-layer key extraction suffers from low key generation rates due to diverse extraction steps. In addition, the extracted keys have low entropy because of minimal channel variations.		
[88]	Traditional cryptographic techniques suffer from high computation and complex key management.	Tag-based authentication	1- The tradeoff between decoding performance and security is a non-negligible issue under different signal-to-tag power allocation ratios. 2- Initial identity verification of the corresponding terminal is still needed based on the existing cryptographic protocols to identify its legitimacy and agree on a secret tag.
[89]	Many contributed tag-based PHY-authentication schemes cannot resist impersonation attacks as the computed tag signal depends on the message contents and is easy to forge by an adversary.		
[90]	Lightweight cryptographic schemes are widely used in resource-constrained applications that cannot provide high levels of security against potential attacks.		

2.4 Cross-layer authentication

This method involves an initial mutual authentication between authorised parties using crypto-based authentication methods described in Section 2.2. Subsequently, the re-authentication process is conducted at the physical layer, utilising the authentication techniques outlined in Section 2.3. It is crucial to note that the initial legitimacy verification is a crucial step in extracting secret features or establishing a symmetric key for the PHY-layer re-authentication process.

Wen et al. [16] patented a cross-layer authentication method that uses PKI-based authentication for handshaking and generating a radio frequency fingerprint for re-authentication. Althunibat et al. [91] proposed a different integration method for mobile multiple-input multiple-output (MIMO) systems, whereby PKI-based authentication is applied as an initial authentication step, followed by the feature tracking method for re-authentication. For improved performance, the adaptive Kalman filter is employed by Wang et al. in [92] to predict the upcoming CSI and RSS based on the previous estimations and compare them with current observations in a 2D hypothesis testing problem. Yang et al. [93] introduced another cross-layer approach for mobile communications. In this work, the PHY response is not transmitted in the bit form but is masked by the channel frequency response between the user terminal and the base station using a fault-tolerant hashing technique. However, the time taken to generate the response signal is not evaluated and compared to the minimum coherence time to ensure the short-term channel reciprocity between the communicating terminals. Gope et al. [94] proposed an approach for incorporating the integrated circuits (ICs) physically unclonable function (PUF) into a pseudo-ID-based authentication. Based on the ICs' physical variation (P), the PUF method effectively generates an unpredictable response $R = P(C)$, where C is the input challenge.

Other cross-layer techniques have been proposed for enhancing the security of wireless communication systems. One such approach involves the integration of cryptographic-based methods with PHY-layer-based methods in diverse ways. Zenger et al. [95] introduced a novel technique in situations where computational capabilities are limited. This technique involves two distinct authentication phases, referred to as Phases I and II. In Phase I, authentication is dependent upon channel characteristics, specifically, the high correlation coefficient observed between channel estimates of different terminals $h_{B\to A}(t) \approx h_{C\to A}(t)$, where A is the access point, B is the authenticated node, and C is the unauthenticated node. This correlation coefficient is observed when the distance between nodes B and C is less than or equal to half of the wavelength ($\lambda/2$), as depicted in Fig. 2.11. The delegation of trust between entities (B) and (C) is examined in the context of their proximity to each other within the vicinity zone and upon receipt of a command by the user. Phase II involves the use of a PHY-layer key extraction method to generate a shared key that can be utilised for cryptographic purposes at the upper layers.

Chen et al. [96] introduced a novel authentication scheme, referred to as clustering-based PHY-layer authentication scheme (CPAS). The proposed scheme is intended for two legitimate entities, namely Alice and Bob, who have a pre-established shared key (*key*). The cryptography-

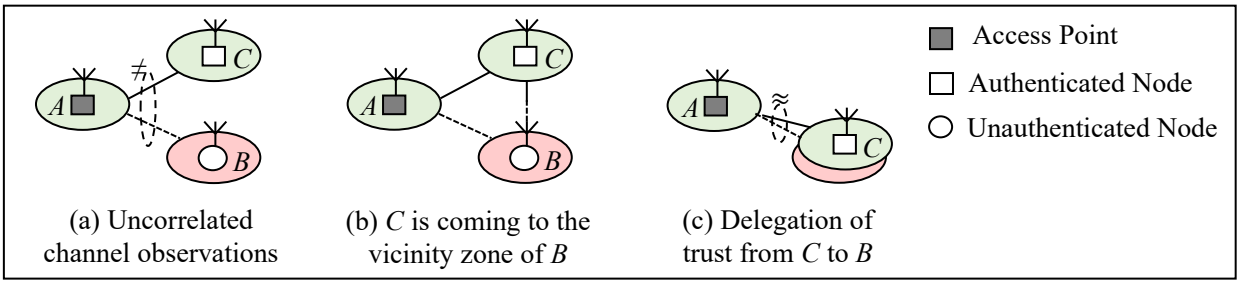


Figure 2.11: PHY-layer authentication based on vicinity zone [95].

based authentication process comprises the following steps.

1. Alice generates a pseudorandom number denoted as PS_1, which is subsequently encrypted using a lightweight symmetric encryption algorithm, resulting in a cipher denoted as $\gamma_1 = E_{key}(PS_1)$. Alice sends the cipher to Bob, who extracts the CSI (H_1) and decrypts the cipher to obtain the original pseudorandom number as $PS'_1 = D_{key}(\gamma'_1)$. Bob then concatenates pairs of pseudorandom numbers, namely (PS_2, PS_3), with the original pseudorandom number PS'_1, resulting in a new sequence that is subsequently encrypted using a shared secret key, denoted as $\gamma_2 = E_{key}(PS'_1 \| PS_2 \| PS_3)$. The resulting cipher γ_2 is then transmitted back to Alice.

2. Upon receiving the encrypted message γ_2 from Bob, Alice decrypts the message using the shared secret key to obtain the concatenated sequence of pseudorandom numbers (PS'_1, PS'_2, PS'_3). To verify the authenticity of Bob, Alice tests whether the decrypted pseudorandom number PS'_1 obtained from γ_1 is equal to the original pseudorandom number PS_1. If Alice is confident in Bob's identity, she encrypts the pseudorandom numbers PS'_2 and PS'_3 using the shared secret key, resulting in two new ciphers denoted as $\gamma_3 = E_{key}(PS'_2)$ and $\gamma_4 = E_{key}(PS'_3)$. Alice subsequently sends the encrypted messages γ_3 and γ_4 to Bob.

3. Upon receiving the encrypted messages γ_3 and γ_4 from Alice, Bob decrypts the ciphers using the shared secret key to obtain the pseudorandom numbers PS'_2 and PS'_3, respectively. Bob then estimates the CSI for the newly established connection, denoted as H_2 and H_3. To ensure the legitimacy of Alice, Bob compares the decrypted pseudorandom numbers (PS'_2, PS'_3) with the original pseudorandom numbers (PS_2, PS_3). If the two sets of pseudorandom numbers are equal, it confirms the authenticity of Alice, and the communication between Alice and Bob can proceed securely.

The CPAS scheme comprises the following steps.

1. Bob obtains the CSI for the three connections, denoted as H_1, H_2, and H_3. To estimate the statistical characteristics of the channel, Bob accumulates the absolute values of the real

and imaginary parts of each element estimated from the number of m subcarriers and n antennas. This process results in three new points denoted as $H_1' = \{x_1, y_1\}$, $H_2' = \{x_2, y_2\}$, and $H_3' = \{x_3, y_3\}$, as depicted in Fig. 2.12.

2. Using the estimated statistical characteristics $H_l' = \{x_l, y_l\}, \forall l \in \{1,2,3\}$, the central point $W_i(x,y)$, and the coverage area $dist_i$ are evaluated. The distance $dist_i$ is determined by adding the radius of the coverage area R (see Fig. 2.12) to an adjusting parameter θ.

3. To authenticate a terminal k, Bob calculates the Euclidean distance between the statistical information received from the terminal and the central point W_i of the coverage area for each legitimate terminal i. The Euclidean distance is denoted as $\|H_k' \; W_i\|$. If $\|H_k' \; W_i\|$ is less than the distance threshold $dist_i$, then the terminal k is authenticated as a legitimate device. Otherwise, k is authenticated as an unauthorised device.

In case Alice fails to pass the CPAS scheme, the lightweight symmetric encryption scheme is executed as an alternative. Once a new terminal has passed the cryptographic scheme, it can be added to the network. Finally, the new central point $W_{new}\{x_{new}, y_{new}\}$ and coverage distance $dist_{new}$ are estimated for the newly added terminal.

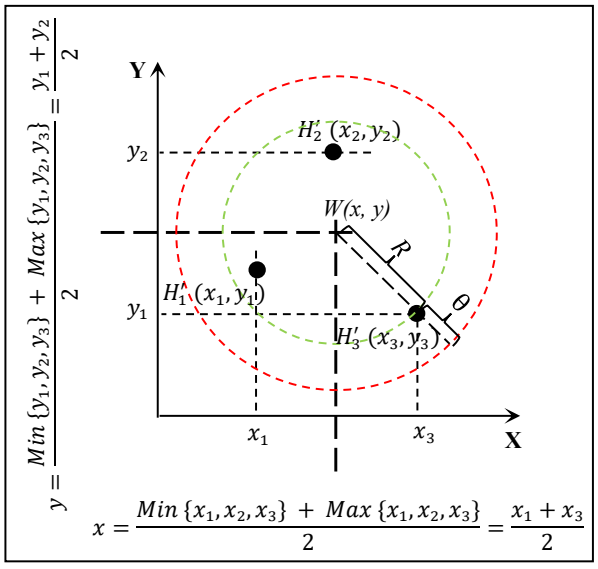


Figure 2.12: Principle map of the PHY-layer channel observations in [96].

In the context of VANETs, the integration of PHY-layer with upper-layer authentication is a critical consideration. The successful integration of these components must be both rational and practical, taking into account the application nature in terms of factors such as dynamicity, resource availability, broadcasting rate, and channel conditions. One of the key challenges in this regard is the selection of an appropriate re-authentication technique. This selection must be made with careful consideration of the aforementioned factors, as well as other relevant considerations such as security, privacy, and scalability. This thesis conducts comprehensive research

on the available PHY-layer re-authentication techniques and their suitability for VANET applications. This research involves a detailed analysis of the strengths and weaknesses of different techniques, as well as an evaluation of their performance under various conditions.

2.5 PHY-layer secret key extraction

Besides authentication, the spatial and temporal variations of the wireless channel can also be exploited to extract a unique location-dependent shared key between the communicating terminals, supporting forward and backward secrecy in VANETs (an adversary cannot predict the previous or upcoming shared key based on the current one [97]). In addition, the development of a PHY-layer-based method for generating secret keys may provide an alternative to existing cryptography-based key-exchanging protocols, e.g., Diffie-Hellman [98]. To provide a comprehensive understanding of this approach, this section provides an overview of the PHY-layer secret key extraction process and discusses the challenges associated with the practical implementation of this method in VANETs.

2.5.1 Overview

The major source of randomness in the key extraction process is the unexpected variations in channel responses, i.e., received signal strength and phase [99]. The former is a random function resulting from the significant and unpredictable spatial and temporal fluctuations in each multipath component's path loss and shadowing, whereas the latter is a function of the delay, frequency offset, and Doppler shift. The key point is that a pair of communicating devices can observe reciprocal estimates of the spatially and temporally varying channel responses within the limited time interval T_c [99]. By probing and having channel estimates in the time division duplex (TDD) mode, the obtained estimates undergo three main stages, i.e., quantisation, information reconciliation, and privacy amplification, see Fig. 2.13 [100]. The quantisation stage is a mapping operation that converts the channel components into bitstreams. While the infor-

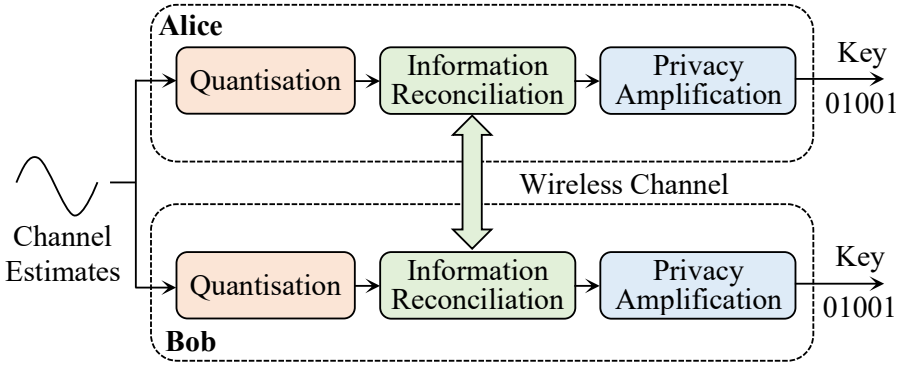


Figure 2.13: PHY-layer-based secret key extraction mechanism.

mation reconciliation stage is an error correction stage that involves correcting the mismatched bits resulting from the imperfect channel reciprocity. The channel non-reciprocity component results from the channel being probed in an interleaved fashion. The final stage utilises a hashing operation to maintain the secrecy of the extracted key. In general, the PHY-layer secret key extraction can be categorised into RSS-based, phase-based, and CIR-based, see Fig. 2.14.

2.5.2 RSS-based methods

Tope et al. [101] presented a novel protocol that evaluates signal attenuation caused by multipath channels, which are extracted from the envelope of received packets. In this work, channel estimates are not directly quantised. Instead, arrays of variations are produced by subtracting half values from the other half, thereby eliminating the predictable slowly changing component due to path loss that is correlated to the distance between the transmitter and the receiver. This process effectively removes any potential vulnerability to attacks that exploit such correlations. Additionally, two fixed thresholds are used to discard the lowest and highest values, respectively, thereby reducing the probability of mismatching and improving key robustness. Notably, the proposed scheme does not account for imperfect channel reciprocity but argues that the correlation between estimates could always be increased by using a sufficient probing rate.

Mathur et al. [102] investigated the relationship between quantisation parameters and key extraction performance. To mitigate the impact of shadow fading, which introduces significant fluctuations in the received signal power, a windowed average is introduced. The quantisation process involves using two thresholds ($q_{\pm}$), derived from the average (μ) and standard deviation

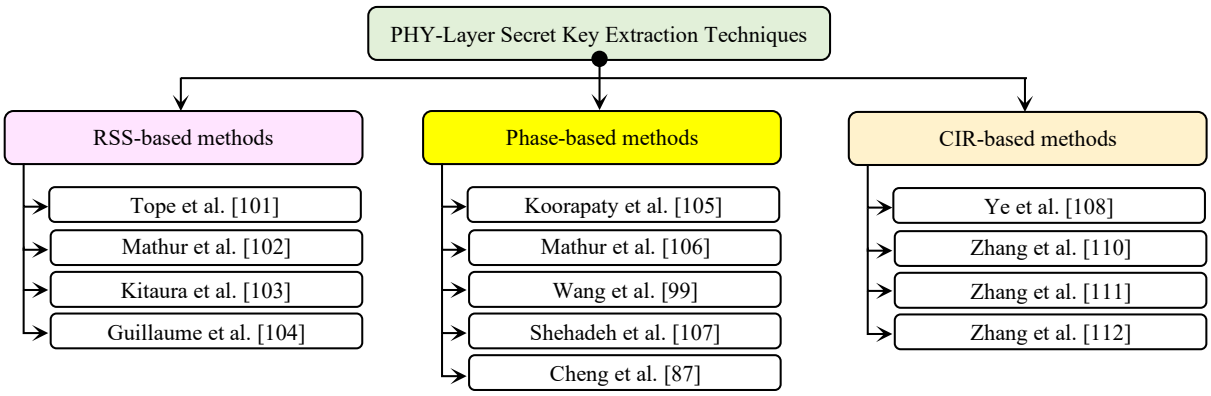


Figure 2.14: Classification of the PHY-layer secret key extraction techniques.

(σ) of the channel estimates $\underline{\hat{h}}$. The following formula calculates the thresholding levels.

$$q_{\pm} = \mu(\underline{\hat{h}}) \pm \alpha \cdot \sigma(\underline{\hat{h}}) \tag{2.6}$$

where α is an adjusting parameter. Finally, the quantisation function $Q(\cdot)$ is defined as

$$Q(x) = \begin{cases} 1, & \text{if } x > q_{+} \\ 0, & \text{if } x < q_{-} \\ \text{dropped} & \text{otherwise} \end{cases} \tag{2.7}$$

In this method, optimal key robustness is attained in Rician and Rayleigh fading models due to their symmetric nature around the distribution means. This symmetry results in an equivalent likelihood of positive and negative quantised samples.

Kitaura et al. [103] proposed an alternative quantisation approach that involves measuring and comparing different signals received by multiple antennas. The relative variations of these signals generate bits, whereby a quantised sample is designated as "1" if the estimated power of the signal received by a particular antenna (A) is greater than that of another antenna (B) from the same transmitter. Conversely, if the estimated power of the signal received by A is less than that of B, the quantised sample is designated as "0". However, noise accumulates from comparing different signal estimates may reduce key extraction performance in low SNR scenarios. In single antenna systems, wireless relays can be utilised to introduce randomness in static scenarios by acting as additional shared antennas. In [104], Guillaume et al. explored the use of a relay connected to legitimate parties via time-variant channels, which may be due to the relay's mobility. The process of key generation starts with Alice sending a randomly chosen variable to Bob and the relay. However, only Bob can deduce the original variable based on the pre-known static channel estimate between Alice and Bob. The relay subsequently transmits the received signal to Bob, who extracts the original signal and obtains an estimate of the channel connecting the relay. This approach exploits the channel randomness introduced by the intermediate relay. Simulations demonstrated that the introduction of a relay allowed for key extraction in static environments, at the cost of an increased bit mismatch rate (BMR).

2.5.3 Phase-based methods

Wireless cards are readily available for the acquisition of the RSS, which is why RSS is widely used [67]. However, RSS-based techniques suffer from limited capabilities for generating group keys due to the difficulty of safely accumulating RSS observations over multiple nodes. A further disadvantage is its inability to cope with slow channel variations (static or indoor cases) due to a lack of sufficient randomness (roughly static path loss and shadowing). Therefore, phase-based quantisation has emerged due to the high sensitivity of the channel-phase response

to the distance between terminals, allowing the high dynamicity of vehicular networks to be an advantage for obtaining high entropy shared keys. In the channel probing step, the imperfect channel reciprocity results in a mismatching error in the extracted bits [67]. To address this issue, Koorapaty et al. [105] employed the difference in the phase estimates between two signals of different frequencies sinusoids as a randomness source to reduce the channel non-reciprocity components' impact, and consequently the mismatching probability. Likewise, the study in [106] improved extraction performance by using the phase differentials and amplitudes as distinct sources of randomness. This approach leads to an improvement in the key generation rate and subsequently accelerates the process of symmetric key establishment.

In [99], a round-trip group key generation mechanism is proposed. This mechanism entails a member of a group of nodes initiating two signals with random phases and transmitting them through a ring group of nodes in clockwise and counterclockwise directions. By combining the channel response estimates generated by each node from both directions, a high degree of correlation is achieved, enabling the generation of a shared group key. However, while the mechanism demonstrates theoretical efficacy, its practical implementation is hindered by the accumulation of noise across multiple group nodes. Furthermore, the requirement for the entire channel probing process between all nodes to be completed within T_c poses a challenge. This time frame is often too short, given the short coherence period of high-speed terminals.

In all the above works, the quantisation process is designed without invalid regions; however, observations near the region's boundaries may result in a high rate of mismatches. To overcome this limitation, the work in [107] explored the utilisation of guard intervals to reduce the probability of mismatches. However, a trade-off exists between the mismatching probability and the bit extraction rate. Larger boundary regions decrease the mismatching probability and result in a lower bit extraction rate due to the greater number of dropped observations. Multi-carrier communication systems encompass partitioning the frequency spectrum into a multitude of parallel subcarriers, which are then utilised as independent sources of randomness. In [87], a single-side probing mechanism was proposed for an OFDM system consisting of N subcarriers. The mechanism involves initiating random phase sequences by a terminal and utilising the reciprocal characteristics of the channel to mask the mapped preliminary shared key.

2.5.4 CIR-based methods

The RSS is a crucial parameter for wireless communication channels. However, it does not fully exploit the diversity and multipath behaviour of the channel. In contrast, the channel impulse response provides a more detailed representation of the channel state by characterising the distinct multipath components that form a train of discrete pulses with varying magnitudes and delays. The individual components of the CIR can be modelled using Rician or Rayleigh fading. A study conducted in [108] investigated the extraction of secret keys from jointly Gaussian random variables, motivated by the observation that wireless channel taps exhibit a complex

Gaussian distribution [109]. The researchers define the secret key capacity as a function of the SNR and evaluate the performance of two distinct quantisation schemes in terms of their ability to improve the key generation rate.

Zhang et al. [110] proposed an approach to increase the key generation rate by utilising CIRs from subcarriers that have been modelled. The quantisation method is based on the cumulative distribution function, which is utilised to achieve an approximately equal distribution of ones and zeros in the extracted bits. In subsequent studies documented in [111,112], the authors improved the scheme presented in [110] by designing a low-pass filter to mitigate the effects of noise. The simulation results show a reduced disagreement rate across all experimental scenarios.

2.5.5 Challenges and limitations

The process of secret key extraction encounters several challenges, one of which is the significant communication cost incurred by the reconciliation stage [100]. Common information reconciliation approaches such as the Cascade algorithm, low-density parity-check (LDPC) [113] and Turbo [114] codes suffer from some performance limitations. The former method involves exchanging information about matched and mismatched bits to align the keys correctly. However, this approach exposes 60% of the matched bits to reconcile only 10% of the mismatched bits, posing a security threat [100]. On the other hand, the LDPC and Turbo Codes are error correction methods that use specific structured codes to detect and fix errors in data transmission by employing iterative decoding algorithms. Unfortunately, these approaches suffer from high computation complexities [115]. This trade-off between security measures and computational efficiency underscores a fundamental challenge in the information reconciliation stage, wherein the need for security solutions often requires computationally expensive algorithms.

In addition, it is worth noting that current secret key extraction techniques have been developed based on the assumption that the network terminals are separated by more than $\lambda/2$ distance, enabling location decorrelation between legitimate and wiretapped channel responses. However, this assumption is unrealistic in V2I communication scenarios since an attacker can position a wireless card near a fixed RSU (i.e., $\leq \lambda/2$), resulting in highly correlated channel features between the surrounding vehicles, the RSU, and the attacker's wireless card, thus compromising the security of V2I applications by gaining access to the secret channel features of surrounding vehicles. In this challenging scenario, cryptography and number theory can be integrated to address some of these limitations, which represents a key objective of this thesis.

2.6 Summary

This chapter aims to address the first research question (Q_1) by examining the feasibility of utilising PHY-layer authentication as an alternative to crypto-based methods. The findings of this chapter suggest that PHY-layer authentication cannot effectively serve as a standalone authentication solution. However, it can be combined with cryptography-based authentication in various ways to leverage the physical layer's unique features in detecting and preventing various attack types while minimising computation and communication overheads. This is due to the low computation cost of the different operations executed at the physical layer ($\approx \mu s$) compared to the cryptographic operations conducted in the upper layers ($\approx ms$). Existing cross-layer authentication schemes, as illustrated in Fig. 2.1, have been introduced to the research community with the assumption that initial authentication has already been performed at the upper layer without discussing in detail how this process is performed. Moreover, the selection of an appropriate PHY-layer authentication method depends on the application's nature, including the broadcasting rate, channel variation, and availability of computation resources. This results from the differing performance of various PHY-layer authentication methods across different SNR levels. In other words, some authentication methods excel in detecting nodes' legitimacy at low SNR, while others do not. This implies that some authentication designs are feasible and effective for outdoor scenarios, such as dynamic and extremely dense applications, while others are not, requiring further investigation. In this context, the subsequent chapters of this thesis introduce various cross-layer authentication designs for VANET applications, providing higher security and authentication performance than traditional authentication techniques.

Chapter 3

Efficient Cross-Layer Authentication

Even though the cross-layer methods discussed in Section 2.4 can provide enhanced authentication, they cannot be applied to VANET applications due to vehicular channels' high mobility and temporal variability, a matter that deserves further investigation. This chapter develops a key-based PHY-layer challenge-response algorithm for re-authentication to fill this gap. In this algorithm, the preliminary key is mapped and masked by the channel-phase response to generate the response signal that can only be equalised at the side of the intended receiver, employing the short-term channel reciprocity and the same encapsulated key. To guarantee the channel reciprocity between high-speed terminals, the time required to generate the response signal is estimated and compared to an indicative minimum coherence time of V2V communication, as a worst-case scenario. Furthermore, this study examined the detection probability of re-authentication at small SNRs for an acceptable false alarm probability. In addition, the scheme's security strength is proven against typical adversarial attacks, including replaying, impersonation, and denial of services.

The following summarises the contributions of this chapter which are published in [18], fulfilling the outlined thesis objectives (1, 2, 5, 6) detailed in Subsection 1.4.3:

1. A low-complexity cross-layer authentication scheme is proposed for VANETs applications, employing the short-term channel reciprocity and randomness for re-authentication to address some of the performance limitation issues, particularly those related to the significant overheads of signatures generation and verification.

2. A lightweight pseudo-identity-based algorithm is proposed to initially verify the legitimacy of the corresponding terminals at the first time slot, which increases the scheme's availability and mitigates the effect of the flooding type of DoS attacks on the network. For re-authentication, a location-dependent-based PHY-layer re-authentication step is proposed for the identity re-verification process, which helps in detecting and preventing Sybil types of attacks.

3. Furthermore, this chapter presents how the proposed scheme can fulfil the security and

privacy requirements of VANETs. In this way, the unforgeability of signatures is proven against adaptive chosen message attacks in the random oracle model (ROM) (for background, see [117]), ensuring the resistance of the proposed scheme to impersonation and modification attacks.

4. Besides theoretical analysis, an extensive simulation is conducted to examine the detection probability of the PHY-layer re-authentication process at small SNRs ≥ 5 dB. In addition, the timing analysis of the challenge-response process is investigated to ensure that the wireless channel exhibits short-term reciprocity under conditions of high-speed terminals of up to $\approx 30\, m/s$. Finally, the computation and communication comparison and security analysis show that the proposed scheme offers security and cost-saving advantages over crypto-based signatures.

The rest of this chapter is organised as follows. The structure of the proposed cross-layer authentication scheme is presented in Section 3.1, while Section 3.2 discusses the adopted threat model. Section 3.3 presents extensive performance analysis and comparisons regarding computation and communication overheads. Finally, Section 3.4 concludes this chapter.

3.1 The proposed cross-layer authentication scheme

In this section, the system model for the proposed cross-layer scheme is presented in subsection 3.1.1. Next, each step is described in detail in subsections 3.1.2, 3.1.3, 3.1.4, and 3.1.5.

3.1.1 System model for the proposed cross-layer authentication scheme

The novelty of the proposed scheme relies on exploiting the short-term channel reciprocity between two communicating terminals for re-authentication. The corresponding terminal is re-authenticated at the physical layer in a challenge-response process, providing efficient and secure verification in a low processing time. Fig. 3.1 presents the flowchart of the proposed approach, which can be described through the following steps.

- **S1. Initial Authentication:** A conditional privacy preservation authentication algorithm (ACPPA) is proposed for mutual identity verification using the upper layer's authentication by exchanging pseudo-identities between both terminals.
- **S2. Secret Key Extraction:** If the initial verification holds, the key extraction algorithm in [115] is employed to extract a location-dependent shared key between both terminals. Otherwise, the authentication process is ended.
- **S3. PHY-Layer Re-authentication:** Under binary hypothesis testing [116], the re-authentication step is performed at the physical layer using a PHY challenge-response algorithm based on the extracted key with a sufficient number of matched bits.

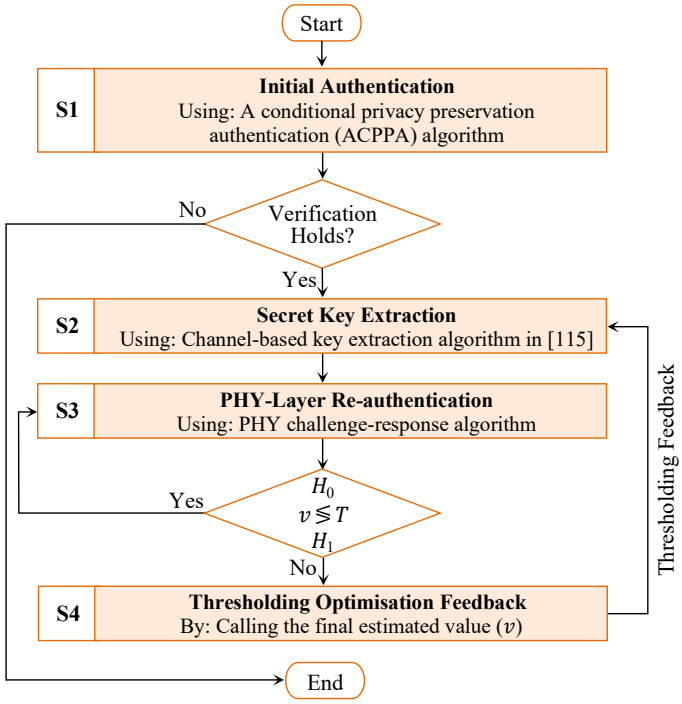


Figure 3.1: Flowchart of the proposed authentication scheme.

- **S4. Thresholding Optimisation Feedback:** In the case of failure, the key extraction step (S2) is re-executed after adapting the thresholding values based on the feedback from the re-authentication step (S3).

The low complexity of the proposed scheme stems from the integration of the re-authentication step S3 into S1. In doing so, the computation and communication overheads associated with signing and distributing signatures are drastically reduced for each transmission. Moreover, the scheme availability is ensured by designing a lightweight initial identity verification step represented in S1, mitigating the effect of DoS attacks. As for Sybil attacks detection, S2 is integrated into S3 to provide location-dependent-based re-authentication at the physical layer. At last, the thresholding optimisation feedback step S4 is used to adjust the key extraction parameters of S2 based on the re-authentication feedback from S3. All network terminals are assumed to be working in the TDD mode with a single antenna and separated by more than $\lambda/2$ distance. The channel responses between legitimate and wiretap channels are uncorrelated. RSUs and vehicles' OBUs are supposed to be synchronised with the TA.

3.1.2 Overview of the initial authentication step (S1)

The proposed ACPPA algorithm is presented in this subsection for V2V as a case study for vehicular communication. This process aims to identify the legitimacy of the corresponding terminal initially. A location-dependent shared key will be extracted according to the signature verification result. A pseudo-identity-based algorithm is proposed to identify the corresponding

Table 3.1: List of notations for the proposed ACPPA algorithm

Symbol	Definition
RID_{V_i}	Real identity of the vehicle V_i
TID_{V_i}	Temporary identity of V_i
PPs	Algorithm's public parameters
β	TA's master key
r_{V_i}	Private key of V_i
PK_{V_i}	Public key of V_i
$PK_{V_i,TA}$	Public key of the V_i and TA
PK_{RV_i}	Public key of the revoked vehicle RV_i
PID_{V_i}	Pseudo-identity of $V_i, PID_{V_i} = \{PID_i^1, PID_i^2\}$
σ_{V_i}	Signature generated by V_i
$SK_{V_{i-j}}$	Session key between two communicating vehicles V_i and V_j
GRL	General revocation list generated by the TA
TID_{GRL}	List of revoked vehicles' TIDs generated by V_i
T_i	Signature's timestamp generated by V_i
T_r	Signature's receiving time at the intended receiver
T_Δ	Freshness expiry time [0:00:59]
$\perp$	Empty string

terminal's legitimacy based on ECC scalar multiplications, avoiding using map-to-point hash functions and bilinear pairing time-consumed operations. The proposed algorithm consists of five phases - i.e., system initialisation, registration, identity authentication, reporting, and real identity tracking. The notations used in this subsection are listed in Table 3.1. Fig. 3.2 presents the top-level description of the S1 algorithm's substeps detailed below.

S1.1. *System initialisation phase*: The TA generates the system's public parameters via the following processes.

- Choosing two large prime numbers p and q, and 160-bits elliptic curve E for 80-bits security defined by $y^2 = x^3 + ax + b \ mod \ p$ over a prime field F_p for $a, b \in F_p$, where $\Delta = 4a^3 + 27b^2 \neq 0$.
- Construction of the cyclic additive group $\mathbb{G}$ of order q based on the generator P, so that $\mathbb{G}$ consists of all the points on E and the infinity point $\mathcal{O}$.
- Randomly choosing the system master key $\beta \in Z_q^*$.
- Selecting the hash function $H_1 : \mathbb{G} \rightarrow \{0,1\}^{N_1}$ and the hash message authentication code $HMAC_{key}(x) : (key : \mathbb{G}, x : \{0,1\}^*) \rightarrow \{0,1\}^{N_2}$.
- Finally, the algorithm's public parameters are $PPs : \langle a,\ b, P, p, q, \mathbb{G}, H_1, HMAC \rangle$.

S1.2. *Registration phase*: Before joining the network, each vehicle V_i must register with the TA to obtain the algorithm's public parameters according to the following sub-steps.

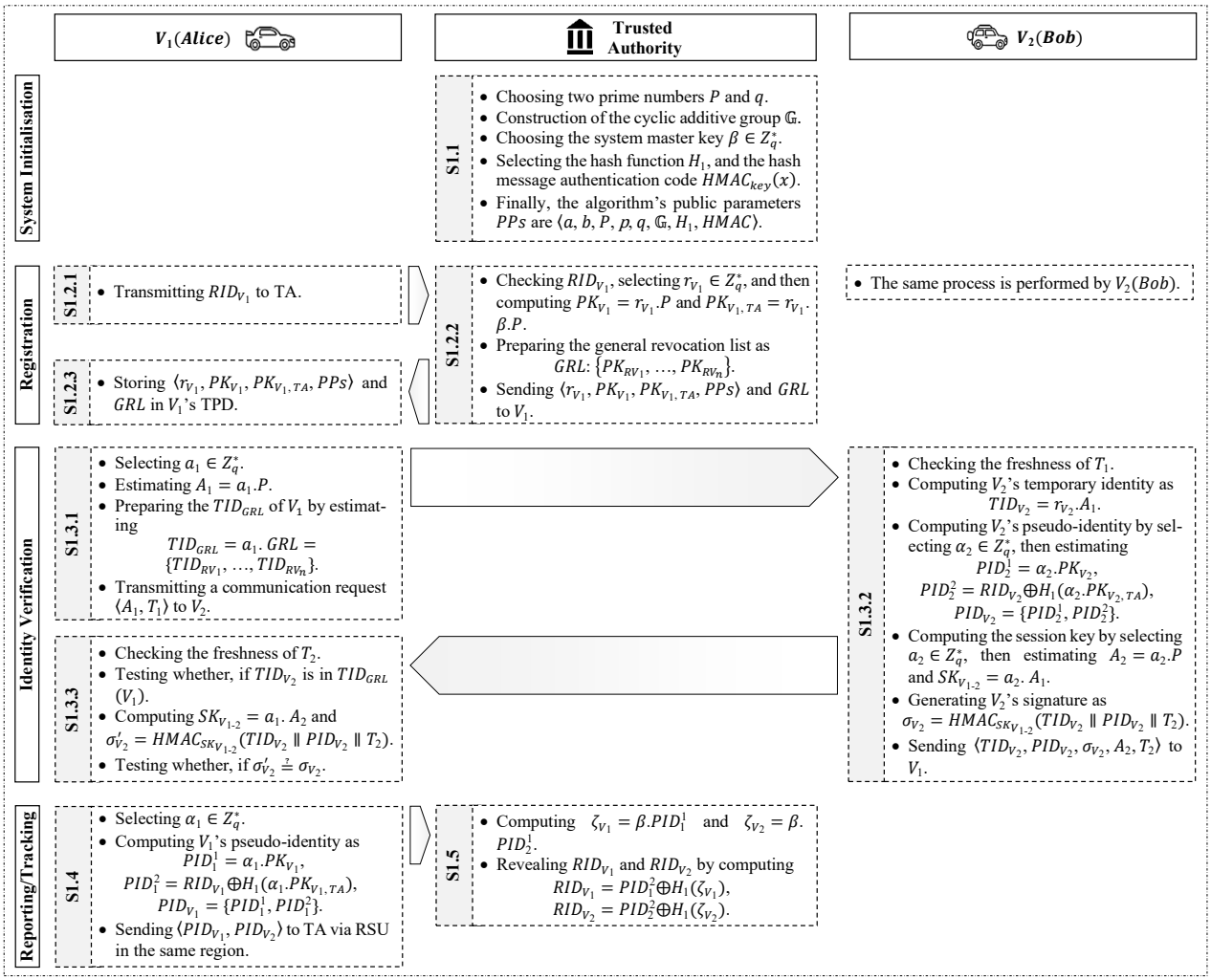


Figure 3.2: The top-level description of the proposed ACPPA algorithm.

- **S1.2.1.** V_i transmits its unique RID_{V_i} (e.g., license number) to the TA to check the validation status of the RID_{V_i}.
- **S1.2.2.** The TA prepares V_i's secret parameters as follows.
  - The TA checks the RID_{V_i}, selects a random private number $r_{V_i} \in Z_q^*$ of V_i, and calculates its relevant public keys as $PK_{V_i} = r_{V_i}.P$, and $PK_{V_i,TA} = r_{V_i}.\beta.P$.
  - The TA prepares the general revocation list GRL, which is a list of public keys of revoked vehicles distributed between vehicles and RSUs and equals GRL: $\{PK_{RV_1}, PK_{RV_2}, \ldots, PK_{RV_n}\}$.
- **S1.2.3.** During V_i's registration, the TA stores the tuple $\langle r_{V_i}, PK_{V_i}, PK_{V_i,TA}, PPs \rangle$ and GRL in V_i's TPD.

S1.3. *Identity authentication phase*: Mutual identity authentication between $V_1(Alice)$ and $V_2(Bob)$ is conducted when V_2 is in the transmission range of V_1. Without loss of generality, the

one-way authentication process consists of three main stages.

- **S1.3.1.** *Communication request stage*: In this stage, a vehicle V_1 randomly selects $a_1 \in Z_q^*$, computes its corresponding public parameter $A_1 = a_1 \cdot P$, then prepares its revocation list by estimating the list of temporary identities TIDs of revoked vehicles based on the general revocation list GRL as $TID_{GRL}(V_1) = a_1 \cdot GRL = \{TID_{RV_1}, \ldots, TID_{RV_n}\}$, and sends a communication request $\langle A_1, T_1 \rangle$ to V_2 at timestamp T_1.

- **S1.3.2.** *Signature generation stage*: In this stage, a vehicle V_2 checks the freshness of the received timestamp T_1 by testing whether $T_r - T_1 \leq T_\Delta$ holds or not, hides its real identity by computing its temporary identity $TID_{V_2} = r_{V_2} \cdot A_1$ and pseudo-identity PID_{V_2}. To generate a valid PID_{V_2}, V_2 chooses at random $\alpha_2 \in Z_q^*$, computes $PID_2^1 = \alpha_2 \cdot PK_{V_2}$ and $PID_2^2 = RID_{V_2} \oplus H_1\left(\alpha_2 \cdot PK_{V_2,TA}\right)$ to attain its pseudo-identity $PID_{V_2} = \left\{PID_2^1, PID_2^2\right\}$. Then, V_2 calculates its signature σ_{V_2} by selecting at random $a_2 \in Z_q^*$, calculating its relevant public parameter $A_2 = a_2.P$ and the key $SK_{V_{1-2}} = a_2.A_1$ to obtain the signature $\sigma_{V_2} = HMAC_{SK_{V_{1-2}}}(TID_{V_2} \| \ PID_{V_2} \| T_2)$ created at the T_2 timestamp. Finally, V_2 replies to V_1's request by sending the tuple $\langle TID_{V_2}, PID_{V_2}, A_2, T_2, \sigma_{V_2} \rangle$ to V_1.

- **S1.3.3.** *Signature verification stage*: In this stage, V_1 checks the freshness of the timestamp T_2, verifies the legitimacy of V_2 by finding out if $TID_{V_2} \in TID_{GRL}(V_1)$, then checks the integrity of the received message by computing $SK_{V_{1-2}} = a_1 \cdot A_2$ and $\sigma'_{V_2} = HMAC_{SK_{V_{1-2}}}(TID_{V_2} \| PID_{V_2} \| T_2)$ and testing whether $\sigma'_{V_2} \stackrel{?}{=} \sigma_{V_2}$ holds or not. The same process is reversed between the communicating terminals for mutual authentication.

S1.4. *Reporting phase*: Misbehaving vehicles can be reported, let us consider V_1 wants to report V_2. In that case, V_1 randomly selects $\alpha_1 \in Z_q^*$, generates vehicle's pseudo-identity by computing $PID_1^1 = \alpha_1 \cdot PK_{V_1}$ and $PID_1^2 = RID_{V_1} \oplus H_1\left(\alpha_1 \cdot PK_{V_1,TA}\right)$ to obtain $PID_{V_1} = \left\{PID_1^1, PID_1^2\right\}$. Finally, V_1 reports V_2 by sending the tuple $\langle PID_{V_1}, PID_{V_2} \rangle$ to the TA through the RSU in the same region, in which PID_{V_1} and PID_{V_2} are the pseudo-identities of the reporter and misbehaving vehicles, respectively.

S1.5. *Real identity tracking phase*: The RIDs of the reporter and misbehaving vehicles can be revealed by the TA based on the received tuple $\langle PID_{V_1}, PID_{V_2} \rangle$ and TA's master key β by computing $\zeta_{V_i} = \beta.PID_i^1$ and $RID_{V_i} = PID_i^2 \oplus H_1(\zeta_{V_i})$. The proof of correction is verified as follows:

$$\begin{aligned} RID_{V_i} &= PID_i^2 \oplus H_1(\zeta_{V_i}) \\ &= RID_{V_i} \oplus H_1\left(\alpha_i \cdot PK_{V_i,TA}\right) \oplus H_1\left(\beta \cdot PID_i^1\right) \\ &= RID_{V_i} \oplus H_1\left(\alpha_i \cdot PK_{V_i,TA}\right) \oplus H_1\left(\alpha_i \cdot \beta \cdot PK_{V_i}\right) \\ &= RID_{V_i} \oplus H_1\left(\alpha_i \cdot PK_{V_i,TA}\right) \oplus H_1\left(\alpha_i.PK_{V_i,TA}\right) = RID_{V_i} \end{aligned}$$

3.1.3 Review of the secret key extraction algorithm in [115] (S2)

Channel randomness is a natural-correlated resource for extracting a high entropy shared key between terminals. Generally, the key generation process consists of four stages - i.e., channel probing, quantisation/thresholding, information reconciliation, and privacy amplification. In the proposed scheme, the key extraction algorithm in [115] is evoked to obtain a symmetric shared key with equiprobabilities of 0s and 1s and a sufficient rate of secret bit generation, defined by the ratio of the number of matching bits to the total number of channel samples. The contribution presented in [115] involves optimising the thresholding values within the quantisation stage, leveraging the perturb-observe algorithm. This algorithm dynamically adjusts the quantisation thresholds based on feedback obtained from the information reconciliation stage, aiming to enhance key extraction performance specifically tailored to varying channel conditions. The perturb-observe algorithm operates iteratively, utilising observed feedback to adaptively optimise the quantisation process, thereby maximising the efficacy of key extraction under varying channel characteristics. However, in order to avoid the high communication overhead of reconciling the discrepancies in the extracted key, the information reconciliation and privacy amplification stages are excluded from the key generation process [115].

In high-density V2V channel conditions with many fixed and moving scatterers (e.g., other vehicles), the received signal is the superposition of L multipath components of different paths with different phase delays ϕ_l and fading coefficients $|a_l|$ [115], see Fig. 3.3. The channel estimations at each side $Ch_{A\leftarrow B}(t)|_A$ for Alice and $Ch_{A\rightarrow B}(t)|_B$ for Bob can be formulated at instance time t as

$$Ch_{A\leftarrow B}(t)|_A \approx Ch_{A\rightarrow B}(t)|_B = \sum_{l=1}^{L} |a_l| e^{(j\phi_l)} e^{2\pi v_l t} \tag{3.1}$$

where v_l is the Doppler shift of each multipath component l which is the sum of that of Alice $v_{A,l}$, Bob $v_{B,l}$, and scatterers $v_{S,l}$ [118] as

$$v_l = v_{A,l} + v_{B,l} + v_{S,l} \tag{3.2}$$

Note that, the scatterers' speed can follow the Weibull distribution (with shape and scale parameters a and ω, respectively) [119].

Since the channel probing stage is performed in the half-duplex mode, the channel gain complement method is utilised to compensate the non-reciprocity components. However, zero-mean complex Gaussian noise $\mathscr{C}N\left(0, 2\sigma_C^2\right)$ still exists and is considered to be the difference between the uplink $Ch_{A\rightarrow B}(t)|_B$ and the downlink $Ch_{A\leftarrow B}(t+\Delta t)|_A$ channel responses at each side of the communicating terminals [115] as

$$Ch_{A\rightarrow B}(t)|_B = Ch_{A\leftarrow B}(t+\Delta t)|_A + \mathscr{C}N\left(0, 2\sigma_C^2\right) \tag{3.3}$$

where $\Delta t \leq T_c$. In [115], the perturb-observe algorithm is used to optimise the quantisation levels

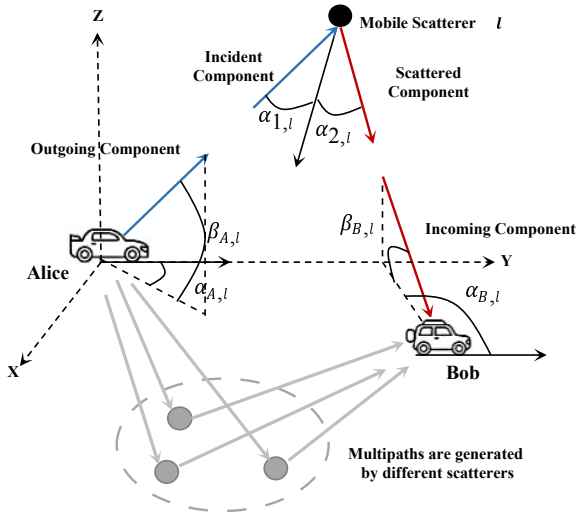
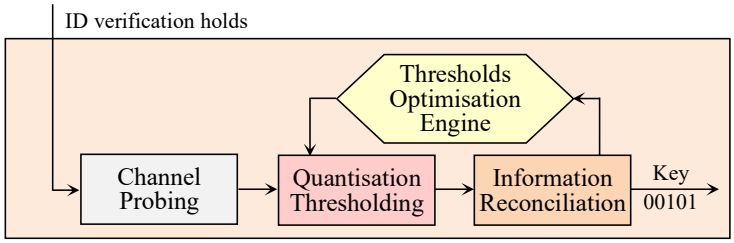


Figure 3.3: Non-line-of-sight V2V channel model [115].

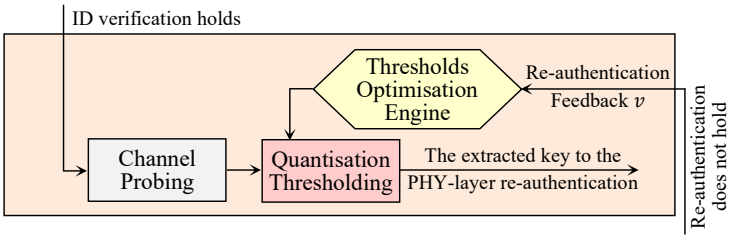
at different estimated non-reciprocity values σ_c based on the feedback from the information reconciliation stage, see Fig. 3.4(a). In this work, the information reconciliation stage is excluded. As a result, the PHY-layer re-authentication is used as alternative feedback for the thresholds optimisation engine, as illustrated in Fig. 3.4(b). This feedback indicates the level of mismatching resulting from different non-reciprocity values between the communicating terminals.

Step (S2) comprises three substeps as follows.

- **S2.1.** *Channel Probing*: Probing signals are exchanged between the communicating terminals to obtain highly correlated estimates within T_c.
- **S2.2.** *Quantisation thresholding*: Two thresholds quantisers (q_+, q_-) are used to convert the estimated channel observations into bits.



(a) Quantisation thresholds optimisation technique in [115].



(b) The developed thresholds optimisation technique.

Figure 3.4: PHY-layer secret key extraction algorithm.

- **S2.3.** *Thresholds optimisation engine*: Applying the perturb-observe algorithm [115] to adapt the quantisation levels in response to the feedback from the re-authentication step (S3).

Eventually, the extracted key $k_{\{a,b\}}$ is used for the mutual re-authentication process that is discussed in the following subsection (for more information about the secret key extraction algorithm, see reference [115]).

3.1.4 Overview of the PHY-layer re-authentication step (S3)

After identity verification and the extraction of the shared key $k_{\{a,b\}}$ between legitimate parties, Alice and Bob, the generated key is partitioned into two equal-length preliminary keys $k_{\{a,b\}} = (k_a \| k_b)$ used for the two-way re-authentication process. Alice transmits a challenge signal to Bob. The latter responds by encapsulating the mapped key k_b into the response signal that can be equalised at the side of Alice by exploiting the short-term channel reciprocity and the same encapsulated key. A one-way re-authentication process for N subcarriers OFDM system is considered, as illustrated in Fig. 3.5. For mutual re-authentication, the process is reversed and repeated between terminals based on the second part of the extracted key k_a.

The detailed sub-steps are as follows:

S3.1. *PHY communication request*: Bob transmits a communication request to Alice. This request contains the pseudo-identity PID_1^1 of Alice and T_i timestamp.

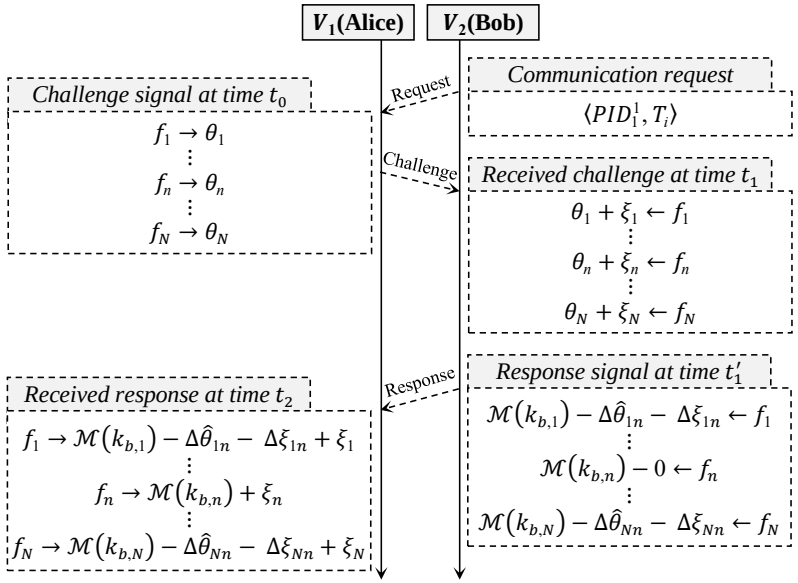


Figure 3.5: One-way PHY challenge-response re-authentication algorithm for OFDM system in the frequency domain.

S3.2. *PHY challenge*: Alice infers from the communication request that a pre-authenticated vehicle is trying to communicate with him. Then Alice initiates a PHY challenge frame for N subcarriers OFDM communication system and sends an initial challenge modulated sinusoidal signal to Bob with random phases θ_i uniformly distributed over $[0,2\pi)$ with frequencies $\{f_1,\ldots,f_N\}$ so that the transmitted signal at instance time t_0 can be expressed as

$$s_a(t_0)=\sum_{i=1}^{N}\sqrt{\frac{2E_s}{T}}\cos\left(2\pi f_i t_0+\theta_i\right),\theta_i\sim U[0,2\pi) \tag{3.4}$$

At the receiver's terminal, the received signal by Bob at time t_1 is formulated in a noiseless channel as

$$r_b(t_1)=\sum_{i=1}^{N}\sqrt{\frac{2|h_i|^2E_s}{T}}\cos\left(2\pi f_i t_1+\psi_i\right) \tag{3.5}$$

where $\psi_i=\theta_i+\xi_i,h_i$ for $i=1,2,\ldots,N$ are independent and identically distributed ($i.i.d.$) random variables with zero mean and variance $Var(h_i)=2\sigma^2$, and $\angle(h_i)=\xi_i\sim U[0,2\pi)$ which is the i^{th} subchannel-phase response of parallel Rayleigh fading channel of N subcarriers with probability density function $p(\xi_i)=1/2\pi$. After that, Bob estimates the phase difference of the received signal $\Delta\hat{\psi}_{in}=\psi_i-\psi_n=\Delta\hat{\theta}_{in}+\Delta\xi_{in}$, in which n is a randomly selected subcarrier index that ranges from 1 to N and can be altered by Bob at each iteration. The phase difference estimation can be expressed as

$$u_i=r_{b,i}r_{b,n}^{*},\Delta\widehat{\psi}_{in}=\tan^{-1}\left(\frac{imag(u_i)}{real(u_i)}\right) \tag{3.6}$$

S3.3. *PHY response*: A Gray code mapping operation $\mathscr{M}(.)$ of order 2 bits is used to map the preliminary key $k_b=\{\varkappa_1\varkappa_2,\ \varkappa_3\varkappa_4,\ldots,\varkappa_{2N-1}\varkappa_{2N}\}$ of length $2N$-bits at the side of Bob as below:

$$\phi_i=\mathscr{M}(k_{b,i})=\begin{cases}0 & k_{b,i}=[\ 0\quad 0\ ]\\ \frac{\pi}{2} & k_{b,i}=[\ 0\quad 1\ ]\\ \pi & k_{b,i}=[\ 1\quad 1\ ]\\ \frac{3\pi}{2} & k_{b,i}=[\ 1\quad 0\ ]\end{cases} \tag{3.7}$$

for $i=1,2,\ldots,N$. After that, Bob responds to Alice's challenge by encapsulating the mapped key ϕ_i and the estimated phase difference $\Delta\widehat{\psi}_{\text{in}}$ into the response signal and transmitting it to Alice at time t_1' as

$$\begin{aligned}s_b(t_1')&=\sum_{i=1}^{N}\sqrt{\frac{2E_s}{T}}\cos(2\pi f_i t_1'+\phi_i-\Delta\widehat{\psi}_{in})\\&=\sum_{i=1}^{N}\sqrt{\frac{2E_s}{T}}\cos(2\pi f_i t_1'+\phi_i-\Delta\hat{\theta}_{in}-\Delta\xi_{in})\end{aligned} \tag{3.8}$$

The received signal by Alice at time t_2 is formulated in a noiseless channel as

$$\begin{aligned} r_a(t_2) &= \sum_{i=1}^{N} \sqrt{\frac{2|h_i|^2 E_s}{T}} \cos(2\pi f_i t_2 + \phi_i - \Delta\hat{\theta}_{in} - \Delta\xi_{in} + \xi_i) \\ &= \sum_{i=1}^{N} \sqrt{\frac{2|h_i|^2 E_s}{T}} \cos(2\pi f_i t_2 + \phi_i - \Delta\hat{\theta}_{in} + \xi_n) \end{aligned} \tag{3.9}$$

Equalising $r_a(t_2)$ by using the phase θ_i of the initial signal $s_a(t_0)$ in (3.4), mapping the preliminary key k_b at the side of Alice $\hat{\phi}_i = \mathscr{M}\left(k_{b,i}\right)$, and computing $r_a(t_2)\, e^{j\left(-\widehat{\phi}_i+\theta_i\right)}$ so that the estimated signal by Alice at time t_2' can be simplified as

$$\begin{aligned} c\left(t_2'\right) &= r_a(t_2)\, e^{j\left(-\widehat{\phi}_i+\theta_i\right)} \\ &= \sum_{i=1}^{N} \sqrt{\frac{2\,|h_i|^2 E_s}{T}} \cos(2\pi f_i t_2' + \phi_i - \Delta\hat{\theta}_{in} + \xi_n - \hat{\phi}_i + \theta_i) \\ &= \sum_{i=1}^{N} \sqrt{\frac{2\,|h_i|^2 E_s}{T}} \cos\left(2\pi f_i t_2' + \theta_n + \xi_n + \phi_{e,i}\right) \end{aligned} \tag{3.10}$$

where $\phi_{e,i}$ is an estimated phase difference error resulting from the i^{th} subcarrier that holds mismatched bits and can be expressed as

$$\phi_{e,i} = \phi_i - \hat{\phi}_i \begin{cases} value & \phi_i \neq \hat{\phi}_i \\ 0 & \phi_i = \hat{\phi}_i \end{cases} \tag{3.11}$$

S3.4. *Verification process*: Alice checks the legitimacy of Bob by verifying the encapsulated key. Suppose the PHY response is sent from a third party (Eve impersonates the legitimate party, Bob). In that case, it is assumed that Eve generated a random binary key vector k_e for authentication, which can be represented as a hypothesis testing problem as indicated:

$$v(t_2') = Var(\sum_{i=1}^{N} \angle c_i(t_2')) \underset{H_1}{\overset{H_0}{\lessgtr}} T, \text{ for } \begin{cases} H_0: & \phi_i = \mathscr{M}\left(k_{b,i}\right) \\ H_1: & \phi_i = \mathscr{M}\left(k_{\mathrm{e},i}\right) \end{cases} \tag{3.12}$$

where T is the threshold value, and $Var(\sum_{i=1}^{N} \angle(c_i))$ is the circular variance of $\angle(c_i)$ which calculated as in [120] as

$$\begin{gathered} r_i = \begin{pmatrix} \cos(\angle(c_i)) \\ \sin(\angle(c_i)) \end{pmatrix}, \bar{r} = \frac{1}{N}\sum_i r_i \\ v = 1 - \|\bar{r}\| \end{gathered} \tag{3.13}$$

In binary hypothesis testing, the authentication judgment of the received signal (r_a) from the corresponding terminal is performed based on $v = (r_a, \phi_i)$. The decision rule is taken ac-

cording to the estimated measurement v, if the received response is sent from Bob $r_{a\leftarrow b}$, then v is estimated according to the joint distribution of $p\left(r_{a\leftarrow b}, \phi_i = \mathscr{M}\left(k_{b,i}\right)\right)$, while, the received response from Eve $r_{a\leftarrow e}$ obeys the distribution $p\left(r_{a\leftarrow e} \mid \phi_i = \mathscr{M}(k_{e,i})\right) \cdot \Pr(\widehat{\phi_i} = \mathscr{M}(k_{e,i}))$. As long as Eve possesses zero information about k_b, the hypothesis testing can be formulated a

$$T = \log \frac{p\left(r_{a\leftarrow b} \mid \phi_i = \mathscr{M}\left(k_{b,i}\right)\right)}{p\left(r_{a\leftarrow e} \mid \phi_i = \mathscr{M}\left(k_{e,i}\right)\right) \Pr(\widehat{\phi_i} = \mathscr{M}\left(k_{e,i}\right))} \tag{3.14}$$

The authentication judgment is further made by comparing v to the threshold value. The proposed algorithm is an extension of the work introduced in [86]. However, there is a distinction in the phase difference operation between the proposed approach, represented as $\Delta\hat{\theta}_{in}$ in (3.10), as opposed to $\Delta\hat{\theta}_{i1}$ in [86]. Since the decision rule depends on the circular variance $v = Var(\sum_{i=1}^{N} \angle(c_i))$, the remaining phase constant $(\theta_n + \xi_n)$ in (3.10) will not affect the final estimation result of v, giving the privilege of randomly selecting the subcarrier index n of the phase difference operation in (3.6).

S3.5. *Multi-vehicle communications*: For each vehicle V_j communicates with a number of n vehicles in the network, V_j stores a *List* of n tuples of vehicle's identities and their corresponding extracted shared keys as $List = \{Tuple_{V_1}, \ldots, Tuple_{V_n}\}$ in which $Tuple_{V_i} = \langle TID_{V_i}, PID_{V_i}, SK_{V_{i-j}} : k_{\{a,b\}}\rangle$. Considering vehicle V_i remains in the communication range of vehicle V_j for T seconds, then the duration T is divided into C time slots TS_L of length ΔT for $TS_L \in [(L-1).\Delta T, L.\Delta T]$ and $L \in [1, C]$.

For successful PHY-layer re-authentication process of n vehicles, the session key at time slot TS_L is periodically updated C times for all the corresponding vehicles in the *List*, as shown in Fig. 3.6, and can be formulated as

$$\begin{gathered} SK_{V_{i-j}}(TS_L) = (S_{L,x} \oplus S_{L,y}) \\ S_{L,x} = H_2^L(SD_x), S_{L,y} = H_2^{C-L+1}(SD_y) \end{gathered} \tag{3.15}$$

where SD_x and SD_y are the seed numbers and the x and y coordinates of the point $SK_{V_{i-j}} = \{SD_x, SD_y\} \in \mathbb{G}$, and $H_2^x(y)$ is the hash function $\{0,1\}^* \rightarrow \{0,1\}^{N_1}$ of the input variable y for x iterations. The computed $SK_{V_{i-j}}(TS_L)$ of length $N_1 = 160$ bits for SHA-1 hash function and the safety-related message m are concatenated with the transmitted PHY response for OFDM system of N subcarriers. The corresponding vehicle V_i verifies the received frame by searching in the *List* for $k_{\{a,b\}}$ related to the received session key $SK_{V_{i-j}}(TS_L)$ from vehicle V_j. In other words, the received $SK_{V_{i-j}}(TS_L)$ can be treated as an address to $k_{\{a,b\}}$ related to vehicle V_j. After that, V_i verifies the response signal by executing the verification process.

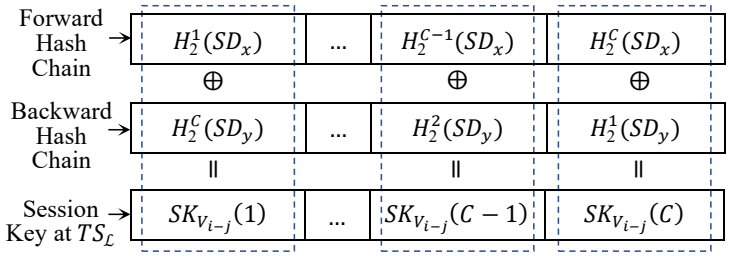


Figure 3.6: Hash chains used to generate $SK_{V_{i-j}}(TS_L)$.

3.1.5 The thresholding optimisation feedback step (S4)

In this step, the feedback value v in (3.13) denotes the level of mismatching between the mapped keys $\phi_{e,i} = \phi_i - \widehat{\phi}_i$, indicating the degree of channel non-reciprocity between both terminals. This feedback is an input to the thresholds optimisation engine S2.3. In the case of false decision-making due to a high mismatching percentage, the key extraction step (S2) is re-executed after adjusting the quantisation region $(q_+ - q_-)$. Increasing the quantisation region reduces the mismatching percentage, improving the detection probability of the re-authentication step at subsequent time slots.

3.2 Threat model of the proposed scheme

In this section, design goals in terms of security and privacy objectives are introduced, followed by a detailed discussion of how the proposed scheme satisfies these goals.

3.2.1 Design goals for the proposed Scheme

The proposed scheme must satisfy the following security and privacy objectives [5, 121].

1. *Privacy preservation*: Semi-trusted terminals (RSUs) or distrusted terminals (surrounding vehicles) cannot extract identifiable data about the sender from message contents.

2. *Non-Repudiation*: The transmitter cannot deny the authorship of the transmitted signatures.

3. *Traceability*: In the proposed scheme, vehicles communicate with each other using their temporary identities to preserve users' real identities, providing conditional privacy. Only the TA has the privilege to trace the real identities of vehicles and prevent malicious vehicles from participating in the network.

4. *Unlinkability*: Distrusted terminals cannot track the transmitter behaviours by determining the origins of two different signatures.

5. *Resistance to attacks*: The attacker's priority is to disrupt the network by applying the following common attacks:

   - *Replay attack*: The attacker retransmits previously captured data from the network after a period, which confuses the targeted terminal.
   - *Impersonation attack*: The attacker is trying to frame as a legitimate terminal and make the transmitted data appear as a normal flow of data.
   - *Modification attack*: The attacker modifies and subsequently retransmits broadcasted messages to the targeted terminal.
   - *Man-in-the-Middle (MITM) attack*: The attacker has the ability to modify and forward broadcasted messages among communicating terminals that assume they are in direct communication with each other.
   - *Sybil attack*: The attacker fabricates and broadcasts multiple messages with different identities in an attempt to affect the network's functionality.
   - *Denial-of-service attack*: This work considers the flooding type of DoS attack [122] in which the attacker tries to deteriorate the network's performance by overwhelming the targeted terminal with fake signatures $\sum \sigma_i$. In response, the targetted terminal verifies these signatures in a timely manner. Accordingly, a low-time cost verification process of σ_i allows for mitigating the impact of such an attack on the network.

3.2.2 Security and privacy evaluation of the ACPPA algorithm

In this part, the security strength of the ACPPA algorithm is proven in the ROM, in which the unforgeability of the signature generation stage is discussed against adversary $\mathscr{A}$ who is trying to impersonate V_2 by estimating $\langle TID_{V_2}, PID_{V_2}, A_2, T_2, \sigma_{V_2} \rangle$ under $RID_{V_2} : \langle r_{V_2}, PK_{V_2}, PK_{V_2,TA} \rangle$. The hardness of the signature generation stage depends on three cryptographic mathematical problems represented in the following definitions.

1. **Definition 1.** The elliptic curve discrete logarithm problem (ECDLP): Given $\langle a, b, P, p, q, \mathbb{G} \rangle$ and $Q = \gamma.P$, output $\gamma \in Z_q^*$.
2. **Definition 2.** Hashing problem: Given s', in which $s' = H_1(x)$, output $x \in \mathbb{G}$.
3. **Definition 3.** The hash message authentication code (HMAC) problem: Given h', in which $h' = HMAC_{key}(x)$, output $x \in \{0,1\}^*$ under $key \in \mathbb{G}$.

The signature generation stage is $(\tau_{\text{Sig.Gen}}, q_{ID}, q_s, \varepsilon_{\text{Sig.Gen}})$ existentially unforgeable against identity and adaptive chosen message attacks in the ROM as

$$\varepsilon_{\text{Sig.Gen}} \geq \varepsilon \left(1 - \frac{q_{ID}^2 q_s^2}{|N_1|\,|N_2|}\right), \tau_{\text{Sig.Gen}} = (6.q_{ID} + q_S)\, T_m \tag{3.16}$$

where T_m is the run time of scalar multiplication, q_{ID} and q_s are the number of queries to oracles $H_1(.)$ and $HMAC_{key}(.)$, respectively, and $\varepsilon_{\text{Sig.Gen}}$ and $\tau_{\text{Sig.Gen}}$ are the probability and time for adversary $\mathscr{A}$ to generate a non-trivial forgery (the proof of (3.16) is derived in the Appendix A). The following proves that the ACPPA algorithm meets the mentioned design goals.

1. *Privacy preservation and identity anonymity*: The real identities RID_{V_i} of the communicating terminals are preserved from adversary $\mathscr{A}$ as the authentication process depends on exchanging the pseudo-identities $PID_{V_i} = \left\{PID_i^1, PID_i^2\right\}$ for $PID_i^1 = \alpha_i.PK_{V_i}$ and $PID_i^2 = RID_{V_i} \oplus H_1\left(\alpha_i \cdot PK_{V_i,TA}\right)$, which means that the attacker needs to compute $\alpha_i.PK_{V_{i,TA}} = \alpha_i.r_{V_i}.\beta.P$ from $PID_i^1 = \alpha_i.PK_{V_i} = \alpha_i.r_{V_i}.P$. Since the tracking phase depends on the knowledge of TA's master key β, $\mathscr{A}$ has no chance to track or identify vehicles' real identities, providing conditional privacy preservation.

2. *Non-Repudiation*: Each side of the communicating terminals cannot deny its authorship of the generated signatures because the TID_{V_i} and PID_{V_i} can only be computed based on the RID_{V_i}, PK_{V_i}, and $PK_{V_i,TA}$ which are stored in V_i's TPD and only accessible by the vehicle itself.

3. *Traceability and revocation*: Only the TA can check the validity of PID_{V_i}, estimate the RID_{V_i} of the misbehaving vehicle, and revoke it based on TA's master key β as clarified in the real identity tracking phase.

4. *Unlinkability*: For each vehicle V_j communicates with V_i, V_i's signatures are generated with different TID_{V_i} and PID_{V_i} whose values are evaluated based on randomly selected parameters a_j and $\alpha_i \in Z_q^*$ that are dynamically updated. Accordingly, it is hard for $\mathscr{A}$ to determine the origins of two randomly captured signatures from the same vehicle.

5. *Attacks resistance*: The proposed algorithm is shown to be resilient to common types of attacks, e.g., replay, impersonation, modification, MITM, Sybil, and DoS attacks as follows:

   - *Resistance to replay attack*: ACPPA algorithm resists replay attack as each terminal checks the freshness of each generated signature σ_{V_i} based on the attached timestamp T_i by testing whether $T_r - T_i \leq T_\Delta$ holds or not. In addition, the randomly generated variables a_j, a_i, and $\alpha_i \in Z_q^*$ are frequently updated to avoid such attacks as the signature generation process depends on the current parameters. These reasons make the ACPPA algorithm immune to replay attacks.
   - *Resistance to impersonation attack*: In this attack, an adversary $\mathscr{A}$ tries to masquerade as a legitimate vehicle V_i by creating a valid signature $\langle TID_{V_i}, PID_{V_i}, A_i, T_i, \sigma_{V_i}\rangle$. To succeed, $\mathscr{A}$ must forge the signature σ_{V_i}, which is existentially unforgeable against identity and adaptive chosen message attacks proved in the ROM. Thus, ACPPA is resilient to such attacks.

- *Resistance to modification attack*: The integrity of the received signature can be easily detected by estimating $\sigma'_{V_i} = HMAC_{SV_{V_{i-j}}}(TID_{V_i} \| PID_{V_i} \| T_i)$, in which, the session key SK_{i-j} is computed using Diffie-Hellman key exchanging protocol under the difficulty of solving the ECDLP. After that, the verifier checks whether $\sigma'_{V_i} \stackrel{?}{=} \sigma_{V_i}$ holds. If not, such an attack is detected, and the received signature is rejected.
- *Resistance to MITM attack*: To avoid this attack, the recipient ensures that the message sender is a legitimate party. The proposed ACPPA algorithm uses the temporary identity TID_{V_j} to identify the sender's legitimacy, computed based on the session parameter $a_i \in Z_q^*$. To execute this attack, an adversary $\mathscr{A}$ must forge a valid signature, which is existentially unforgeable against identity and adaptive chosen message attacks proved in the ROM. Thus, this attack is prevented.
- *Resistance to Sybil attack*: An internal attacker (an authenticated user from inside the network who is aware of the network configuration) has multiple-fabricated *PID*s that can be used singularly or simultaneously to masquerade multiple vehicles. This type of attack is common in many contributed VANETs' signatures-based techniques. In the ACPPA scheme, a unique shared key is obtained using a location-dependent channel-based secret key extraction algorithm (S2). This means that there is no opportunity for a single vehicle in the network to extract more than a shared key within T_c. In other words, whatever the number of the generated *PID*s, there is no chance of generating more than one shared key between two terminals within T_c that varies at different terminal speeds, mitigating the effect of such an attack on the network.
- *Resistance to DoS attack*: Considering communication availability and since this study aims to reduce the computation and communication overheads, this study examines the common flooding type of DoS attack [123] on S1. In the latter (S1), the recipient verifies the sender's legitimacy and eventually discards fake requests (see Fig. 3.1), preventing $\mathscr{A}$ from proceeding to S2. In this attack, an adversary $\mathscr{A}$ attempts to flood V_j with several requests in the form of $\langle A_i, T_i \rangle$ or flood V_i with signatures in the form of $\langle TID_{V_j}, PID_{V_j}, A_j, T_j, \sigma_{V_j} \rangle$. In both cases, the targeted terminal replies by signing or verifying *HMAC*-based signatures in which the computation overhead of the $HMAC_{key}(x)$ process is low within a few μsecs, which reduces the effect of DoS attacks on the network compared to the computationally-expensive elliptic curve digital signature algorithm (ECDSA)-based signatures.

3.2.3 Security evaluation of the PHY challenge-response algorithm

In this subsection, the security strength of the PHY challenge-response algorithm is evaluated under different adversarial scenarios by considering Eve as a passive and active attacker who

knows the algorithm's schematic diagram. Eve is a passive attacker who can eavesdrop on the challenge signal and its related response and try to deduce any helpful information about the extracted shared key. However, the key cannot be deduced easily from the PHY response for two main reasons:

a) The high sensitivity of the channel multipath components to the distance between the communicating terminals, which makes it hard to differentiate between the initial signal's random phases θ_i and channel-phase response ξ_i.

b) According to the Avalanche effect [124]; By considering the PHY response generation process as a separate cryptographic operation $R(.)$ with input $I = (\theta_i, \xi_i)$ and output $O \leftarrow R(I); R(.)$ depends on the phase difference operation $\Delta\widehat{\psi}_{in}$ in (3.6), in which, Bob's random choice of the subcarrier index $n \in [1, N]$ denotes different output O under the same input I with probability $1/N$.

For these reasons, it is hard for Eve to estimate sensible information about the extracted key. Thus, by considering Eve as an active attacker, three primary potential attacks can be constructed in this scenario: replay, impersonation, and modification attacks.

1. *Resistance to impersonation attack*: Under this attack, Eve attempts to impersonate Alice or Bob. Suppose Eve is trying to impersonate Bob by generating a valid response. In that case, she possesses zero information about the extracted shared key and the correct session key $SK_{V_{i-j}}(TS_L)$ and has no chance to pass the authentication process successfully. If Eve is trying to impersonate Alice by sending a challenge signal to Bob, she can barely succeed to drive Bob's authentication key k_b. However, Eve cannot estimate or predict the upcoming $SK_{V_{i-j}}(TS_{L+1})$ to generate a correct response signal at TS_{L+1}. In addition, she cannot pass the mutual authentication process as she knows nothing about the other part of the extracted key k_a.

2. *Resistance to replay attack*: Eve can capture the transmitted signal from a legitimate terminal at time t and retransmit it back at time $t + \Delta t$. The replayed signal can be the challenge signal as case 1 or the response signal as case 2. In case 1, the challenge signal can be treated as an impersonation attack when Eve is trying to impersonate Alice. She has no opportunity to estimate the subsequent $SK_{V_{i-j}}(TS_{L+1})$ to generate a correct PHY response. In case 2, it depends on Δt. For $\Delta t > T_c$, the attack can easily be detected as the challenge signal varies over time; and the decision rule depends on the phase of the current challenge signal, while for $\Delta t \leq T_c$, Eve has no chance of success due to the small correlation coefficient of channel-phase responses between the legitimate and wiretap channels.

3. *Resistance to modification attack*: Eve attempts to alter the message contents. In that case, such an attack can easily be detected, and the altered message is rejected due to the lack

of reciprocity between the channel-phase response of the forward link $Ch_{A\rightarrow B}(t)$ and that of the reverse link $Ch_{A\leftarrow E\leftarrow B}(t+\Delta t)$ for $\Delta t \leq T_c$.

3.3 Performance evaluation

In this section, the performance of the PHY challenge-response algorithm is evaluated, as well as the computation and communication overheads, in order to elicit its advantages over existing alternatives.

3.3.1 Performance analysis of the PHY challenge-response algorithm

As part of this section, the detection probability of the re-authentication process is evaluated. Then, simulation and timing analyses are presented.

1. *Detection P_D vs. false alarm P_{FA} probabilities*: Estimating the probability density function (PDF) is necessary to investigate the probabilities of detection and false alarm under different threshold values. Based on the hypothesis testing problem in (3.12), at a certain threshold value T, P_D is the probability of the corresponding terminal is successfully authenticated as a legitimate party, while P_{FA} is the probability of a third party being authenticated as an authorised terminal. By deriving the cumulative distribution function (CDF) from the PDF of both hypotheses, one can estimate the optimum value of T for an acceptable false alarm probability. According to the central limit theorem (CLT) [125], v in (3.12) is the circular variance of a specific number of $N \in \{64, 128, 256\}$ subcarriers that can be approximated as a normally distributed random variable with means $\mu_{H_{0,1}}$ and variances $\sigma^2_{H_{0,1}}$ for both hypotheses $H_{0,1}$.

$$\mu_{H_{0,1}} \triangleq E\left(v \mid H_{0,1}\right), \sigma^2_{H_{0,1}} \triangleq Var\left(v \mid H_{0,1}\right) \tag{3.17}$$

Thus, the PDF $\mathscr{F}(.)$ for both hypotheses $H_{0,1}$ can be formulated as

$$\mathscr{F}\left(x\right)\big|_{\mu_{H_{0,1}},\sigma^2_{H_{0,1}}} = \frac{1}{\sqrt{2\pi\sigma^2_{H_{0,1}}}} e^{-\left(x-\mu_{H_{0,1}}\right)^2/2\sigma^2_{H_{0,1}}} \tag{3.18}$$

Then, the CDF $\phi(.)$ for both hypotheses can be expressed as

$$\phi\left(x\right)\big|_{\mu_{H_{0,1}},\sigma^2_{H_{0,1}}} = \frac{1}{2}\left[1+\text{erf}\left(\frac{x-\mu_{H_{0,1}}}{\sqrt{2\sigma^2_{H_{0,1}}}}\right)\right] \tag{3.19}$$

where the error function $\text{erf}(z) = \frac{2}{\sqrt{\pi}}\int_0^z e^{-t^2}dt$. Successful authentication is estimated for $v \mid H_0 \leq T$, in which the threshold value T is obtained for acceptable probability of false

alarm $P_{FA} = \phi(T)\,|_{\mu_{H_1},\sigma^2_{H_1}} \leq \alpha$ as

$$\phi(T)\,|_{\mu_{H_1},\sigma^2_{H_1}} = \frac{1}{2}\left[1+\text{erf}\left(\frac{T-\mu_{H_1}}{\sqrt{2\sigma^2_{H_1}}}\right)\right] \leq \alpha \tag{3.20}$$

Then,

$$T = \underset{T'}{\arg\max}\ \ erf\left(\frac{T'-\mu_{H_1}}{\sqrt{2\sigma^2_{H_1}}}\right) \leq 2\alpha - 1 \tag{3.21}$$

Given T, the probability of detection can be estimated as

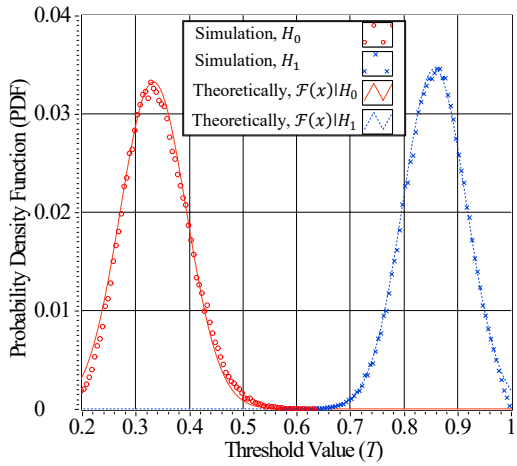
$$P_D = \phi(T)\,|_{\mu_{H_0},\sigma^2_{H_0}} \tag{3.22}$$

2. *Simulation results*: The empirical PDFs under both hypotheses $H_{0,1}$ are estimated through Monte-Carlo simulations. For better performance and since v in (3.12) obeys the CLT, the decision rule can be taken based on the mean value $\bar{v}$ of the last computed M estimates of v, decreasing the variances $\sigma^2_{H_0}$ and $\sigma^2_{H_1}$ of v's distributions in (3.18). Thus, the hypothesis testing problem can be expressed as

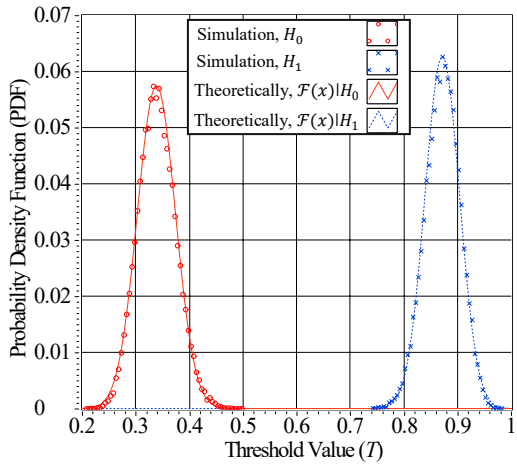
$$\bar{v} = \frac{1}{M}\sum_{\tau=0}^{M-1} v\left(t'_2-\tau\right) \underset{H_1}{\overset{H_0}{\lessgtr}} T, \text{ for } \begin{cases} H_0: & \phi_i = \mathscr{M}\left(k_{b,i}\right) \\ H_1: & \phi_i = \mathscr{M}\left(k_{\text{e},i}\right) \end{cases} \tag{3.23}$$

Note that, (3.12) equals (3.23) at $M = 1$. Fig. 3.7 presents the simulation results, and the theoretical normal distributions $\mathscr{F}(x) \mid H_0$ and $\mathscr{F}(x) \mid H_1$ of (3.18) for OFDM system with 64 subcarriers at SNR = 5 dB and $M = \{1,3\}$. As a proof of concept, Fig. 3.7(b) shows that the variance of $\bar{v}$'s distributions for both hypotheses is smaller than that of v's distributions in Fig. 3.7(a), enhancing the authentication performance. Moreover, from the same figure, the theoretical and simulation distributions are well matched, as well as $\mathscr{F}(x) \mid H_0$ is well separated from $\mathscr{F}(x) \mid H_1$, making it easier to choose the optimum threshold value T. As the SNR value decreases, the variance of $Var(\sum_{i=1}^{N} \angle(c_i))$ in (3.12) increases, leading to an increased mean value μ_{H_0} in (3.18). Consequently, this increase results in greater overlap between the distributions of both hypotheses ($\mathscr{F}(x) \mid H_0$, $\mathscr{F}(x) \mid H_1$), thereby increasing the false alarm probability $\phi(x\,|_{\mu_{H_1},\sigma^2_{H_1}})|_{x=T}$. Since the secret key extraction algorithm is executed without the information reconciliation and privacy amplification stages, the re-authentication process is performed based on the mutuality percentage $R(\%)$ of the extracted key between both terminals that can be expressed as

$$R(\%) = \left(1-\frac{BMR}{BGR}\right)\times 100 \tag{3.24}$$



(a) $\bar{v}$'s PDF for both hypotheses at $M = 1$ and SNR = 5 dB.



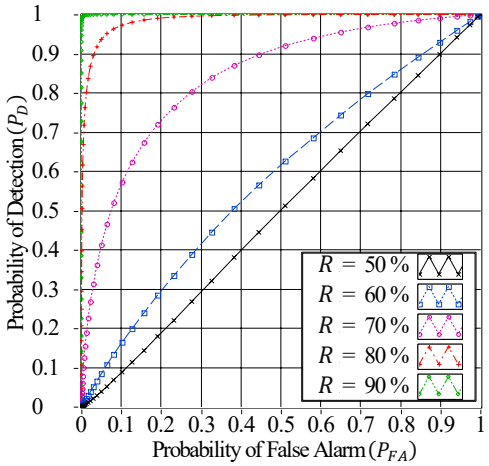
(b) $\bar{v}$'s PDF for both hypotheses at $M = 3$ and SNR = 5 dB.

Figure 3.7: Simulation and theoretical $\bar{v}$'s distribution for both hypotheses $H_{0,1}$ at $M = \{1,3\}$ and SNR = 5 dB. $\bar{v}$'s distribution is based on the mean value of v's last M estimates.

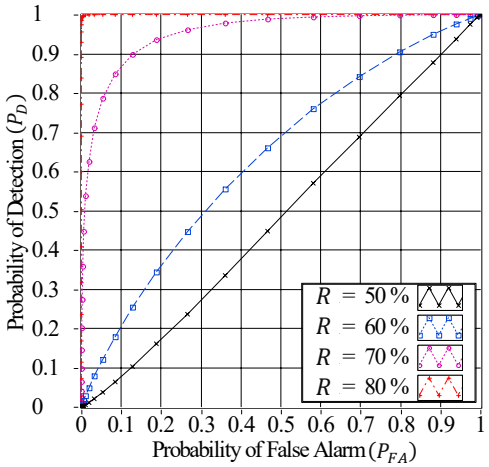
for

$$BGR = \frac{no.\ extracted\ bits}{no.\ channel\ samples}, \qquad BMR = \frac{no.\ erroneous\ bits}{no.\ channel\ samples} \tag{3.25}$$

where BGR and BMR are the bit generation rate and bit mismatch rate, respectively [115]. The independent mapping operation $\mathscr{M}(.)$ in (3.7) is a one-to-one mapping operation (each 2-bits for each subcarrier) which means that a sufficient number of matched bits in the extracted key from S2 is required to discriminate between Bob and Eve, avoiding false decision making. In other words, a sufficient mutuality, indicated by R in (3.24), must be assured to successfully authenticate the communicating vehicle. Fig. 3.8 shows the ROC curves (P_D versus P_{FA}) at different $R = \{50,60,70,80,90\}\%$ percentages and $M = \{1,3\}$. It can be noted from Fig. 3.8 that Alice and Bob must maintain over 80% and 70% mutuality of the shared key for $M = 1$ and 3, respectively, to achieve a high $P_D \geq 0.9$



(a) ROCs at $M = 1$ and and $R = \{50, 60, 70, 80, 90\}\%$.



(b) ROCs at $M = 3$ and $R = \{50, 60, 70, 80\}\%$.

Figure 3.8: P_D versus P_{FA} at SNR = 5 dB and $M = \{1, 3\}$ for different key mutuality percentages.

at $P_{FA} \leq 0.1$. This represents the trade-off relationship between the value of R and P_{FA}. As R increases, P_{FA} decreases, and conversely, as R decreases, P_{FA} increases.

In case of miss-detection $v \mid H_0 > T$, v in (3.12) is used as feedback to express the mutuality percentage R of the extracted key from S2. The value of $v \in [0, 1]$ in (3.12) is exploited to indicate the level of channel non-reciprocity, modelled through the standard deviation σ_c in (3.3). In [115], the perturb-observe algorithm is used to adjust the quantisation levels at different σ_c values by employing the cumulative distribution function and average fade duration statistics to determine the new threshold levels. Fig. 3.9 demonstrates the relationship between the expectation $E(v \mid R)$ at different $R = [50, 100]\%$ and SNR = {5, 10} dB. It can be noted that increasing the matching percentage R decreases the expectation $E(v \mid R)$ and vice versa. This proves the ability of the re-authentication process to be an alternative to the information reconciliation stage for the thresholds optimisation engine S2.3.

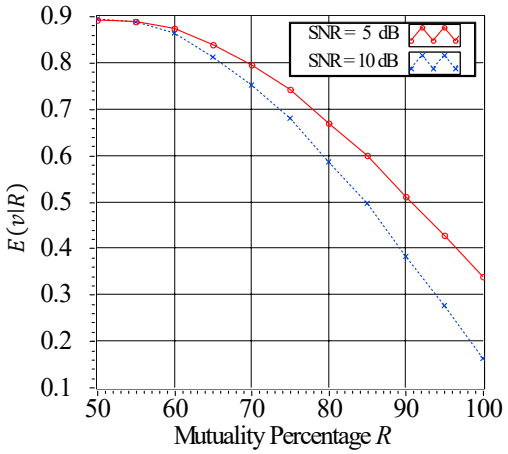


Figure 3.9: The key mutuality percentages $R(\%)$ versus the expectation value of v in (3.17) $E(v \mid R)$ at SNR = {5, 10} dB.

3. *Timing analysis*: In a real environment and the case of high-speed dynamic terminals, the time difference between transmitting the PHY challenge and receiving its related response must be less than the coherence time $(t_2 - t_0) < T_c$, which is the sum of the uplink $(t_1 - t_0)$ and the downlink $(t_2 - t_1')$ propagation time and the processing time of generating the PHY response $(t_1' - t_1)$, where t_0, t_1, t_1', and t_2 are the time of the signals in (3.4), (3.5), (3.8), and (3.9), respectively. For V2V communication, the DSRC bandwidth is assigned from 5.85 to 5.925 GHz [39]; thus, the maximum Doppler shift arising from the vehicles' and scatterers' speeds, $u_{V_{1(2)}}$ and u_S, is $f_{d(max)} = (u_{V_{1(max)}} + u_{V_{2(max)}} + 2u_{S(max)})/\lambda = 2360$ Hz [115], where $u_{V_{1(max)}} = u_{V_{2(max)}} = u_{S(max)} = 30\ m/s$ at 5.9 GHz carrier frequency. While the minimum coherence time is $T_{c(min)} = 1/f_{d(max)} = 0.4237\ msec$ [115]. The propagation time T_P is evaluated to be 10 μsec for 3 km distance between both terminals.

   Since v's distribution obeys the CLT [125], increasing the number of subcarriers N decreases the variances $\sigma^2_{H_0}$ and $\sigma^2_{H_1}$ of v's distribution in (3.18), improving the ROCs at small mutuality percentages, as demonstrated in Fig. 3.10. Table 3.2 presents the processing time of the PHY challenge $T_{PHY_{chang}}$, response $T_{PHY_{resp}}$, and verification $T_{PHY_{verf}}$ processes at different numbers of subcarriers $N = \{64, 128, 256\}$ subcarriers, which evaluated using Intel Core i7 $2.7-$GHz processor with 16.0 GB RAM. From Table 3.2, the estimated $T_{PHY_{resp}}$ is in the order of 0.39 *msec* at $N = 64$ subcarriers; thus, the total processing time $\left(2T_P + T_{PHY_{resp}}\right)$ is 0.41 $msec\,|_{N=64}$, smaller than $T_{c(min)}$. In addition, it can be noted from the same table that increasing the number of subcarriers (i.e., $N = \{128, 256\}$ subcarriers), increases the processing time $T_{PHY_{resp}}$, limiting the efficiency of the proposed algorithm at high-speed terminal conditions (i.e., $(2T_P + T_{PHY_{resp}}) = 0.843\ msec\,|_{N=128} = 1.74\ msec\,|_{N=256} > T_{c(min)}$). It is considered a tradeoff between high ROCs at low mutuality percentages and that at high-speed terminals.

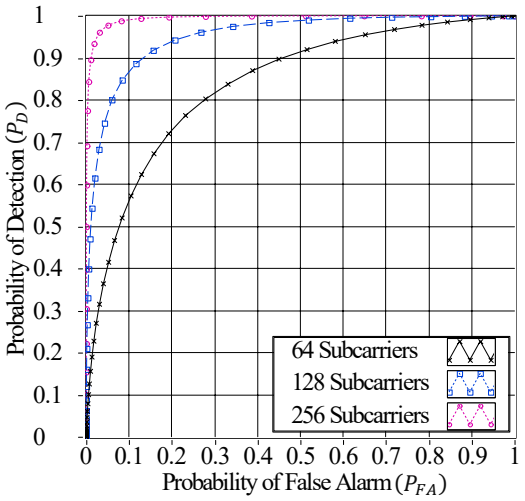


Figure 3.10: P_D versus P_{FA} at $R = 70\%$, $M = 1$, SNR = 5 dB, and number of subcarriers $N = \{64, 128, 256\}$ subcarriers.

Table 3.2: Computational overhead of the PHY challenge-response algorithm in *msec*

Execution Time	***N*=64**	***N*=128**	***N*=256**
Challenge $T_{PHY_{chang}}$	0.562	1.011	2.053
Response $T_{PHY_{resp}}$	0.39	0.823	1.72
Verification $T_{PHY_{verf}}$	0.125	0.291	0.469

3.3.2 Comparison of computation and communication overheads

Computation and communication complexities are important aspects to be considered when evaluating system performance. Table 3.3 compares the computation and communication overheads for verifying and sending n signatures from a single vehicle using the proposed scheme, identity-based message authentication scheme using proxy vehicles (ID-MAP) [44], CPPA [47], and new and efficient RSU based authentication (NERA) [45]. The following time quantities, $T_m, T_e, T_{M\to P}, T_{HMAC}$, and $T_{PHY_{verf}}$, represent the time consumed by scalar multiplication of the ECC, bilinear pairing, map-to-point hashing, hash message authentication code, and PHY-layer verification (S3.4), respectively. Furthermore, Table 3.3 classifies the performance metrics of each scheme according to the classification represented in Table 2.1.

1. *Computation overhead analysis*: This part demonstrates the computational comparison in detail. For an accurate computational evaluation, in Table 3.4, the execution time of multiple cryptographic operations over different curve parameters is computed in [55] by using Intel Core i7 and the widely used MIRACL cryptographic library [126]. In the proposed scheme, the time consumed for verifying n received signatures from a single vehicle is $T_m + T_{HMAC} + nT_{PHY_{verf}}$, in which $T_m + T_{HMAC}$ is the running time for the signature verification stage (S1.3.3) at the first time slot and $nT_{PHY_{verf}}$ for the PHY-layer verification

Table 3.3: Computation and communication overheads of verifying and distributing n signatures

Scheme	Computation overhead at the		Classification based on Table 2.1	Communication overhead at the		Classification based on Table 2.1
	proxy vehicle	endpoint terminal		proxy vehicle	endpoint terminal	
ID-MAP	$(d+6)T_m$	$5\lceil\frac{n}{d}\rceil T_m$	Low (endpoint)	$204d$	$184\lceil\frac{n}{d}\rceil + 124n$	High (endpoint)
CPPA	–	$(n+2)T_m$	Low	–	$107n$	High
NERA	–	$3T_e + nT_m + nT_{M\to P}$	Medium	–	$62n$	Medium
Proposed	–	$T_m + T_{HMAC} + nT_{PHY_{verf}}$	Low	–	$176 + 58.5n$	Medium

Table 3.4: Computational overhead of different cryptographic operations in *msec* [55]

Definition of the operation	Symbol	Run time
Scalar multiplication of the ECC in $\mathbb{G}$	T_m	0.442
Point addition of the ECC in $\mathbb{G}$	T_a	0.0018
Scalar multiplication of the BP in $\mathbb{G}_1$	T_{sm-BP}	1.709
Point addition of the BP in $\mathbb{G}_1$	T_{pa-BP}	0.0071
One-way hash function operation	T_h	0.0001
The map-to-point hashing operation in $\mathbb{G}_1$	$T_{M\to P}$	4.406
Bilinear Tate pairing operation in $\mathbb{G}_1$	T_e	4.211

(S3.4) of the subsequent n received PHY-responses. In ID-MAP [44], the verification process at the side of the proxy vehicle costs about $(d+6)T_m$ (for $d_{\max} = 300$ messages as recommended in [43]), while this value at the endpoint terminals is $5\lceil\frac{n}{d}\rceil T_m$. Furthermore, it can be noted from Table 3.3 that the verification processes in CPPA [47] and NERA [45] require about $(n+2)T_m$ and $3T_e + nT_m + nT_{M\to P}$, respectively.

To verify 1000 subsequent signatures sent from a single vehicle, the time required for the verification process at the endpoint in the proposed scheme is 125.4 *msec* $[= T_m + T_{HMAC} + nT_{PHY_{verf}} = 0.44 + 0.0008 + (1000 \times 0.125)]$ for T_{HMAC} = 0.0008 *msec* and $T_{PHY_{verf}}$ = 0.125 *msec* of 64 subcarriers (Table 3.2), while this value in ID-MAP at $\lceil\frac{n}{d}\rceil$ proxy vehicles and the endpoint (RSU) are 135.2 *msec* $[= (d+6) \times T_m = 306 \times 0.44]$ and 8.84 *msec* $[= 5 \times \lceil\frac{n}{300}\rceil \times T_m = 5 \times \lceil\frac{1000}{300}\rceil \times 0.44]$, respectively. It can be noted that ID-MAP provides lower computational overhead at the RSU as an endpoint terminal than the proposed scheme, as shown in Fig. 3.11, whereas the latter provides a lower computational overhead than that of ID-MAP at the side of the proxy vehicles. However, if there are no existing proxy vehicles with enough computational resources, all the generated signatures will be singularly verified by the RSU with computational overhead equals 443 *msec* $[= (d+6) \times T_m = 1006 \times 0.44]$. The time required for the verification process in CPPA and NERA are 442.8 *msec* $[= (n+2) \times T_m = 1002 \times 0.44]$ and 4858 *msec* $[= 3T_e + nT_m + nT_{M\to P} = (3 \times 4.2) + (1000 \times 0.44) + (1000 \times 4.4)]$, respectively. It is proven that the proposed scheme is more computationally efficient than the mentioned signature-based schemes [47], [45], and [44] at the side of the proxy vehicle. Also, ap-

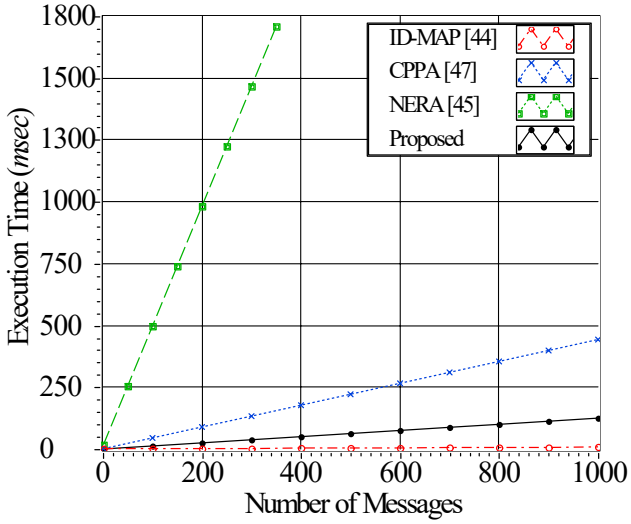


Figure 3.11: Computation overheads of verifying $n = 1000$ subsequent signatures transmitted from a single vehicle.

plying the proposed approach in V2I authentication using proxy vehicles as a future work can provide better performance than [44] at the RSU as an endpoint terminal.

2. *Communication overhead analysis*: This subsection evaluates and compares the proposed scheme's communication overhead. For the 80-bit security level of the ECC, $|q|$ and $|\mathbb{G}|$ are assumed to be 20 and 40 bytes, respectively. In addition, the length of the timestamp is assumed to be 4 bytes. The size of the communication request $\langle A_1, T_1 \rangle$ in (S1.3.1) is $40+4=44$ bytes, where $A_1 \in \mathbb{G}$. Also, the size of the generated signature $\langle TID_{V_2}, PID_{V_2}, \sigma_{V_2}, A_2, T_2 \rangle$ in (S1.3.2) is $40+60+32+40+4=176$ bytes long for Hash-SHA-1 and HMAC-SHA256 with 160 and 256 output-bits, respectively, and $(TID_{V_2}, PID_2^1, A_2) \in \mathbb{G}$.

   This part presents a detailed comparison of communication overheads. From Table 3.3, the overall communication overhead of the proposed scheme equals $176+58.5n$ bytes, which is the sum of that of the ACPPA signature at the first time slot (176 bytes), PHY communication request ($22.5n$ bytes), PHY response with a key length of 128 bits for 64 subcarriers ($16n$ bytes), and $SK_{V_{i-j}}(TS_L)$ of length ($20n$ bytes) at subsequent n time slots. From Table 3.3, the signature size sent to the proxy vehicles in ID-MAP [44] is $204d$, while this value at the endpoint (RSU) is $184\lceil \frac{n}{d} \rceil + 124n$. In CPPA [47] and NERA [45], the lengths of the generated signatures are $107n$ and $62n$, respectively. To transmit 1000 subsequent signatures from a single vehicle, the size of the transmitted signatures in the proposed scheme is 58674 bytes $[= 176 + (58.5 \times 1000)]$, while this value in ID-MAP [44] at the proxy vehicle, ID-MAP [44] at the endpoint terminal, CPPA [47], and NERA [45] are 61200 bytes $[= 204 \times 300]$ for $d = 300$, 124736 bytes $\left[= \left(184 \times \lceil \frac{1000}{300} \rceil\right) + (124 \times 1000)\right]$, 107000 bytes $[= 107 \times 1000]$, and 62000 bytes

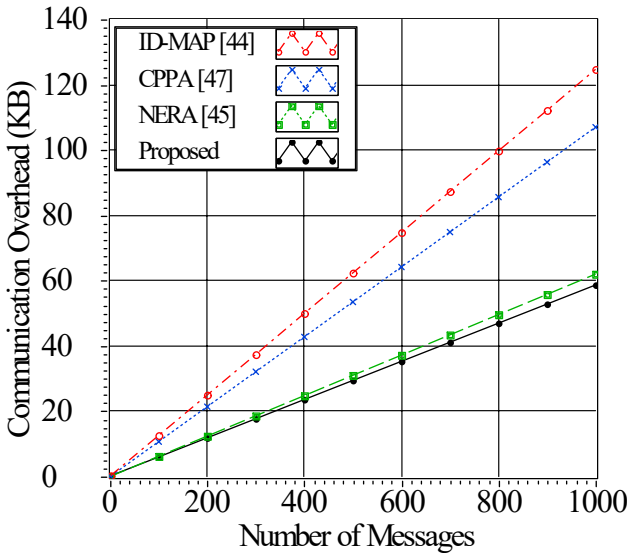


Figure 3.12: Communication overheads of transmitting $n = 1000$ subsequent signatures from a single vehicle.

$[= 62 \times 1000]$, respectively, as shown in Fig. 3.12. Based on the communication analysis, the proposed scheme has the lowest communication overhead for $n \geq 51$ compared to traditional methods.

Based on the overall computation and communication analyses, it is concluded that the proposed scheme outperforms CPPA [47]. Even though ID-MAP [44] is slightly more computationally efficient under a specific condition of proxy vehicles' existence, it has a significantly higher communication overhead in V2I communication, see Fig. 3.12. Furthermore, Fig. 3.11 shows that NERA [45] is significantly more computationally costly than all its competitors since it is bilinear pairing-based, despite having a slightly higher communication overhead than ours in Fig. 3.12. In this regard, the proposed scheme's lightweight re-authentication at the physical layer maintains a balance and optimises the trade-off between the computation and communication overheads, thereby enhancing network scalability. Aside from this, considering the channel's physical characteristics, the proposed scheme is more effective in detecting Sybil attacks and reducing the impact of the flooding type of DoS attacks on the network, as demonstrated in Section 3.2. Both of these attacks are common for signature-based authentication.

3.4 Summary

This chapter introduces a novel cross-layer authentication scheme for secure vehicular communication. In this scheme, a signature-based authentication algorithm is proposed to determine the legitimacy of the corresponding vehicle at the first time slot, employing the secret key generation algorithm in [115] for extracting a high entropy shared key with a minimum number of mismatched bits, avoiding the high communication overhead of the information reconciliation stage. The proposed scheme is the first authentication scheme that uses the PHY-layer challenge-response algorithm in VANETs applications, offering a high and successful authentication rate of up to 8000 $signatures/sec$. Simulation and implementation results proved the capability of the proposed algorithm to support a high probability of detection ≥ 0.9 at low false alarm probabilities ≤ 0.1 under small SNR values ≥ 5 dB, and key mutuality percentages $\geq 70\%$. According to the comprehensive comparison, the time required for verifying 1000 signatures in the proposed scheme is improved by $71\%, 72\%$, and 97% compared to ID-MAP [44] at the side of the proxy vehicle, CPPA [47], and NERA [45], respectively. As a further advantage, the proposed scheme can detect and mitigate Sybil and DoS attacks, which are common for crypto-based authentication approaches. The following chapter aims to explore the feasibility of developing a cross-layer approach that can effectively operate without relying on the typical assumption of a $\lambda/2$ spacing between network terminals. This investigation directly addresses the third question (Q_3) posed in subsection 1.4.2.

Chapter 4

Chaotic Map-based Key Extraction

The current state-of-the-art for secret key extraction has been developed, given that more than $\lambda/2$ separates the network terminals. Thus allowing for location decorrelation between legitimate and wiretapped channel responses, which can be specified by a zero-order Bessel function, where the first zero occurred at a $\lambda/2$ distance between the legitimate user and the adversary, see Fig. 2.3. In fact, a compromised RSU allows an attacker to gain access to the surrounding vehicles' secret features, making this condition unrealistic in V2I applications. Therefore, a Diffie-Hellman secret key extraction algorithm is designed, incorporating the Chebyshev mapping operation [127] for probing the channel. By doing so, the algorithm does not need a $\lambda/2$ distance between terminals, thus providing an efficient performance for V2X applications.

The following summarises the contributions of this chapter which are published in [20], fulfilling the outlined thesis objectives (1, 2, 3, 5, 6, 7) detailed in Subsection 1.4.3:

1. For key extraction, this chapter proposes a fast and secure key agreement technique for V2X applications. Accordingly, the unique cryptographic properties of the Chebyshev chaotic mapping and the spatially and temporally correlated channel phase responses within the coherence period are leveraged to design a PHY-layer key extraction algorithm for the OFDM-based DSRC system. A thresholding optimisation strategy is proposed to adjust the size of the thresholding region to the noisy channel phase estimation error in order to optimise the tradeoff between the bit generation rate (BGR) and the BMR.

2. For authentication, key-based and feature-tracking mechanisms are offered to allow for PHY-layer identity and integrity verifications, respectively, following the initial legitimacy detection using the upper layer's signature-based approaches. By creating a PHY-layer signature as an alternative to the existing crypto-based signatures, the corresponding terminal can verify the sender's legitimacy, employing the correlated channel attributes to check the integrity of the received data.

3. For validation, the proposed scheme's theoretical effectiveness is investigated as well as its security robustness is proven against passive and active attacks, including impersonation,

replaying, and modification.

4. Further, a Doppler emulator block is introduced to simulate the Doppler components of a three-dimensional (3D) scattering V2I scenario in the time domain, see Fig. 4.12. This block allows for empirically exploring the receiver operating characteristics of the PHY-layer authentication process at various speeds and SNRs for a realistic vehicular wireless channel using a software-defined radio platform, the universal software radio peripheral (USRP).

The following structure summarises the rest of the chapter. Section 4.1 introduces the preliminary knowledge, while the scheme model is presented in Section 4.2. Performance evaluation and threat modelling are given in Section 4.3. In Section 4.4, simulation and hardware implementation analyses are presented. Finally, Section 4.5 concludes this study.

4.1 Preliminaries and theoretical concepts

In this section, the network structure is described, followed by a review of VANETs' security and privacy requirements, along with several fundamental concepts used in this study.

4.1.1 Network configuration

VANETs are typically composed of the following entities, see Fig. 4.1.

1. *The TA*: As a trusted third party, the TA is responsible for initialising the public parameters and terminals' secret keys as well as preloading them onto registered vehicles and RSUs before joining the network. It is also capable of detecting and revoking misbehaving vehicles through their pseudo-identities.

2. *The RSUs*: Road infrastructures on both sides have wireless communication devices to connect with nearby vehicles and wired communications to connect with the TA. As a gateway, it serves primarily as a cooperative relay and broadcast point within VANETs.

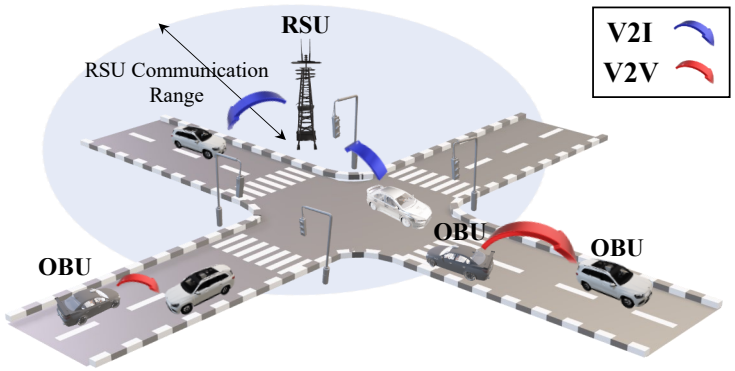


Figure 4.1: VANETs architecture for the PHY-layer key extraction and authentication methods.

3. *Vehicles' OBUs*: Vehicles are equipped with onboard units that provide wireless communication services and perform all computing functions. Further, each OBU has enough computational resources to generate large integer numbers that function as the vehicle's secret parameters.

The notations used in this chapter are listed in Table 4.1.

4.1.2 Security and privacy objectives

The proposed scheme complies with the requirements of VANETs, including security and privacy [128] as

1. *Message authentication*: Messages can be verified for their integrity by the recipient.

2. *Privacy preservation/identity anonymity*: Neither semi-trusted (RSUs) nor distrusted (adjacent vehicles) terminals can deduce identifiable information about the transmitter based on message contents.

Table 4.1: List of notations for the PHY-layer key extraction and authentication methods

Symbol	Definition
θ_i	The initial primitive root of the i^{th} subcarrier
n_i, m_i	Tx and Rx secret parameters
r	The quantisation order
$\mathscr{M}(.)$	The mapping function
$\mathscr{M}^{-1}(.)$	The inverse of the mapping function
Δt	The transmission time interval
h_i, ξ_i	Channel amplitude and phase responses
k	The symmetric key and equals $(k_a \| k_b)$
T_r	The signal receiving time
T_Δ	The timestamp expiry period, e.g., [00:00:59]
n_τ	The normalization coefficient
u_a	Vehicle's speed
δ	The distance driven by the vehicle within Δt
r_l	Angle's resolution value of the l^{th} scatterer
ϕ_a, ϕ_b	The mapped signatures
$\alpha_{a,l}, \beta_{a,l}$	Azimuth and elevation angles of departure
Δ_l	The step angle (rad) of the l^{th} scatterer
D_l	Direct distance between the Tx and scatterer
$d_{a,l}$	The Doppler component of the l^{th} scatterer
P_e, P_d	Probabilities of error and detection, respectively
P_{fa}	Probability of false alarm
a_1	The acceptable probability of error
a_2	The acceptable probability of false alarm

3. *Forward and backward secrecy* [129]: A malicious adversary can't discover the shared keys for previous and upcoming sessions based on that of the current session.

4. *Security strength*: The proposed scheme must be immune to typical adversarial attacks as follows [130].

   (a) *Immunity to impersonation*: An adversary tries to forge a trusted terminal's secret parameters to impersonate it. In this case, two potential scenarios are analysed in which the attacker is further or closer than $\lambda/2$ distance from the transmitter (Tx) or the receiver (Rx).

   (b) *Immunity to modification*: In this case, an adversary tampers with the transmitted messages by altering or modifying their contents.

   (c) *Immunity to replaying*: In this case, an adversary retransmits previously broadcasted messages after a period to deteriorate the network performance.

4.1.3 Mathematical foundations

The Chebyshev chaotic mapping finds applications in secure communications due to its complex behaviour, providing a basis for encryption algorithms. Its role extends to image encryption and data hiding techniques, leveraging its chaotic properties to enhance security in digital transmission. Furthermore, in computational mathematics, Chebyshev maps are employed in pseudo-random number generation algorithms, facilitating simulations in various fields such as economics, physics, and computer graphics, owing to their ability to generate diverse and seemingly random sequences. As part of the proposed scheme, the proposed secret key extraction algorithm takes advantage of the unique cryptographic properties of the Chebyshev chaotic mapping in terms of the significant computational complexities of solving the discrete logarithm and Diffie-Hellman problems to probe the channel. The following are some important theoretical concepts.

1. *Chebyshev chaotic mapping* [127]: $T_n(x)$ is a polynomial mapping function of input $x \in [-1,1]$ and output $y \in [-1,1]$ with a constant density $1/\left(\pi\sqrt{1-x^2}\right)$, and n is an integer number. The formulation of $T_n(x)$ is given by:

$$T_n(x) = \begin{cases} \cos\left(n \cdot \cos^{-1}(x)\right), & x \in [-1,1] \\ \cos(n \cdot \theta), & x = \cos(\theta) \end{cases} \tag{4.1}$$

   where $\theta \in [0,\pi]$.

2. In [131], the chaotic mapping operation in (4.1) is extended for x within the interval $(-\infty,+\infty)$. The extended map function has two important properties denoted by the following definitions:

(a) *Definition 1*: Given the two variables $x \in (-\infty, +\infty)$ and y, it is infeasible for an attacker to deduce the integer n, such that $T_n(x) \bmod p \equiv y$, where p is a large prime number. This problem is defined by the discrete logarithm problem (DLP).

(b) *Definition 2*: Given $x \in (-\infty, +\infty), T_n(x) \bmod p$, and $T_m(x) \bmod p$, the attacker has no chance to estimate $T_{nm}(x) \bmod p$, referred to as the Diffie-Hellman problem (DHP).

3. The Chebyshev mapping operation $T_n(x) : [-1,1] \rightarrow [-1,1]$ is employed in the key generation process by taking the inverse cosine of (4.1) and doubling the input range to get $T'_n(\theta) : [0,2\pi) \rightarrow [0,2\pi)$, denoted by

$$T'_n(\theta) = \begin{cases} n \cdot \theta \bmod p, & \theta \in [0,2\pi) \\ n \cdot \cos^{-1}(x) \bmod p, & x = \cos(\theta) \end{cases} \tag{4.2}$$

where $p = 2\pi$, and n is a large integer number of order $\lceil \log_2(n+1) \rceil$ bits.

4.2 Scheme modelling

This section describes the adaptive Diffie-Hellman secret key extraction process, and then the PHY-layer re-authentication process is presented in detail.

4.2.1 The Diffie-Hellman key extraction algorithm

As mentioned in Subsection 2.5.5, current secret key extraction techniques rely on the assumption that network terminals are separated by over $\lambda/2$ distance, allowing for location decorrelation between legitimate and wiretapped channel responses. However, in V2I communication scenarios, this assumption becomes impractical as attackers can position wireless cards near fixed RSU (i.e., $\leq \lambda/2$), leading to highly correlated channel features among surrounding vehicles, the RSU, and the attacker's device. Consequently, this compromises V2I application security by granting access to the secret channel features of neighbouring vehicles.

By employing the Chebyshev-based Diffie Hellman key exchanging protocol in the channel phase response-based key extraction process, the communicating terminals can obtain a high entropy secret key (k) in any wireless propagation environment (dynamic or even static). In addition, this mechanism allows the channel to be probed repeatedly within the same coherence period, thereby increasing the BGR. In general, the key generation process involves channel probing and thresholding, information reconciliation, and privacy amplification. The former includes exchanging probe signals between vehicles to obtain channel estimates, quantising these estimates, and converting them into bitstreams. The reconciliation stage corrects the mismatched bits. As for the privacy amplification stage, this further enhances the secrecy of the extracted bits

by hashing the corrected secret key. By applying (4.2) for an OFDM system of N subcarriers, (4.2) can be rewritten as

$$T'_{n_i}(\theta_i) = \begin{cases} n_i \cdot \theta_i \bmod p, & \theta_i \in [0, 2\pi) \\ n_i \cdot \cos^{-1}(x_i) \bmod p, & x_i = \cos(\theta_i) \end{cases} \tag{4.3}$$

where $i = 1, \ldots, N$ and $p = 2\pi$. Let θ_i be the initial primitive root of the i^{th} subcarrier and equals $2\pi/2^r$ for $r \in \{1,2,3\}$. The choice of the r value depends on the size of the phase-based thresholding region. In the polar coordinates, for $r = 3, e^{j\theta_i} = e^{j\pi/4}$ is considered as the generator (g) of the finite cyclic group $Z_{2^r=2^3}$ of order 8, defined as $Z_8 = \langle g \rangle = \{1, g, g^2, g^3, g^4, g^5, g^6, g^7\}$ in which $g^{n_i \bmod 8} = e^{jn_i.\theta_i \bmod 2\pi} = e^{jT'_{n_i}(\theta_i)}$ such that $g^{8 \bmod 8} = e^{j2\pi \bmod 2\pi} = 1$. Any element in the group can create its subgroup. For example, $Z_4 = \langle g^2 \rangle = \{1, g^2, g^4, g^6\}$ of order 4 and $Z_2 = \langle g^4 \rangle = \{1, g^4\}$ of order 2, as shown in Fig. 4.2. Based on the cyclic group theorem [132], it is computationally infeasible to determine: 1) $T'_{n_i m_i}(\theta_i)$, given $T'_{n_i}(\theta_i)$ and $T'_{m_i}(\theta_i)$, where n_i and m_i are large integer private parameters of the i^{th} subcarrier at the side of Alice and Bob, respectively; 2) the secret parameter n_i, given θ_i and $T'_{n_i}(\theta_i)$, so that the attacker needs $2^{\lceil \log_2(n_i+1) \rceil - r}$ trials to construct a brute-force attack and have a correct estimation, similarly for m_i. Fig. 4.3 shows the Diffie-Hellman channel probing mechanism between the vehicle $V_i(Alice)$ and the RSU $R_j(Bob)$. For simplicity, in this study, all formulas are denoted in the frequency domain. In a three-step process, the probing and thresholding stage is performed in the half-duplex mode as follows.

1. *Channel probing*: In this step, Alice initiates two subsequent OFDM symbols with random phases $T'_{2n_i}(\theta_i)$ and $T'_{n_i}(\theta_i)$ at time t_0 and $t_0 + \Delta t$ so that the transmitted signals can be

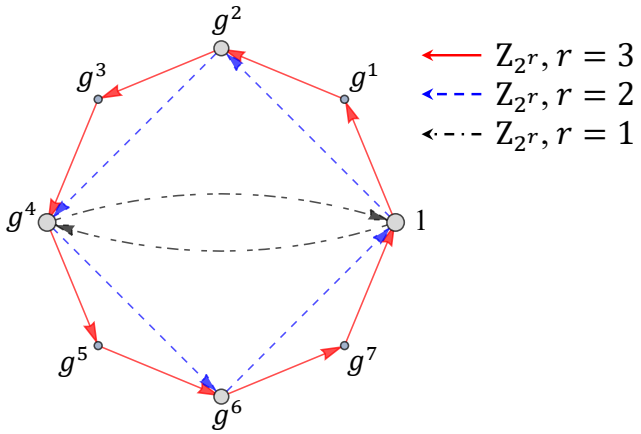


Figure 4.2: Cycle graph of order 2^r, for $r = 1, 2, 3$.

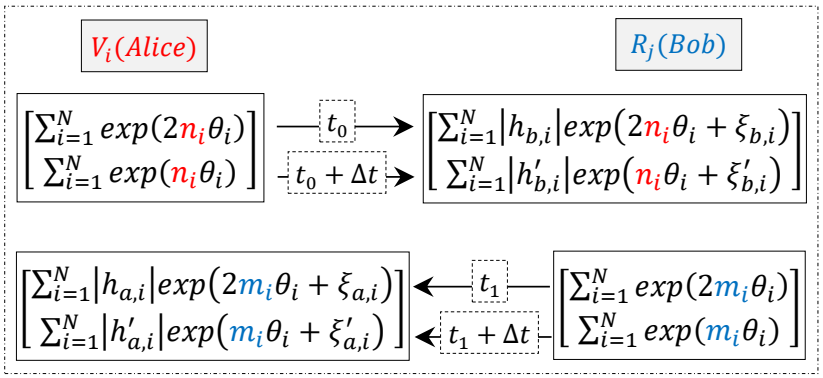


Figure 4.3: Diffie-Hellman probing step in a noiseless channel.

formulated as

$$\begin{aligned} s_a(t_0) &= \sum_{i=1}^{N} \sqrt{\frac{2E_S}{T}} e^{j(2n_i\theta_i)} \\ s_a(t_0+\Delta t) &= \sum_{i=1}^{N} \sqrt{\frac{2E_s}{T}} e^{j(n_i\theta_i)} \end{aligned} \tag{4.4}$$

where Δt is the transmission time interval $\ll T_c$. The received signal by Bob can be formulated as

$$\begin{aligned} r_b(t'_0) &= \sum_{i=1}^{N} \sqrt{\frac{2|h_i|^2 E_s}{T}} e^{j(2n_i\theta_i+\xi_{b,i})} + N_i \\ r_b(t'_0+\Delta t) &= \sum_{i=1}^{N} \sqrt{\frac{2|h_i|^2 E_s}{T}} e^{j(n_i\theta_i+\xi'_{b,i})+} N'_i \end{aligned} \tag{4.5}$$

where N_i and N'_i are complex additive Gaussian noises $\mathscr{CN}(0,2EN_0)$ with zero means and variances $2EN_0$, h_i and ξ_i are the Rayleigh fading channel responses of the i^{th} sub-carrier for the signal amplitude and phase, respectively, also ξ_i is a uniformly distributed random variable $U[0,2\pi)$ for $i = 1,\ldots,N$. In a similar way to (4.4), Bob replies by initiating two OFDM symbols with phases $T'_{2m_i}(\theta_i)$ and $T'_{m_i}(\theta_i)$ at time t_1 and $t_1+\Delta t$.

2. *Signal equalisation*: In this step, the received signals by Bob are equalised to compensate the channel phase responses $(\xi_{b,i},\xi'_{b,i})$ at time slots $(t'_0,t'_0+\Delta t)$ by calculating $e_b(t) = r_b(t'_0)\, r_b(t'_0+\Delta t)^*$ so that $\angle e_{b,i}(t)$ can be expressed as

$$\angle e_{b,i}(t) = n_i\theta_i + \varepsilon_{b,i} + (\omega_{b,i} - \omega'_{b,i}) \tag{4.6}$$

where $\varepsilon_{b,i} = \xi_{b,i} - \xi'_{b,i}$ is the error results from the imperfect channel reciprocity and $\omega_{b,i}$ and $\omega'_{b,i}$ are the phase estimation errors resulted from N_i and N'_i in (4.5). It is noteworthy that observations at different nodes or time slots are affected by independent realizations

of the noise [99]. With more samples in the observation, the estimation error becomes a zero-mean Gaussian random variable with variance $\sigma^2 \geq$ Cramer-Rao bounds of the phase variance estimation [133], so that the distribution of ω and ω' are $\mathcal{N}\left(0,\sigma^2\right)$. Thus, the distribution of $\angle e_{b,i}(t)$ in (4.6) is also normally $\mathcal{N}\left(n_i\theta_i+\ \varepsilon_{b,i},2\sigma^2\right)$ with mean $n_i\theta_i+\varepsilon_{b,i}$ and variance $2\sigma^2$. After that, Bob computes the round function of $\angle e_{b,i}(t)$ to get $\hat{T}'_{n_i}(\theta_i)$ as

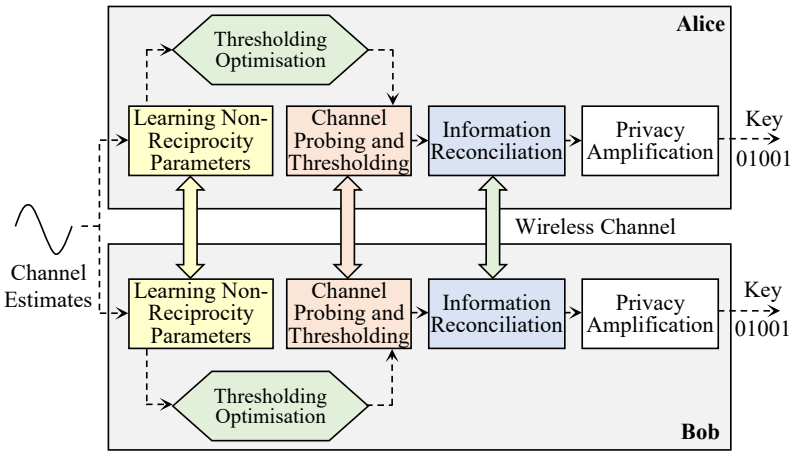
$$\begin{aligned}\hat{T}'_{n_i}(\theta_i) &= Round\left(\angle e_{b,i}(t)\right)\\ &= Round\left(n_i\theta_i+\varepsilon_{b,i}+\left(\omega_{b,i}-\omega'_{b,i}\right)\right)\end{aligned} \tag{4.7}$$

where $Round(x)$ is a function that rounds x to the nearest multiple of $2\pi/2^r$. Then, Bob obtains $T'_{n_im_i}(\theta_i)$ by computing $T'_{m_i}(\hat{T}'_{n_i}(\theta_i))$. It is important to perform the *Round* function before calculating $T'_{m_i}(\hat{T}'_{n_i}(\theta_i))$ to avoid the significant error caused from multiplying $\varepsilon_{b,i}$ by the large integer number m_i. The same process is performed at the side of Alice to get $T'_{n_im_i}(\theta_i)$ =$T'_{n_i}(\hat{T}'_{m_i}(\theta_i))$. The estimated $T'_{n_im_i}(\theta_i)$ at both sides are inversely mapped $\mathscr{M}^{-1}(.)$ to convert it into bitstreams k. The order of the inverse mapping operation depends on the r value. For simplicity, a Gray code $\mathscr{M}^{-1}(.)$ of order $r=2$ bits can be formulated as

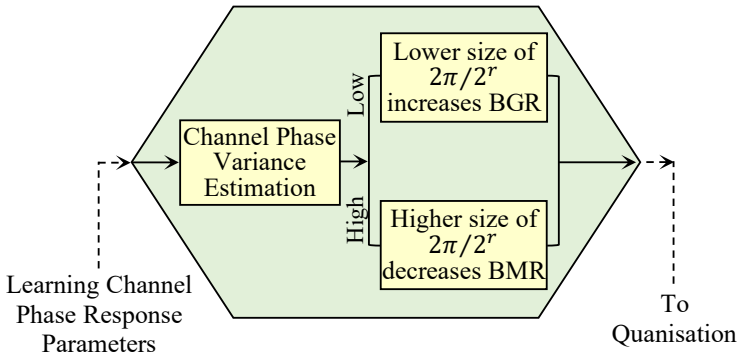
$$\mathscr{M}^{-1}\left(T'_{n_im_i}(\theta_i)\right)=\begin{cases}00 & T'_{n_im_i}(\theta_i)\in[-\frac{\pi}{4},\frac{\pi}{4})\\ 01 & T'_{n_im_i}(\theta_i)\in[\frac{\pi}{4},\frac{3\pi}{4})\\ 11 & T'_{n_im_i}(\theta_i)\in[\frac{3\pi}{4},-\frac{3\pi}{4})\\ 10 & T'_{n_im_i}(\theta_i)\in[-\frac{3\pi}{4},-\frac{\pi}{4})\end{cases}\quad \text{for } i=1,\ldots,N \tag{4.8}$$

Note that a Gray code spaces adjacent codes one hamming distance apart, thus reducing the BMR of the extracted keys.

3. *Thresholding optimisation*: In this step, the order of the thresholding region r is optimised for θ_i and $\mathscr{M}^{-1}(.)$ at both sides of the communicating terminals to provide a high secret bit generation rate (SBGR) for acceptable BMR, where SBGR is the number of correct/matched bits to the total number of channel samples. By adapting the size of the quantisation region $2\pi/2^r$ to different conditions of SNRs, the performance of the key extraction process will be optimised. A small quantisation region (i.e., a high order of r) denotes high BGR and BMR, and vice versa for large regions. For zero-value private keys $(n_i=m_i=0)$ and negligible non-reciprocity parameter $\left(\varepsilon_{a(b),i}\approx 0\right)$ due to the small transmission time interval ($\Delta t\approx 16\mu$s for 64 subcarriers and 16 cyclic prefix samples, in [87]), the distribution of the equalised phase $\angle e_{a(b)}(t)$ in (4.6) will be $\mathcal{N}\left(0,2\sigma^2\right)$. Similar to [134], both terminals can exchange m probing packets and have their channel phase estimates $\left(\widehat{\xi}_a^{t_a},\widehat{\xi}_b^{t_b}\right)$ at timestamps (t_a,t_b) to learn the noisy channel phase error distribution parameters, which equals $\mathcal{N}\left(\mu_{\xi,i},var_{\xi,i}=\sigma^2\right)$ for mean $\mu_{\xi,i}$ and variance



(a) Flowchart of the secret key extraction algorithm.



(b) Flowchart of the optimisation engine.

Figure 4.4: Modelling of the key extraction algorithm.

$var_{\xi,i}$ denoted by

$$\mu_{\xi,i} = \frac{1}{m}\sum_{x=1}^{m}\left(\widehat{\xi}^{t_{a(b)}}_{a(b)}(f_i)\right),$$
$$var_{\xi,i} = \frac{1}{m-1}\sum_{x=1}^{m}\left(\widehat{\xi}^{t_{a(b)}}_{a(b)}(f_i) - \mu_{\xi,i}\right)^2 \tag{4.9}$$

where m equals 100 probe packets, as recommended in [134]. By learning and doubling the estimated variance in (4.9), both terminals can learn the variance of $\angle e_{a(b)}(t)$'s distribution $2\sigma^2$ and agree on the quantisation order r. This method acts as a forward indicator for the channel probing and thresholding stage, see Fig. 4.4(a). Note that the quantisation region $2\pi/2^r$ is large for a large value of $var_{\xi,i}$, and vice versa for a small value, see Fig. 4.4(b).

Finally, the extracted key will be used for the PHY-layer re-authentication process discussed in the following subsection.

4.2.2 PHY-layer re-authentication algorithm

In the first transmission slot, mutual identity authentication between the communicating terminals is performed using conventional signature-based algorithms implemented at the upper layers of the protocols stack. This facilitates legitimacy detection, as well as the exchange of authenticated Chebyshev probing sequences used to extract the symmetric shared key k. This key allows for re-authenticating the received messages sent from the same transmitter for the OFDM system of N subcarriers. This study extends the work introduced in [19]. In a two-step process, the identity of the corresponding terminal is re-authenticated using a PHY-layer signature-based identity authentication mechanism (PHY-SIAM), while the integrity of the attached data packet is verified using a PHY-layer feature tracking mechanism (PHY-FTM). In this study, the assumption is made that the subcarriers are well separated to ensure independent fading. Fig. 4.5 depicts the structure of M OFDM symbols for $N = 64$ subcarriers in which $N/4$, and $3N/4$ subcarriers are used for channel probing and zero-padding, and signature/data transmission, respectively. The detailed steps are as follows.

1. *PHY-layer signature-based identity authentication mechanism* (*PHY-SIAM*): In this part, and after mutual identity verification, the receiver checks the sender's identity based on the extracted key k. This key is divided into two subkeys, k_a and k_b, with equal lengths, which are used to generate the PHY-layer signature of the attached data. The created signature is transmitted from the vehicle V_i to the RSU R_j within the same region along with its related data, as demonstrated in Fig. 4.6. In general, PHY-SIAM consists of three primary phases, i.e., initialisation, signature generation, and message verification.

   (a) *System initialisation*: In this phase, the TA generates the PHY-layer public parameters and preloads them into all registered network terminals. Accordingly, the system is initially configured as follows.

   - Mapping operation: A Gray coded 2-bit mapping function is used to map the input variable $K = \left\{\kappa_1\kappa_2, \ldots, \kappa_{(3N/2)-1}\kappa_{3N/2}\right\}$ to ϕ, such that $\mathscr{M}(K) \rightarrow \phi$ is designed as

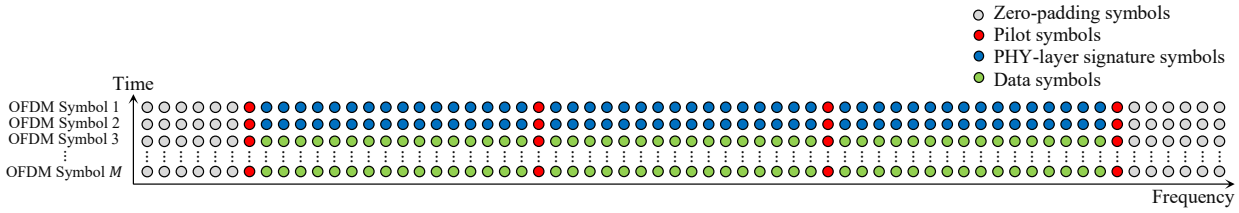


Figure 4.5: Symbols structure for OFDM system of 64 subcarriers.

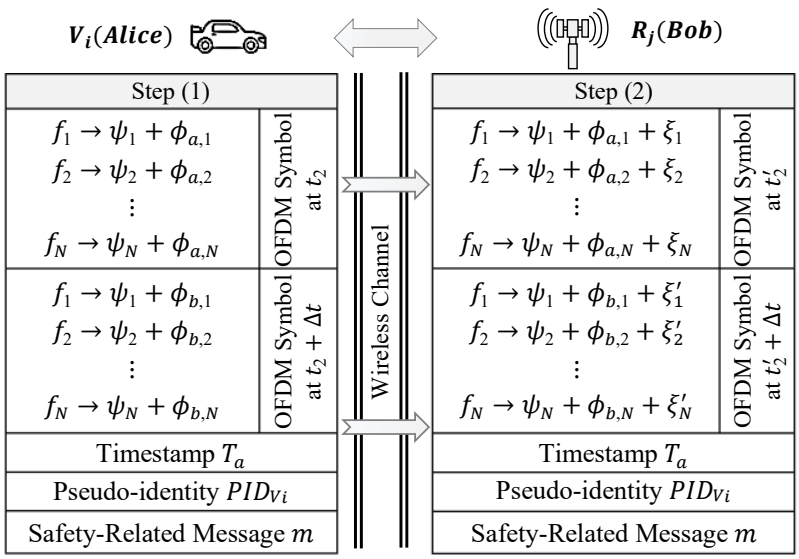


Figure 4.6: The PHY-layer identity re-authentication mechanism in a noiseless channel.

$$\phi_i = \mathscr{M}(K_i) = \begin{cases} 0 & K_i = [00] \\ \frac{\pi}{2} & K_i = [01] \\ \pi & K_i = [11] \\ \frac{3\pi}{2} & K_i = [10] \end{cases} \quad \text{for } i = 1, \ldots, 3N/4 \tag{4.10}$$

- Secure hash function H_1: $\{0,1\}^* \rightarrow \{0,1\}^{3N/2}$.
- Finally, the tuple $\langle H_1, \mathscr{M}(.)\rangle$ is preloaded into the network terminals during the registration.

(b) *Signature generation*: In this phase, each safety-related message m is attached with the sender's pseudo-identity PID_{V_i}, timestamp T_a, and two OFDM symbols. These symbols are collectively referred to as the PHY layer signature, which is generated in a two-stage process as follows.

- *Signature preparation*: In this stage, k_a is concatenated with the attached timestamp T_a, hashed, and mapped to obtain ϕ_a of the first OFDM symbol. T_a is defined here as a "nonce." Similarly, ϕ_b is obtained by using k_b for the second OFDM symbol. The following is the formulation of the mapping operations ϕ_a and ϕ_b.

$$\begin{aligned} \phi_a &= \mathscr{M}(H_1(k_a \| T_a)) \\ \phi_b &= \mathscr{M}(H_1(k_b \| T_a)) \end{aligned} \tag{4.11}$$

- *Signature generation*: In this stage, the mapped signatures are masked by uniformly distributed random phases $\psi_i \sim U[0, 2\pi)$ for $i = 1, \ldots, N$ with probability

density function $1/2\pi$. Afterwards, Alice initiates two subsequent signals, $s_a(t_2)$ and $s_a(t_2+\Delta t)$, with time difference Δt less than the coherence time, and frequencies $f_1,\ldots,f_N$. Then, Alice sends them to Bob in the form of

$$\begin{aligned} s_a(t_2) &= \sum_{i=1}^{N} \sqrt{\frac{2E_s}{T}} e^{j(\psi_i+\phi_{a,i})} \\ s_a(t_2+\Delta t) &= \sum_{i=1}^{N} \sqrt{\frac{2E_s}{T}} e^{j(\psi_i+\phi_{b,i})} \end{aligned} \tag{4.12}$$

so that the received signals by Bob are denoted by

$$\begin{aligned} r_b(t_2') &= \sum_{i=1}^{N} \sqrt{\frac{2|h_i|^2 E_s}{T}} e^{j(\psi_i+\phi_{a,i}+\xi_i)} + N_i \\ r_b(t_2'+\Delta t) &= \sum_{i=1}^{N} \sqrt{\frac{2|h_i|^2 E_s}{T}} e^{j(\psi_i+\phi_{b,i}+\xi_i')} + N_i' \end{aligned} \tag{4.13}$$

- *Message verification*: In this stage, avoiding replaying attacks, Bob verifies the validity of the attached timestamp T_a by checking if $T_r - T_a \leq T_\Delta$ holds or not. If holds, the received OFDM symbols are verified to avoid impersonation attacks using the symmetric key k and the attached timestamp T_a. In a similar way to (4.11), Bob computes the mapped signatures ϕ_a' and ϕ_b', and calculates $r_b'(t_2') = r_b(t_2')\,e^{-j\phi_a'}$ and $r_b'(t_2'+\Delta t) = r_b(t_2'+\Delta t)\,e^{-j\phi_b'}$. Note that ξ_i and ξ_i' in (4.13) are highly correlated for $\Delta t \leq T_c$. Thus, Bob verifies the sender's identity by computing the circular variance $c.var(.)$ of $\angle c(t) = r_b'(t_2')\,r_b'(t_2'+\Delta t)^*$ as

$$v = c.var\left(\sum_{i=1}^{N} \arctan\left(\frac{Im(c_i(t))}{Re(c_i(t))}\right)\right) \tag{4.14}$$

where the circular variance [120] $c.var(.)$ is given by

$$\begin{gathered} \alpha_i = \begin{pmatrix} \cos(\angle(c_i)) \\ \sin(\angle(c_i)) \end{pmatrix}, \bar{\alpha} = \frac{1}{N}\sum_{i=1}^{N} \alpha_i, \\ v = 1 - \|\bar{\alpha}\| \end{gathered} \tag{4.15}$$

where $\|.\|$ is the norm function. Suppose an impersonator, Eve, is attempting to masquerade as Bob. In this case, Eve uses a different key k_e to initiate a PHY layer

signature, which is considered a hypothesis testing problem, given that

$$v \underset{H_1}{\overset{H_0}{\lessgtr}} \tau_1, \text{ for } \begin{cases} H_0: & \phi_a' = \phi_a \& \phi_b' = \phi_b \\ H_1: & \phi_a' \neq \phi_a \& \phi_b' \neq \phi_b \end{cases} \tag{4.16}$$

2. *PHY-layer feature tracking mechanism* (*PHY-FTM*): After verifying the sender's identity, the recipient can verify the integrity of the received message in order to avoid modification attacks. In this context, Bob employs the received probe symbols of the i^{th} subcarrier in the j^{th} OFDM symbol for channel estimation, where $j = 1, \ldots, M$ data symbols. The channel observations vector $\bar{H}_j$ of the j^{th} OFDM symbol consists of all the channel estimates obtained from the i^{th} subcarriers that hold probe symbols (i.e., subcarriers highlighted in red in Fig. 4.5). To determine whether the received data is sent from the same source, the recipient compares the channel observation vector of the j^{th} symbol with that of the $(j-1)^{th}$ symbol, starting from the PHY-layer signature at $j = \{1,2\}$ to the M^{th} OFDM data symbol. In this study, the channel estimation is performed using the least square (LS) as well as the minimum mean-square error (MMSE) [135] methods. The integrity verification process can be characterised as a hypothesis testing problem in which H_0 indicates that all data packets are transmitted from the sender whose identity is verified using the proposed PHY-SIAM, otherwise H_1. The hypothesis testing of the normalised likelihood ratio test (LRT) can be represented as

$$\Lambda_{LRT} = \frac{n_{\tau_2} \left\| \bar{H}_j - \bar{H}_{j-1} \right\|^2}{\left\| \bar{H}_{j-1} \right\|^2} \text{ for } j = 2, \ldots, M,$$
$$\Lambda_{LRT} \underset{H_0}{\overset{H_1}{\lessgtr}} \tau_2 \tag{4.17}$$

where n_{τ_2} is the normalisation coefficient that makes the threshold value $\tau_2 \in [0,1]$. While the hypothesis testing of the sequential probability ratio test (SPRT) [136] is formulated as

$$\Lambda_j = \frac{n_{\tau_2} \left\| \bar{H}_{M-j+1} - \bar{H}_{M-j} \right\|^2}{\left\| \bar{H}_{M-j} \right\|^2} \text{ for } j = 1, \ldots, M-1,$$
$$\Lambda_{SPRT} = n_{\tau_3} \sum_{j=2}^{M} \Lambda_j, \Lambda_{SPRT} \underset{H_0}{\overset{H_1}{\lessgtr}} \tau_3 \tag{4.18}$$

where n_{τ_3} is the normalisation coefficient that makes the threshold value $\tau_3 \in [0,1]$. In SPRT-based hypothesis testing, the sum of the LRTs between the j^{th} and $(j-1)^{th}$ symbols

$\forall j \in [2, M]$ is compared with the threshold value τ_3 to make the decision rule, which can improve the detection rate compared to the simple LRT.

4.3 Performance evaluation and threat modelling

This section presents the theoretical analysis of the key extraction and re-authentication processes and then discusses in depth the security strength of the re-authentication algorithm.

4.3.1 Theoretical analysis of the key extraction algorithm

In order to evaluate the key extraction performance, it is necessary to calculate the probability of error/mismatching P_e. In this context and since the distribution of the equalised phase $\angle e_{a(b)}(t)$ in (4.6) is normally distributed $\mathcal{N}(T'_{n_i(m_i)}(\theta_i) = n_i(m_i)\theta_i, 2\sigma^2)$ at negligible $\varepsilon_{a(b),i}$, the cumulative distribution function $\phi(.)$ can be formulated as

$$\phi(x) = \frac{1}{2}\left[1 + \text{erf}\left(\frac{x - T'_{n_i(m_i)}(\theta_i)}{2\sigma}\right)\right] \tag{4.19}$$

where the error function is given by $\text{erf}(z) = \frac{2}{\sqrt{\pi}}\int_0^z e^{-t^2}dt$. Thus, P_e is the probability that $\angle e_{a(b)}(t) \notin [T'_{n_i(m_i)}(\theta_i) - \pi/2^r, T'_{n_i(m_i)}(\theta_i) + \pi/2^r)$, which makes the output of the *Round* (.) function in (4.7) doesn't equal $T'_{n_i(m_i)}(\theta_i)$, i.e., $\hat{T}'_{n_i(m_i)}(\theta_i) \neq T'_{n_i(m_i)}(\theta_i)$. Finally, P_e is given by

$$P_e = 2\phi\left(T'_{n_i(m_i)}(\theta_i) - \frac{\pi}{2^r}\right), r \in \{1, 2, 3\} \tag{4.20}$$

For an acceptable probability of error less than or equal to the scalar value a_1, r can be calculated by both terminals to optimise the size of the thresholding region $2\pi/2^r$ based on the estimated learned parameter $var_\xi = \sigma^2$ in (4.9) as

$$x = \arg\max_{x'} \text{erf}\left(\frac{x' - T'_{n_i(m_i)}(\theta_i)}{2\sigma}\right) \leq a_1 - 1 \tag{4.21}$$

Given x, r can be obtained as

$$r = \arg\max_{r'} 2^{r'} \leq \frac{\pi}{x} \quad \text{for } r' = 1, 2, 3 \tag{4.22}$$

4.3.2 Detection vs. false alarm probabilities of re-authentication

As part of the performance evaluation, it is important to examine the receiver operating characteristics of the identity verification process. ROC is a measurement of the detection probability P_d at different values of false alarm probabilities P_{fa}. To determine the ROC, the prob-

ability density function must be investigated. The estimated differential baseband signal $c_i = r'_b(t'_2)\, r'_b(t'_2+\Delta t)^*$ of the i^{th} subcarrier can be simplified as

$$c_i = 2h_i h_i^* E e^{j(\varepsilon_{e,i})} + h_i^* N_i + h_i N_i' = X + jY \tag{4.23}$$

where $\varepsilon_{e,i} = \varepsilon_{a,i} - \varepsilon_{b,i}$ for $\varepsilon_{a,i} = \phi_{a,i} - \phi'_{a,i}$ and $\varepsilon_{b,i} = \phi_{b,i} - \phi'_{b,i}$, and $h_i = |h_i| e^{j\xi_i}$. Considering Alice is communicating with Bob (i.e., H_0), which makes $\varepsilon_{e,i}$ equals zero, so that the real and imaginary parts of (4.23) can be formulated as

$$\begin{aligned} X &= 2|h_i|^2 E + Re\left(h_i^* N_i + h_i N_i'\right), \\ Y &= Im\left(h_i^* N_i + h_i N_i'\right) \end{aligned} \tag{4.24}$$

where the expectation $E(X) = 2|h_i|^2 E = \mu$ and $E(Y) = 0$ while the variance $var(X) = var(Y) = 4EN_0 h_i^2 \cong \sigma_0^2$. Similar to [105], the joint probability density function of X and Y can be expressed as

$$P\left(x, y|_{H_0}\right) = \frac{1}{2\pi\sigma_0^2} e^{\frac{-\left[(x-\mu)^2+y^2\right]}{2\sigma_0^2}} \tag{4.25}$$

By changing the variables $R = \sqrt{X^2+Y^2}$ and $\Theta = \arctan(Y/X)$. The joint probability density function of (4.25) yields to

$$P\left(R, \Theta|_{H_0}\right) = \frac{R}{2\pi\sigma_0^2} e^{\frac{\left[2\mu R\cos\Theta - R^2 - \mu^2\right]}{2\sigma_0^2}} \tag{4.26}$$

By integrating (4.26) over $R \in [0,\infty)$ [105], (4.26) can be simplified as

$$P(\Theta \mid \Gamma) = \frac{1}{2\pi} e^{-\Gamma} + \frac{1}{\sqrt{\pi}} (\sqrt{\Gamma}\cos\Theta) \cdot e^{-\Gamma\sin^2\Theta} [1 - \mathbb{Q}(\sqrt{2\Gamma}\cos\Theta)] \tag{4.27}$$

where

$$\begin{aligned} \Gamma &= \frac{h_i^2}{2} \cdot \frac{E_S}{N_0}, \\ \mathbb{Q}(x) &= \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-t^2/2} dt \end{aligned} \tag{4.28}$$

Fig. 4.7 shows $P(\Theta)|_{H_0}$ parametrised by different Γ values. It can be observed that increasing Γ value decreases the variance of $P(\Theta)$, and vice versa. Since v in (4.14) represents the variance of a restricted number of N subcarriers, its distribution is normally with mean equals to the variance of $P(\Theta \mid \Gamma)$ and variance depends on the N value used to estimate v in (4.14), following the central limit theorem. Thus v's normal distribution for both hypotheses $H_{0,1}$ is represented

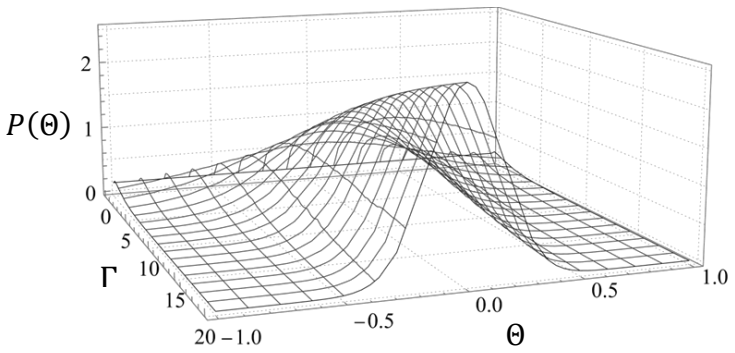


Figure 4.7: The $P(\Theta)|_{H_0}$ parametrised by different Γ values.

by mean $\mu_{H_{0,1}} \cong E(v \mid H_{0,1})$ and variance $\sigma^2_{H_{0,1}} \cong var(v \mid H_{0,1})$, which can be formulated as

$$\mathscr{F}\left(x \mid \mu_{H_i}, \sigma^2_{H_i}\right) = \frac{1}{\sqrt{2\pi\sigma^2_{H_i}}} e^{-\left(x-\mu_{H_i}\right)^2/2\sigma^2_{H_i}}, i = 0,1 \tag{4.29}$$

with CDF equals

$$\phi\left(x \mid \mu_{H_i}, \sigma^2_{H_i}\right) = \frac{1}{2}\left[1 + \operatorname{erf}\left(\frac{x-\mu_{H_i}}{\sqrt{2\sigma^2_{H_i}}}\right)\right], i = 0,1 \tag{4.30}$$

In this study, the acceptable performance of re-authentication is referred to as the successful probability of detection $\phi(x \mid \mu_{H_0}, \sigma^2_{H_0})\Big|_{x=\tau_1}$ for an acceptable false alarm $\phi(x \mid \mu_{H_1}, \sigma^2_{H_1})\big|_{x=\tau_1}$ less than or equal to the scalar value a_2. Thus, the threshold value τ_1 is obtained as

$$\tau_1 = \arg\max_{\tau_1'} \operatorname{erf}\left(\frac{\tau_1' - \mu_{H_1}}{\sqrt{2\sigma^2_{H_1}}}\right) \leq 2a_2 - 1 \tag{4.31}$$

4.3.3 Security analysis of the re-authentication algorithm

As a part of this section, the security strength is evaluated against passive and active attacks. Consider Eve as a passive attacker who eavesdrops on the broadcasted messages and their associated PHY-layer signatures. In this case, there is no way for Eve to derive the symmetric key k from the message contents for two primary reasons: 1) By considering the signature generation stage as a single cryptographic process $C(.)$ with input $I(k, T_a, \psi_i, \xi_i)$ and output $O \leftarrow C(I); C(.)$ depends on the current timestamp T_a, which denotes different output O given the same input variables $(k, \psi_i, \xi_i); C(.)$ depends on the randomly varying ψ_i, which masks the phase response ξ_i and the mapped signatures $\phi_{a(b)}$ thus Eve cannot differentiate between $\phi_{a(b)}$, ψ_i, and ξ_i. 2) For $y \leftarrow H_1(x)$, it is hard for Eve to deduce the input variable $x : \{0,1\}^*$ given the hashed variable $y : \{0,1\}^{3N/2}$. Therefore, Eve is considered an active attacker who can impersonate a legitimate terminal, replay a previously captured message, or alter message contents.

1. *Impersonation attacks*: In this attack, Eve tries to impersonate Alice by creating a valid PHY-layer signature. In this case, and since she is unaware of the symmetric key k, she cannot succeed under the challenge of forging Alice's signatures because of the reasons mentioned above that make such an attack easily detected.

2. *Replaying attacks*: In this attack, Eve captures the message created by Alice at time t and retransmits it after a period of time. However, each received message is checked for freshness using the attached timestamp T_a by verifying if $T_r - T_a \leq T_\Delta$ holds. Hence, providing immunity from replay attacks.

3. *Modification attacks*: In this attack, Eve attempts to alter the data packets. However, the integrity of the received messages is verified using the proposed feature tracking algorithm. In case of Eve is trying to alter only the subcarriers that hold the safety-related message m without any modification in the received probe symbols and the PHY-layer signature and retransmits the altered message at time $T_a + \Delta T_a$. In this scenario and if and only if $T_r - T_a \leq T_\Delta$, Eve can deceive the feature tracking mechanism at the side of Bob as the channel estimation vectors $\bar{H}_j$, for $j = 1, \ldots, M$, will be highly correlated as all the probe symbols have the same channel response from Alice to Bob passing through Eve. However, the accumulated noises significantly increase the value of the estimated variance v in (4.14), thereby failing to pass the hypothesis test in (4.16), accordingly, the received message will be discarded. Thus, providing immunity against modification attacks.

4.4 Simulation and hardware implementation

This section presents the simulation of the key extraction and then describes the Doppler shift emulation employed for the hardware implementation of the re-authentication process.

4.4.1 Simulation analysis of the key extraction algorithm

During the tests, Monte-Carlo simulations are conducted of 100,000 runs to evaluate the key extraction performance. In this study, a Rayleigh environment is employed to effectively model urban areas. This is achieved through the utilisation of the generic stochastic vehicular channel model presented in [118] with $L = 16$ multipath components. Since the DSRC protocol operates within the range of 5.85 to 5.925 GHz [137], the carrier frequency f_c is set at 5.85 GHz. The parameters of the simulated channel are listed in Table 4.2. According to [119], the scatterers' speeds follow the Weibull distribution with shape ζ and scale ρ. Tx/scatterer speeds are set to 30 m/s. In this study, the extraction performance is defined as the achievable SBGR for an acceptable BMR $\leq a_1$, for $a_1 = 0.1$. In Fig. 4.8, the extraction performance is plotted at different SNR and r values. It can be noted that the highest SBGR is obtained for an acceptable BMR at

Table 4.2: Channel simulation settings

Description	**Value**
The number of multipath components L	16
Maximum speed of the Tx	30 m/s
Maximum speed of the Rx (for V2I scenario)	0 m/s
Maximum speed of the scatterers	30 m/s
Azimuth angles of departure (arrival) $\alpha_{A(B),l}$	$U[-\pi,\pi)$
Elevation angles of departure (arrival) $\beta_{A(B),l}$	$U[0,\pi/3)$
Scatterers' angles of incident/departure $\alpha_{1(2),l}$	$U[-\pi,\pi)$
The Weibull distribution's scale coefficient ρ	2.985
The Weibull distribution's shape coefficient ζ	0.428
Carrier frequency f_c	5.85 GHz

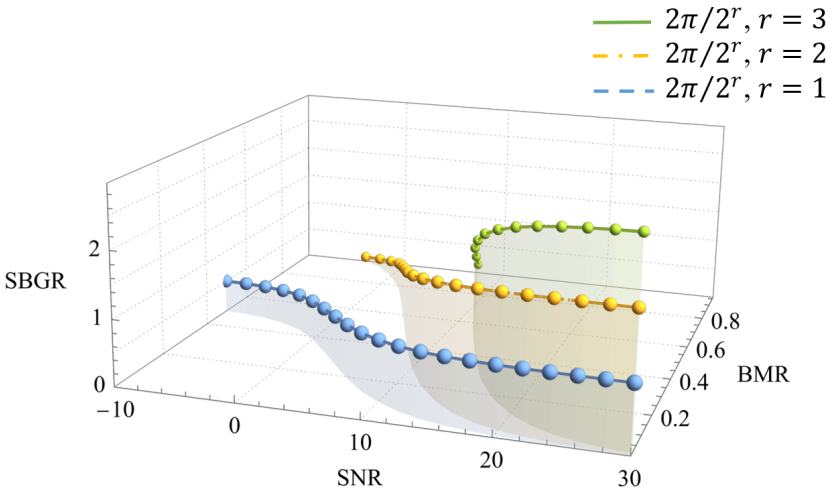


Figure 4.8: Key extraction performance at different r values.

the thresholding region of order 3 and SNR $\geq$ 22 dB. For 16 $\leq$ SNR $\leq$ 22, $r = 2$ is evidently the optimum choice for an acceptable performance within this range. While $r = 1$ is clearly the unique acceptable quantisation order at 7 $\leq$ SNR $\leq$ 16. In Fig. 4.9, the simulation analysis of the CDFs of $\angle e_{a(b)}(t)$ in (4.6), for $r = 1,2,3$, are compared to its theoretical formulation in (4.19) across different SNRs. The results show that there exists an optimal quantisation order r across different ranges of SNRs. By adjusting the r value to different SNR conditions, the extraction performance can be optimised for an acceptable P_e formulated in (4.20).

What's more, the extracted bitstreams are checked for any statistical defects by using the well-known randomness test suite developed by the national institute of standards and technology (NIST) [138]. By doing so, each test returns a P-value, as shown in Table 4.3. This value is compared to the significance level (0.01) to determine whether the extracted bitstreams have successfully passed the test. It can be noted that the extracted keys have sufficient randomness, as their chaotic characteristics are mostly determined by the randomly selected users' secret

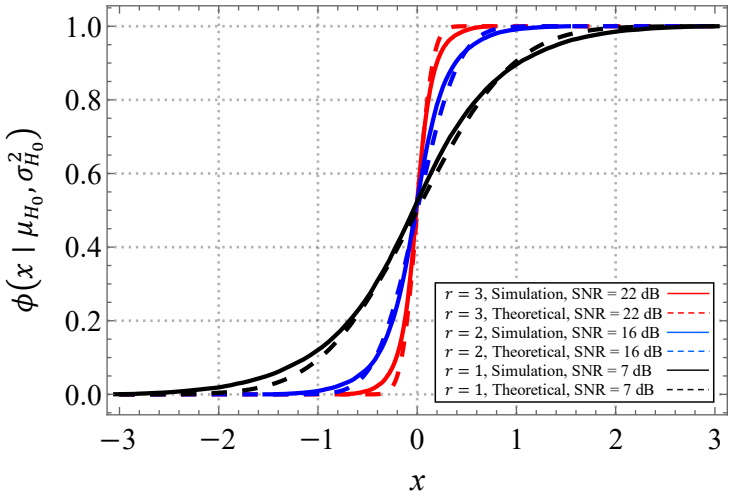


Figure 4.9: $\phi(x)\mid_{H_0}$ at different SNRs and r values.

Table 4.3: Randomness evaluation of the extracted keys

NIST Statistical Test Suite (128 bits)	**P-value**
Block Frequency Test	0.486427
Long Runs Test	0.487804
Monobit Test	0.58592
Key Entropy	0.300445
Maurer Universal Statistical Test	0.163067
Discrete Fourier Transform (Spectral) Test	0.495118
Overlapping Template Matchings Test	0.486427

parameters n_i and m_i of the Chebyshev mapping operation in (4.3).

4.4.2 Hardware implementation and Doppler shift emulation

In a realistic V2I scenario, measuring the ROCs at different Tx speeds of a moving vehicle is a challenging issue for performance evaluation due to the speed instability of the transmitter at different distances from the receiver, resulting in an unstable and inaccurate measurement of the detection probabilities. Therefore, the re-authentication performance is evaluated using the two channels of the ETTUS-USRP X310 device as separate Tx/Rx terminals, with LabView utilised as a software-defined radio. The evaluation is conducted in a real vehicular wireless channel by setting a random fixed distance (e.g., 5 meters) between the Tx and Rx antennas, as shown in Fig. 4.10, and varying the power of the added complex Gaussian noise at the side of the Rx (i.e., different SNRs). This subsection presents a solution for a realistic Doppler emulation. By simulating the Doppler components of a moving vehicle at the Tx side, the ROCs of the re-authentication process are successfully investigated at different speeds for the OFDM communication system. Fig. 4.11(a) and 4.11(b) show the 3D azimuth and elevation angles

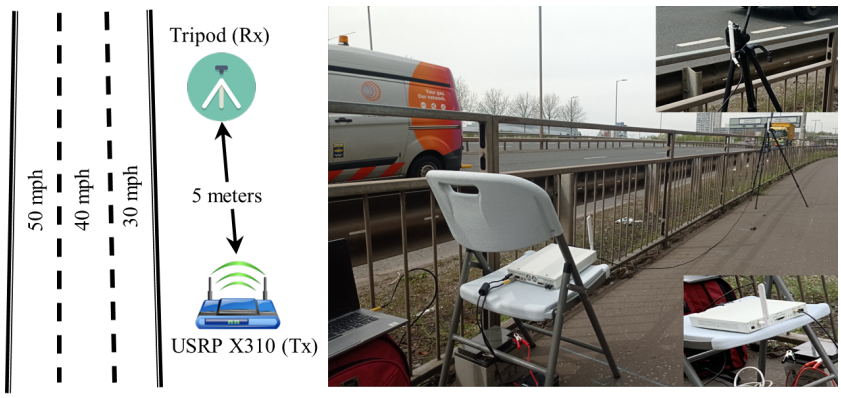


Figure 4.10: Experimental settings for performance evaluation.

of departure $\alpha_{a,l}(t_2) \sim U[-\pi,\pi)$ and $\beta_{a,l}(t_2) \sim U[0,\pi/3)$ for the l^{th} multipath component of the j^{th} OFDM symbol at the Tx side. It can be noted from the same figure that the upcoming $\alpha'_{a,l}(t_2+\Delta t)$ and $\beta'_{a,l}(t_2+\Delta t)$ of the $(j+1)^{th}$ symbol depends on the speed of the transmitter u_a and the transmission time Δt between the j^{th} and $(j+1)^{th}$ symbols. The distance δ (meter) driven by the Tx can be obtained as

$$\delta = u_a \times \Delta t \tag{4.32}$$

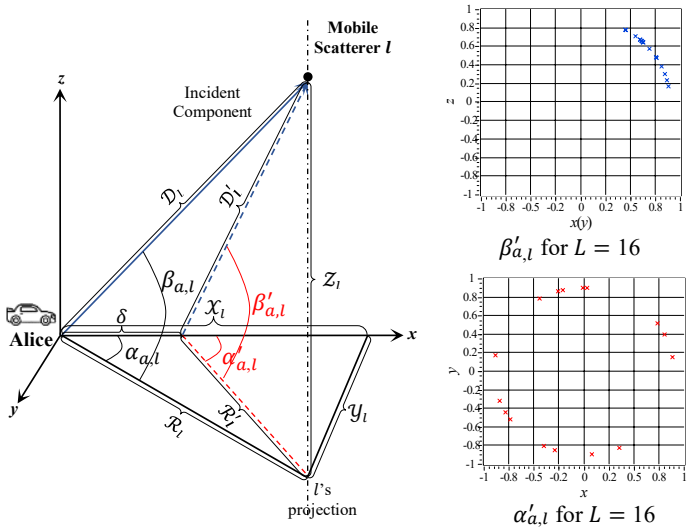
In urban areas, it is assumed that the direct distance D_l between the transmitter and the scatterer l with coordinates $\{X_l, Y_l, Z_l\}$ is a uniformly distributed random variable within few meters from the transmitter $D_l \sim U[1,3]$ since most of the received power at the Rx side is coming from the multipath components with short distances, referred to as specular components [118]. In this scenario, the upcoming azimuth angle $\alpha'_{a,l}$ of the l^{th} multipath component can be formulated using trigonometric as

$$\alpha'_{a,l} = \arctan\left(\frac{D_l \cos(\beta_{a,l}) \sin(\alpha_{a,l})}{D_l \cos(\beta_{a,l}) \cos(\alpha_{a,l}) - \delta}\right) \tag{4.33}$$

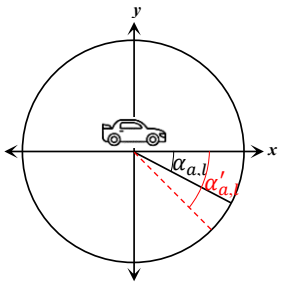
By dividing the range of the azimuth angle (2π) into a number of r_l step angles $\Delta_l(\text{rad}) = \left|\alpha'_{a,l} - \alpha_{a,l}\right|$, as shown in Fig. 4.11(c). In this case, the resolution value r_l of the l^{th} scatterer can be approximated by

$$r_l = \left\lfloor \frac{2\pi}{\left|\alpha'_{a,l} - \alpha_{a,l}\right|} \right\rfloor \tag{4.34}$$

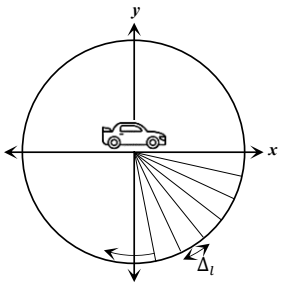
so that the azimuth angle of the j^{th} OFDM symbol equals $\alpha'_{a,l}(j) = \alpha_{a,l} + (j-1)\Delta_l$ for $j =$



(a) 3D angles of departure at the Tx side.



(b) $\alpha_{a,l}$ and $\alpha'_{a,l}$ at the Tx.



(c) Azimuth step angle Δ_l.

Figure 4.11: 3D V2V departure angles of the l^{th} scatterer.

$1,\ldots,M$. While the elevation angle $\beta'_{a,l}(j)$ is approximated using trigonometric by

$$\beta'_{a,l}(j) = \left| \arctan\left(\frac{\sin(\beta_{a,l}) \sin\left(\alpha'_{a,l}(j)\right)}{\cos\left(\beta_{a,l}\right) \sin\left(\alpha_{a,l}\right)} \right) \right| \tag{4.35}$$

Then, the Doppler shift at the Tx side can be expressed as

$$v_{a,l} = u_a \frac{f_c}{c} \cos\left(\alpha'_{a,l}(j)\right) \cos\left(\beta'_{a,l}(j)\right) \tag{4.36}$$

where c is the speed of light. Eventually, the l^{th} Doppler multipath component can be approximated by

$$d_{a,l}(t) = e^{j2\pi v_{a,l} t} \tag{4.37}$$

The Doppler emulation steps at the Tx side can be summarised in the algorithmic form as shown in Algorithm 1. By creating $L = 16$ Doppler components at the Tx side and convoluting them with the generated symbols before transmitting through the USRP, see Fig. 4.12, the

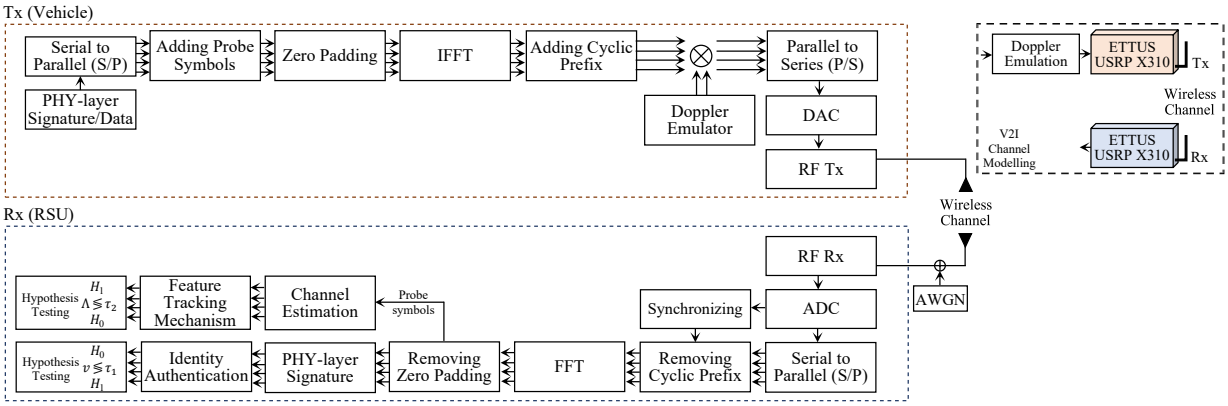
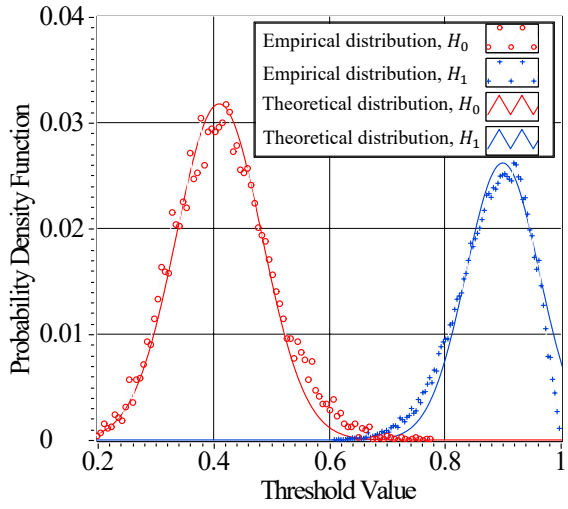
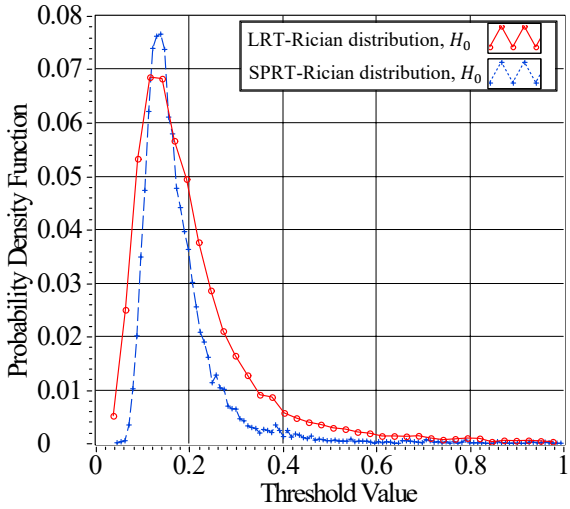


Figure 4.12: OFDM Tx/Rx block diagram.



(a) PDFs for hypothesis testing of the PHY-SIAM.



(b) PDFs for hypothesis testing of the PHY-FTM.

Figure 4.13: PDFs for both hypotheses of the re-authentication process at 64 subcarriers, u_a = 30 m/s, and SNR = 5 dB.

impact of the transmitter's speed on the re-authentication performance can be determined.

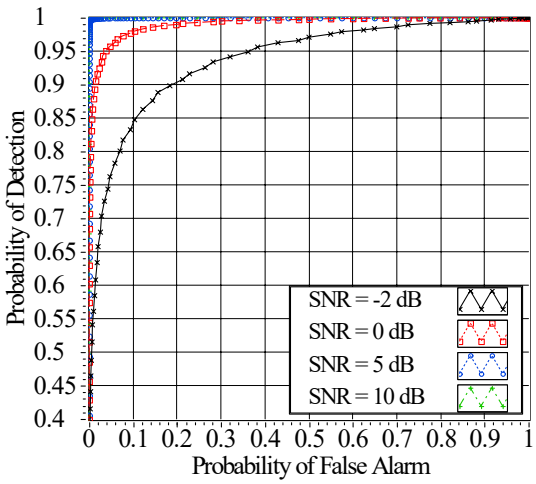
4.4.3 Hardware implementation results

Experimentally, f_c and the sampling rate are set to 5.85 GHz and 1 MHz, respectively. It is vital to examine the effect of the distance between the Tx and Rx antennas by comparing different SNRs independently from the Tx speed. Fig. 4.13(a) shows the empirical PDFs of the identity authentication mechanism for both hypotheses $H_{0,1}$ in comparison to their theoretical Gaussian distribution $\mathscr{F}(x)\,|_{H_{0,1}}$ in (4.29) at SNR = 5 dB, N = 64 subcarriers, and Tx speed u_a = 30 m/s. Based on the results, the theoretical and empirical Gaussian distributions are well-matched, and both hypotheses are well-separated, allowing for an easy determination of the threshold value τ_1. In the same settings, Fig. 4.13(b) compares the empirical LRT with the SPRT-Rician distribution of the feature tracking mechanism for the H_0 hypothesis.

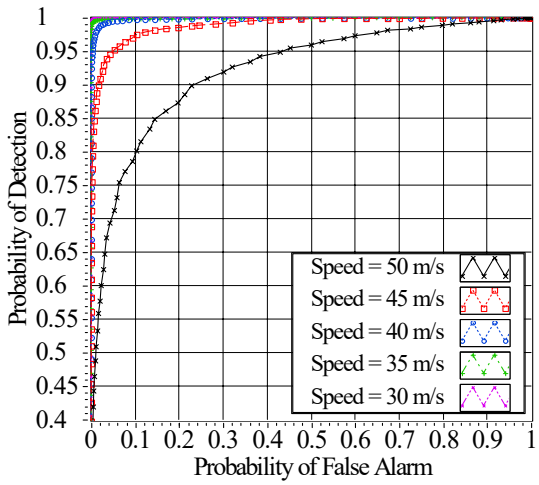
Algorithm 1 Doppler Shift Emulation

Require
Adjust the speed of the Tx $u_a \geq 0\ m/s$
Adjust the transmission time interval $\Delta t = 16\mu$s
Adjust the value of L to 16 multipath components
Calculate the horizontal distance $\delta = u_a \times \Delta t$
for $j = 1 : M$ **do**
  for $l = 1 : L$ **do**
    Select the Tx azimuth angle $\alpha_{a,l} \leftarrow U[-\pi, \pi)$
    Select the Tx elevation angle $\beta_{a,l} \leftarrow U[0, \pi/3)$
    Select the direct distance $D_l \leftarrow U[1,3]$
    Calculate the upcoming azimuth angle $\alpha'_{a,l}$ using (4.33)
    Calculate the step angle $\Delta_l(\text{rad}) = \left|\alpha'_{a,l} - \alpha_{a,l}\right|$
    Get the j^{th} azimuth angle $\alpha'_{a,l}(j) = \alpha_{a,l} + (j-1)\Delta_l$
    Get the j^{th} elevation angle $\beta'_{a,l}(j)$ using (4.35)
    Using $\alpha'_{a,l}(j)$ and $\beta'_{a,l}(j)$ of the j^{th} OFDM symbol
    Calculate $v_{a,l}$ using (4.36)
    Return the Doppler component $d_{a,l}$ using (4.37)
  end for
end for

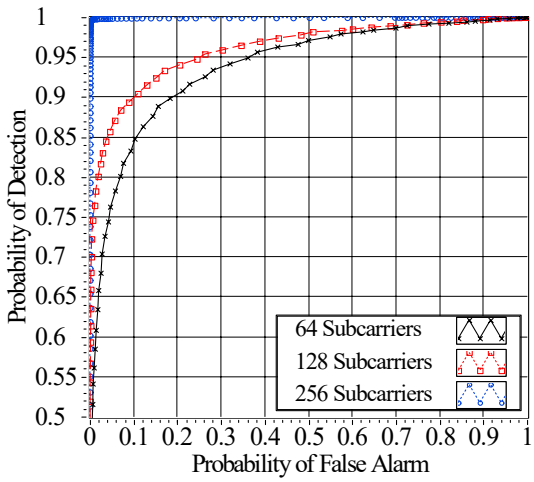
Fig. 4.13(b) illustrates two important observations: 1) The distribution is Rician due to the direct line-of-sight path between the Tx and Rx antennas; 2) The variance of the SPRT distribution is smaller than that of the LRT since the SPRT is considered to be a LRT for a plurality of M OFDM symbols, indicating better performance than the LRT. In Fig. 4.14(a), the ROC curves are plotted at SNR = [10, 5, 0, -2] dB, N = 64 subcarriers, and u_a = 30 m/s. It can be seen that the proposed mechanism for identity authentication offers acceptable performance $(P_{fa} \leq 0.1)$ at SNR $\geq$ 0 dB. In addition, the ROCs are investigated at different Tx speeds u_a = [30, 35, 40, 45, 50] m/s and SNR = 5 dB, as shown in Fig. 4.14(b). In test settings up to 45 m/s, it is proven that PHY-SIAM exhibits high authentication performance. Further, the ROCs are identified at N = [64, 128, 256] subcarriers, u_a = 30 m/s, and SNR = -2 dB, as shown in Fig. 4.14(c). As can be seen in the figure, increasing the number of subcarriers results in enhanced ROC since PDFs follow the central limit theorem. Therefore, the more subcarriers, the smaller the variance $var(v \mid H_{0,1})$ of the Gaussian distribution of v in (4.29), which, in turn, leads to reduced overlapping between the two hypotheses, thereby improving the authentication performance.



(a) ROCs at different SNRs for $u_a = 30$ m/s and $N = 64$.



(b) ROCs at different speeds for SNR = 5 dB and $N = 64$.



(c) ROCs at different N for $u_a = 30$ m/s and SNR = -2 dB.

Figure 4.14: ROCs of the PHY-SIAM at different parameters for a fixed distance (5m) between the Tx and Rx.

4.5 Summary

This study proposes a novel, efficient, and secure cross-layer authentication scheme that supports forward and backward secrecy in VANETs. This chapter demonstrates that by using the cryptographic features of the Chebyshev mapping in combination with the physical layer properties, it is possible to obtain high entropy secret bitstreams, not only applicable for V2V but also for V2I. With the proposed key extraction technique, the tradeoff relation between BMR and BGR is optimised for optimal performance in any wireless propagation conditions, moving beyond the current state-of-the-art in achieving SBGR $\simeq 0.85 \sim 2.76$ bits/packet at SNR of $7 \sim 22$ dB. By leveraging the existing PHY-layer authentication techniques, this chapter introduces PHY-SIAM and PHY-FTM, two PHY-layer re-authentication mechanisms that can be used for identity and

integrity verification, respectively, mitigating the considerable costs of traditional cryptographic techniques. Besides theoretical analysis, an efficient Doppler emulator is developed to experimentally investigate the re-authentication performance of a realistic vehicular wireless channel at different speeds and SNRs of a V2I scenario. Experimental measurements demonstrate the effectiveness of the re-authentication algorithm in providing high detection at low false alarm probabilities ($P_{fa} \leq 0.1$) for SNR ≥ 0 dB and Tx speed ≤ 45 m/s. The next chapter examines the advantages of incorporating RIS technology within the PHY-layer authentication process, particularly in enhancing the detection probability for NLoS communication scenarios.

Chapter 5

RIS-Assisted Cross-Layer Authentication

Recently, the reconfigurable intelligent surface (RIS) has emerged as a technology that can improve communication quality by adjusting reflection coefficients such as phase, amplitude, frequency, or polarization [155, 157]. The RIS has been employed in various applications, such as satellite communications [158], physical layer security [159], and IoT networks [160], demonstrating its versatility and potential for diverse use cases. RISs comprise many passive reflecting units that can be placed at adaptable locations and independently alter the incident signal, thereby improving signal transmission. The performance of PHY-layer authentication in terms of detection and false alarm probabilities depends on the SNR value. The higher the SNR, the higher the detection probability, and vice versa. Considering the significant wireless channel variations and the instability of vehicular communication links caused by unpredictable obstructions, the re-authentication performance can be adversely affected, posing a challenge. The RIS technology can enhance wireless communication systems' SNR values, resulting in improved PHY-layer re-authentication performance.

The following summarises the contributions of this chapter which are published in [22], fulfilling the outlined thesis objectives detailed in Subsection 1.4.3:

1. This chapter extends the work introduced in Chapter 4 by developing a pseudo-identity-based PHY-layer re-authentication method, following the initial legitimacy detection using PKI-based authentication. This significantly reduces the communication and computation costs of transmitting and verifying a cryptographic signature for every message transmission while maintaining the security and privacy requirements of VANETs.

2. For enhanced performance, this chapter demonstrates how the RIS can help improve the PHY-layer authentication's detection probability for non-line-of-sight V2I communication scenarios. Accordingly, the scheme's performance for RIS-assisted vehicular communication is investigated through theoretical analysis and practical experimentation using 1-bit RIS with 64×64 reflective elements.

3. In addition, this chapter demonstrates that the proposed scheme satisfies VANETs' secu-

rity and privacy requirements and resists passive and active attacks. The final analysis compares the scheme's computation and communication costs to traditional crypto-based approaches.

The structure of the remainder of this chapter is as follows. Section 5.1 presents the proposed scheme. Section 5.2 analyses the scheme's security and privacy. Section 5.3 evaluates the scheme's performance. Finally, Section 5.4 provides concluding remarks.

5.1 RIS-assisted authentication: The proposed scheme

This section describes the system model, discusses the proposed scheme in detail, and explains how the RIS enhances the scheme's performance at low SNR values.

5.1.1 System modelling

The system model of the proposed RIS-assisted vehicular communication scheme is depicted in Fig. 5.1. The considered system model consists of the following entities.

- *TA*: The TA is a trusted entity for all network terminals, possessing sufficient computational resources to register and revoke any network terminal. It is also responsible for generating and distributing the system's public parameters. In addition, it is the only terminal capable of revealing vehicles' real identities in case of misbehaving (such as constructing an attack or violating traffic laws).
- *RSU*: The RSU authenticates vehicles within range by verifying their broadcasted messages. It is also assumed to have a reliable communication link with the RIS's smart

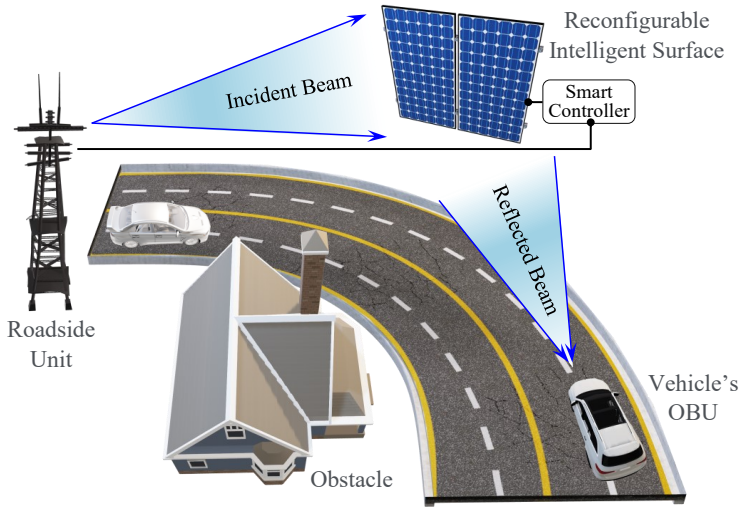


Figure 5.1: System modelling for the proposed RIS-assisted cross-layer authentication scheme.

controller, where it can control the phase shift of the RIS elements. The RSU aims to optimise the RIS's configuration to form a directed beam toward the communicating vehicle in the shadowed areas.

- *Vehicle's OBU*: The OBU is a vehicle-mounted wireless communication device with limited computing capabilities. It can authenticate with nearby RSUs to send and receive real-time traffic conditions. It is assumed that both RSU and OBU are equipped with a single antenna.

- *RIS*: The RIS comprises L reconfigurable passive reflectors and is deployed to provide reliable communication links between the RSU and vehicles' OBUs (see Fig. 5.1). By doing so, the reflected signal towards the designated vehicle/RSU can be deliberately strengthened or impaired. Each RIS has a smart controller that allows the RSU to adjust the phase shift of the RIS reflecting units by choosing between different configuration patterns.

The notations used in this chapter are summarised in Table 5.1 for ease of understanding.

Table 5.1: List of notations for the proposed RIS-assisted cross-layer authentication scheme

Symbol	**Definition**
PPs	The system's public parameters
β, PK_{TA}	The system's master key and TA's public key
PK_{V_i}, SK_{V_i}	V_i's public and private keys, respectively
$Cert_{V_i}$	V_i's long-term digital certificate
T_R	The certificate validation time
TID_{V_i}, PID_{V_i}	V_i's temporary and pseudo identities
PK_{R_k}, SK_{R_k}	R_k's public and private keys, respectively
$Cert_{R_k}$	R_k's long-term digital certificate
TID_{R_k}	R_k's temporary identities
Sk_{i-k}	The shared key between V_i and R_k
σ_{V_i}, σ_{R_k}	V_i's and R_k's signatures
$\sigma_{V_i}^{PHY}$	V_i's PHY-layer signature
ϕ_a, ϕ_b	The PHY-layer signature's phase shifts
T_i, T_r	signatures' creating and receiving timestamps
T_Δ	Timestamps' expiration period, e.g., [00:00:59]
P_d, P_{fa}	The detection and false alarm probabilities

5.1.2 The proposed authentication scheme

This section provides a detailed discussion of the proposed scheme. In this work, each terminal has a long-term digital certificate for initial verification and handshaking between two legitimate parties. For re-authentication and secure message verification between vehicles and RSUs, PHY-SIAM and PHY-FTM [20] are used as a two-factor re-authentication method for the OFDM system of N subcarriers. The proposed scheme comprises four phases, i.e., initialisation, registration, initial authentication, and message signing and verification.

System initialisation phase

The TA follows the following steps to initialise the system's public parameters.

- The scheme is designed based on the 80-bit security level of the elliptic curve $E : y^2 = x^3 + ax + b \bmod p$. In this context, the 160-bit elliptic curve is adopted, which is parameterised using the recommended domain settings of "$secp160k1$" [141], see Table 5.2.
- Based on the generator P, the TA generates a cyclic group $\mathbb{G}$ of order q, which consists of all E's points as well as the infinity point $\mathscr{O}$.
- The TA chooses the system master key $\beta \in Z_q^*$, then computes its related public parameter $PK_{TA} = \beta.P$.
- The TA selects two hash functions $H_1 : \{0,1\}^* \rightarrow \{0,1\}^{N_1}$ and $H_2 : \{0,1\}^* \rightarrow \{0,1\}^{2N_2}$ for $N_2 = 3N/4$ (i.e., $3/4$ of the total number of subcarriers). It also selects the 2-bit Gray code mapping function $\mathscr{M}(x_i) \rightarrow \phi_i$ that maps x_i to ϕ_i as follows.

$$\phi_i = \mathscr{M}(x_i) = \begin{cases} 0 & x_i = [00] \\ \frac{\pi}{2} & x_i = [01] \\ \pi & x_i = [11] \\ \frac{3\pi}{2} & x_i = [10] \end{cases}, \forall i \in [1, N_2] \tag{5.1}$$

Table 5.2: The 160-bit EC's recommended parameters of "$secp160k1$" in the Hexadecimal form [141]

Par.	Value
a	00000000 00000000 00000000 00000000 00000000
b	00000000 00000000 00000000 00000000 00000007
p	$FFFFFFFF$ $FFFFFFFF$ $FFFFFFFF$ $FFFFFFFE$ $FFFFAC73$
q	01 00000000 00000000 0001$B8FA$ 16$DFAB9A$ $CA16B6B3$
P	04 $3B4C382C$ $E37AA192$ $A4019E76$ $3036F4F5$ $DD4D7EBB$ $938CF935$ $318FDCED$ $6BC28286$ $531733C3$ $F03C4FEE$

- Finally, the system's public parameters PPs can be represented by the tuple $\langle a, b, p, q, P, Pk_{TA}, H_1, H_2, \mathscr{M} \rangle$.

Registration phase

The TA registers all terminals before being part of the network by performing the following steps.

- For vehicle registration, the TA checks the vehicle V_i's real identity RID_{V_i}, selects at random V_i's secret key $SK_{V_i} \in Z_q^*$, and calculates its related public parameter $PK_{V_i} = SK_{V_i}.P$. Finally, the TA preloads the tuple $\langle PPs, SK_{V_i}, Cert_{V_i} \rangle$ onto V_i's OBU, where V_i's long-term digital certificate $Cert_{V_i} = \langle PK_{V_i}, T_R, \sigma_{TA} \rangle$, $\sigma_{TA} = Sign_\beta(PK_{V_i} \| T_R)$ and T_R is the certificate validation time.

- Each RSU R_k undergoes the same registration process.

- The TA creates a list of revoked vehicles' and RSUs' digital certificates known as the certificate revocation list $CRL = \{Cert_1, ..., Cert_z\}$, where z is the total number of revoked vehicles and RSUs. At last, the TA distributes the CRL among vehicles via RSUs in different regions.

Initial authentication phase

Consider a scenario where V_i is within the communication range of R_k and wants to initiate a secure connection. In this case, both terminals, V_i and R_k, exchange certificate-based initial authentication packets for mutual legitimacy detection and extracting a symmetric shared key SK_{i-k}. The following steps constitute this phase.

- V_i selects at random a temporary identity $TID_{V_i} \in \{0,1\}^{N_1}$ and sends R_k a request to communicate in the form of $\langle TID_{V_i}, T_1, Cert_{V_i}, \sigma_{V_i} \rangle$, where the signature $\sigma_{V_i} = \text{Sign}_{SK_{V_i}}(TID_{V_i} \| T_1 \| Cert_{V_i})$ and T_1 is the attached timestamp.

- Avoiding replay attacks, R_k checks T_1's freshness by testing whether if $T_r - T_1 \leq T_\Delta$ holds or not. Then, R_k checks V_i's legitimacy by determining if $Cert_{V_i} \in CRL$ holds or not. After that, R_k authenticates the received tuple by verifying σ_{V_i} as Verify $(\sigma_{V_i})_{PK_{V_i}}$.

- In response to V_i's request, R_k computes $SK_{i-k} = PK_{V_i}.SK_{R_k}$ using the Diffie-Hellman key exchanging protocol and sends the tuple $\langle TID_{R_k}, T_2, Cert_{R_k}, \sigma_{R_k} \rangle$ to V_i, where TID_{R_k} is the R_k's temporary identity and $\sigma_{R_k} = \text{Sign}_{SK_{R_k}}(TID_{R_k} \| T_2 \| Cert_{R_k})$.

- At last, V_i checks if $T_r - T_2 \leq T_\Delta$ and $Cert_{R_k} \in CRL$ hold or not, verifies σ_{R_k} as Verify $(\sigma_{R_k})_{PK_{R_k}}$, and computes its own symmetric key $SK_{i-k} = SK_{V_i}.PK_{R_k}$.

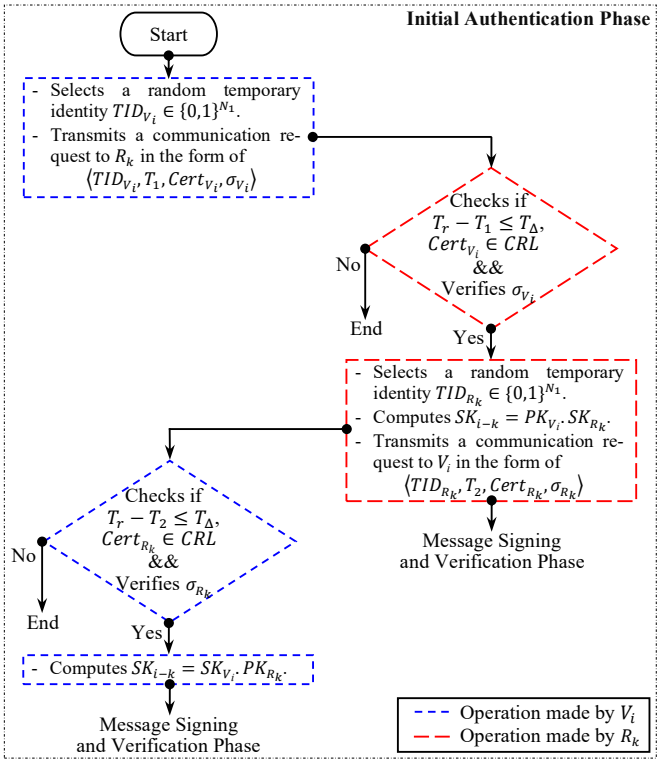


Figure 5.2: The top-level description of the initial authentication.

- Each R_k in a coverage area stores a list of communicating vehicles' temporary identities and their associated shared key so that $list_{R_k} = \{Tuple_1, ..., Tuple_n\}$, where $Tuple_i = \langle Cert_{V_i}, TID_{V_i}, SK_{i-k} \rangle \; \forall i \in [1, n]$.

Fig. 5.2 shows the top-level description flowchart of the initial authentication phase.

Message signing and verification phase

This phase adopts PHY-SIAM and PHY-FTM proposed in [20] (i.e., Chapter 4) as a two-factor re-authentication process performed at the physical layer. A PHY-layer signature is used as a lightweight re-authentication technique based on the symmetric shared key SK_{i-k} and the message payload. Throughout this part, $\mathbb{C}^{N_x \times N_y}$, $\odot$, $()^*$, and $[]^T$ refer to a $N_x \times N_y$ matrix of complex elements, element-wise multiplication, conjugate, and transpose, respectively. While variables in uppercase and Bold represent matrices. The following steps constitute this phase.

- For each specific number Q of message (m) transmissions, V_i selects a random $a_1 \in Z_q^*$ and calculates its related public parameter $A_1 = a_1.P$. Next, V_i computes its pseudo-identity $PID_{V_i} = TID_{V_i} \oplus H_1(a_1.PK_{R_k})$.

- Then, V_i sends R_k the message in the form of $\langle m, PID_{V_i}, A_1, T_3, \sigma_{V_i}^{PHY} \rangle$, where $\sigma_{V_i}^{PHY}$ is the PHY-layer signature computed in a 2-step process as follows.
  - *Signature preparation step*: V_i computes two OFDM symbols' phase shifts, $\Phi_a = [e^{j\phi_{a,1}},$

$..., e^{j\phi_{a,N_2}}]^T \in \mathbb{C}^{N_2\times 1}$ and $\Phi_b = [e^{j\phi_{b,1}}, ..., e^{j\phi_{b,N_2}}]^T \in \mathbb{C}^{N_2\times 1}$, where $\phi_a = \mathscr{M}(H_2(\{SK_{i-k}\}_x \| T_3 \| A_1 \| PID_{V_i} \| m))$, $\phi_b = \mathscr{M}(H_2(\{SK_{i-k}\}_y \| T_3 \| A_1 \| PID_{V_i} \| m))$, and $\{.\}_x$ and $\{.\}_y$ represent the x and y coordinates of the elliptic curve point $SK_{i-k} \in \mathbb{G}$, respectively.

- *Signature generation step*: In this step, V_i encapsulates ϕ_a and ϕ_b onto two subsequent OFDM symbols of N subcarriers and sends it to R_k at times t and $t+\Delta t$ so that the transmitted symbols can be represented as

$$\begin{aligned} \mathbf{S}_1 &= [s_{1,1}, ..., s_{1,N_2}]^T = \Phi_a \odot \mathbf{X}, \\ \mathbf{S}_2 &= [s_{2,1}, ..., s_{2,N_2}]^T = \Phi_b \odot \mathbf{X} \end{aligned} \tag{5.2}$$

where $\mathbf{X} = [e^{j\psi_1}, ..., e^{j\psi_{N_2}}]^T \in \mathbb{C}^{N_2\times 1}$, ψ_i is a uniformly distributed random variable $\psi_i \sim U[0, 2\pi)$, and Δt is the transmission time interval. Note that the OFDM symbols in (5.2) are collectively referred to as $\sigma_{V_i}^{PHY}$. Also, the OFDM system is considered as a superposition of N independently operating narrow-band subsystems.

- R_k receives $\sigma_{V_i}^{PHY}$ in (5.2) at times t' and $t'+\Delta t$, which can be represented in the frequency-domain, following the removal of the cyclic-prefix and calculating the fast Fourier transform (FFT), as

$$\begin{aligned} \mathbf{R}_1 &= [r_{1,1}, ..., r_{1,N_2}]^T = (\mathbf{H}_{VR} \odot \mathbf{S}_1) + \mathbf{N}, \\ \mathbf{R}_2 &= [r_{2,1}, ..., r_{2,N_2}]^T = \left(\mathbf{H}'_{VR} \odot \mathbf{S}_2\right) + \mathbf{N}' \end{aligned} \tag{5.3}$$

where $\mathbf{H}_{VR} = [|h_{1,1}|e^{j\xi_{1,1}}, ..., |h_{1,N_2}|e^{j\xi_{1,N_2}}]^T \in \mathbb{C}^{N_2\times 1}$, $\mathbf{H}'_{VR} = [|h'_{1,1}|e^{j\xi'_{1,1}}, ..., |h'_{1,N_2}|e^{j\xi'_{1,N_2}}]^T \in \mathbb{C}^{N_2\times 1}$, $\{|h_{1,i}|, \xi_{1,i}\}$ and $\{|h'_{1,i}|, \xi'_{1,i}\}$ are the channel amplitude and phase responses of the i^{th} subcarrier at times t' and $t'+\Delta t$, respectively, and $\{\mathbf{N}, \mathbf{N}'\}$ are complex additive Gaussian noises $\mathbb{CN}(0, \sigma_n^2)^{N_2\times 1}$ with means and variances equal zero and σ_n^2, respectively. Note that $\mathbf{H}_{VR}$ is highly correlated with $\mathbf{H}'_{VR}$ for $\Delta t \leq T_c$.

- R_k checks T_3's freshness, computes $TID_{V_i} = PID_{V_i} \oplus H_1(A_1.SK_{R_k})$ and finds out if $TID_{V_i} \in list_{R_k}$ holds or no. If yes, R_k uses SK_{i-k} associated with TID_{V_i} and the message payload $\langle m, PID_{V_i}, A_1, T_3\rangle$ to compute $\phi'_a = \mathscr{M}(H_2(\{SK_{i-k}\}_x \| T_3 \| A_1 \| PID_{V_i} \| m))$ and $\phi'_b = \mathscr{M}(H_2(\{SK_{i-k}\}_y \| T_3 \| A_1 \| PID_{V_i} \| m))$.

- Then, R_k uses a two-factor authentication process, PHY-SIAM and PHY-FTM, in two binary hypothesis testing problems for identity and message verification. This process comprises the following steps.

  - *Message verification step using PHY-SIAM*: In this step, R_k uses the computed ϕ'_a and ϕ'_b to equalise the received PHY-layer signature in (5.3) by computing $\mathbf{R}'_1 = \mathbf{R}_1 \odot \Phi'^*_a$ and $\mathbf{R}'_2 = \mathbf{R}_2 \odot \Phi'^*_b$, where $\Phi'_a = [e^{j\phi'_{a,1}}, ..., e^{j\phi'_{a,N_2}}]^T \in \mathbb{C}^{N_2\times 1}$ and $\Phi'_b = [e^{j\phi'_{b,1}}, ..., e^{j\phi'_{b,N_2}}]^T \in \mathbb{C}^{N_2\times 1}$. Since $\xi_{1,i}$ and $\xi'_{1,i}$ are highly correlated within T_c, R_k verifies the received message by computing $\mathbf{C} = [c_1, ..., c_{N_2}]^T = \mathbf{R}'_1 \odot \mathbf{R}'^*_2$. Then, R_k calculates the circular variance

$c.var(.)$ of $\angle(\mathbf{C}) = [\angle(c_1), ..., \angle(c_{N_2})]^T$ as

$$v = c.var\left(\sum_{i=1}^{N_2} \arctan\left(\frac{\mathrm{Im}\,(c_i(t))}{\mathrm{Re}\,(c_i(t))}\right)\right) \tag{5.4}$$

where $c.var$ is defined as

$$\alpha_i = \begin{pmatrix} \cos\left(\angle\left(c_i\right)\right) \\ \sin\left(\angle\left(c_i\right)\right) \end{pmatrix}, \bar{\alpha} = \frac{1}{N_2}\sum_{i=1}^{N_2} \alpha_i, \qquad v = 1 - \|\bar{\alpha}\| \tag{5.5}$$

where $\|.\|$ represents the norm function. Avoiding impersonation and modification attacks, R_k verifies $\sigma_{V_i}^{PHY}$ in a hypothesis-testing problem given by

$$\begin{matrix} H_0 \\ v \lessgtr \tau_1, \\ H_1 \end{matrix} \text{ for } \begin{cases} H_0: & \Phi'_a = \Phi_a \;\&\; \Phi'_b = \Phi_b \\ H_1: & \Phi'_a \neq \Phi_a \;\&\; \Phi'_b \neq \Phi_b \end{cases} \tag{5.6}$$

where τ_1 is the threshold value and H_0 and H_1 are the hypotheses that state whether the received message has been successfully authenticated or unauthenticated, respectively.

- *Message verification step using PHY-FTM*: Based on the OFDM symbols structure of order M symbols in Fig. 5.3, R_k measures the correlation coefficient between the channel observation vector $\bar{H}_j$ estimated from the reference symbols of the j^{th} OFDM symbol and that $\bar{H}_{j+1}$ of the $(j+1)^{th}$ OFDM symbol, starting from $\sigma_{V_i}^{PHY}$ at $j = \{1, 2\}$ to the M^{th} symbol. Hence, if $\bar{H}_j$ is highly correlated with $\bar{H}_{j+1}$, this means that these symbols are sent from the same transmitter. Otherwise, the received message is discarded. Hence, message verification can be described as a hypothesis-testing process based on the normalised LRT, which is given by

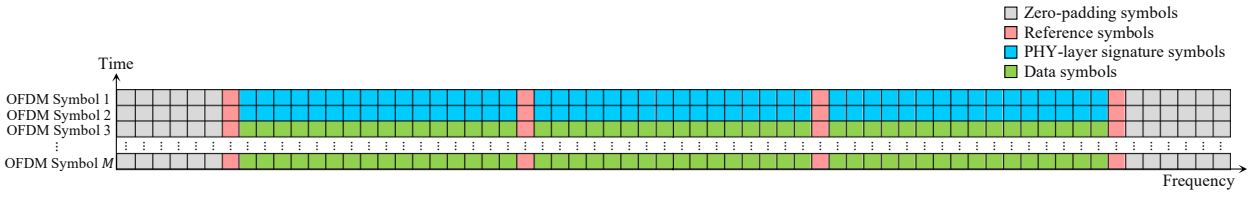


Figure 5.3: OFDM symbols' structure for 64 subcarriers.

$$\Lambda_{LRT} = \frac{n_{\tau_2} \left\| \bar{H}_j - \bar{H}_{j-1} \right\|^2}{\left\| \bar{H}_{j-1} \right\|^2} \quad \forall j \in [2, M],$$
$$H_1 \tag{5.7}$$
$$\Lambda_{LRT} \lessgtr \tau_2$$
$$H_0$$

where $\tau_2 \in [0,1]$ is the threshold value and n_{τ_2} is the normalisation coefficient. The decision rule can be made based on the SPRT that sums the LRTs between the j^{th} and the $(j-1)^{th}$ OFDM symbols $\forall j \in [2, M]$. The SPRT-based hypothesis-testing problem can be expressed as

$$\Lambda_j = \frac{n_{\tau_2} \left\| \bar{H}_{M-j+1} - \bar{H}_{M-j} \right\|^2}{\left\| \bar{H}_{M-j} \right\|^2} \forall j \in [1, M-1],$$
$$H_1 \tag{5.8}$$
$$\Lambda_{SPRT} = n_{\tau_3} \sum_{j=2}^{M} \Lambda_j, \Lambda_{SPRT} \lessgtr \tau_3$$
$$H_0$$

where $\tau_3 \in [0,1]$ is the threshold value and n_{τ_3} is the normalisation coefficient.

- Finally, R_k accepts or discards the received message from V_i based on the decision rule of both PHY-SIAM and PHY-FTM hypothesis problems. Accepted messages are those that are identified by both problems as being H_0. Otherwise, the message will be discarded.

Fig. 5.4 shows the top-level description flowchart of the message authentication and integrity verification phase.

5.1.3 RIS-assisted PHY-layer authentication

One of the challenging issues of PHY-layer authentication is that the detection probability P_d primarily depends on the received signal's SNR value, whereas P_d defines the probability of authenticating legitimate users as authorised terminals. A higher SNR value indicates a higher P_d for an acceptable false alarm probability P_{fa}, and vice versa, where P_{fa} defines the probability of authenticating legitimate users as unauthorised terminals. This makes the PHY-layer authentication impractical in long-range and NLoS vehicular communications. In this challenging scenario, RIS can enhance the power of the received signal at the receiver for the NLoS communication, see Fig. 5.1. As a result, the proposed scheme can effectively authenticate the received messages from the vehicles in the shadowing areas. Thus, the received signals in (5.3) for the i^{th} subcarrier is the superposition of L multipath components coming from L RIS's

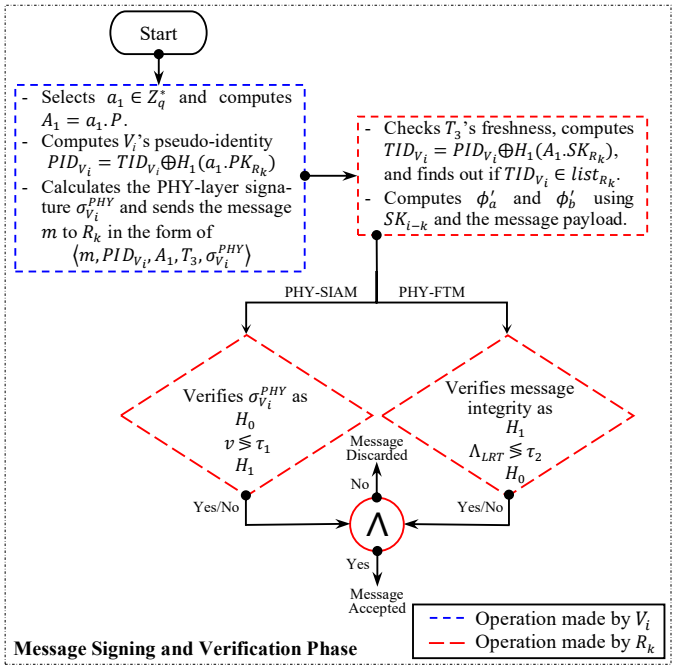


Figure 5.4: The top-level description of the message authentication and integrity verification phase.

reflective elements and can be reformulated as

$$\begin{aligned} r_{1,i} &= (\mathbf{H}_{VI} \odot \mathbf{H}_{IR})\ \omega_\theta\ s_{1,i} + n_i, \\ r_{2,i} &= (\mathbf{H}'_{VI} \odot \mathbf{H}'_{IR})\ \omega_\theta\ s_{2,i} + n'_i \end{aligned} \quad \forall i \in [1, N_2] \tag{5.9}$$

where $\mathbf{H}_{VI} = [|h_{2,1}|e^{j\xi_{2,1}}, ..., |h_{2,L}|e^{j\xi_{2,L}}] \in \mathbb{C}^{1\times L}$, $\mathbf{H}_{IR} = [|h_{3,1}|e^{j\xi_{3,1}}, ..., |h_{3,L}|e^{j\xi_{3,L}}] \in \mathbb{C}^{1\times L}$, and $\omega_\theta = [e^{j\omega_1\theta_1}, ..., e^{j\omega_L\theta_L}]^T \in \mathbb{C}^{L\times 1}$. $\mathbf{H}_{VI}$ and $\mathbf{H}_{IR}$ represents the channel responses from V_i to RIS and from RIS to R_k, respectively. While ω_θ defines the phase shift matrix related to the L reflective elements of the RIS, where θ_l and ω_l defines the l^{th} reflective element phase shift value and state, respectively $\forall l \in [1, L]$, for example, $\theta_l = \pi$ and $\omega_l \in \{0, 1\}$ for a 1-bit RIS. Note that $\{\mathbf{H}_{VI}, \mathbf{H}_{IR}\}$ is highly correlated with $\{\mathbf{H}'_{VI}, \mathbf{H}'_{IR}\}$ within T_c. The RSU in each region optimises the RIS configuration ω_θ to maximise the power of the received signals at the side of the intended user. Hence, improving the ROC of the two-factor re-authentication process at poor SNRs.

5.2 Security and privacy analyses

This section presents the proposed scheme security and privacy analyses.

5.2.1 Security and privacy informal analysis

1. *Message authentication*: The proposed scheme offers legitimacy detection and ensures message integrity for the following reasons:

   - For legitimacy detection, the recipient V_i/R_k verifies the sender's legitimacy R_k/V_i by checking if $Cert_{V_i/R_k} \in CRL$, where $\sigma_{TA} \in Cert_{V_i/R_k}$ is signed using $\beta \in Z_q^*$ and verified by the recipient using $PK_{TA} \in PPs$, which is infeasible to be forged under the difficulty of solving the ECDLP. In addition, the transmitted tuple $\langle TID_{V_i/R_k}, T_1, Cert_{V_i/R_k}, \sigma_{V_i/R_k} \rangle$ is verified for its integrity using the signature σ_{V_i/R_k} that is signed using V_i/R_k's secret key Sk_{V_i/R_k} and verified by the recipient using $Pk_{V_i/R_k} \in Cert_{V_i/R_k}$.
   - For message authentication at subsequent transmission slots, the tuple $\langle m, PID_{V_i}, A_1, T_3, \sigma_{V_i}^{PHY} \rangle$ is verified by R_k for its integrity in a two-factor authentication process, PHY-SIAM and PHY-FTM, that's infeasible to be forged for the following reasons: A) The phase shifts, Φ_a and Φ_b, in (5.2) are computed based on the shared key $SK_{i-k} \in \mathbb{G}$ and masked by $\mathbf{X} = \{e^{j\psi_1}, ..., e^{j\psi_{N_2}}\}$, where ψ_i is a uniformly distributed random variable $\sim U[0, 2\pi)$, which makes it infeasible for an adversary to differentiate between Φ_a and Φ_b and $\mathbf{X}$. B) The high correlation coefficient between subsequent channel observation vectors $\{\bar{H}_{j-1}, \bar{H}_j\}$ in (5.7) $\forall j \in [2, M]$ or $\{\bar{H}_j, \bar{H}_{j+1}\}$ in (5.8) $\forall j \in [1, M-1]$ helps in detecting modification attempts in the message payload.

2. *Privacy preservation*: In the proposed scheme, vehicles communicate using their temporary identities TID_{V_i} at the first transmission slot, while pseudo identities PID_{V_i} are used at subsequent transmissions. This preserves users' real identities RID_{V_i} from exposure as no network terminals possess RID_{V_i} or even the link between RID_{V_i} and its associated long-term digital certificates $Cert_{V_i}$ except for the TA. Only the TA is authorised to expose RID_{V_i} in cases of misbehaviour (for example, when the vehicle constructs an attack or when a driver drives an unregistered vehicle).

3. *Unlinkability*: For each Q number of message transmissions per session, V_i uses a different pseudo-identity $PID_{V_i} = TID_{V_i} \oplus H_1(a_1.PK_{R_k})$, where $a_1 \in Z_q^*$ is dynamically updated for each session. Hence, no parameter is used twice per session, thereby avoiding location-tracking attacks.

4. *Traceability and revocation*: Each RSU in a specific area can report misbehaving vehicles to the TA by sending its associated digital certificate $Cert_{V_i}$. The TA, in turn, reveals its

associated real identity, appends $Cert_{V_i}$ to the *CRL*, and distributes the updated *CRL* among vehicles via RSUs.

5. *Resistance to passive and active attacks*: This part discusses the scheme's resistance against typical adversarial attacks. By considering an adversary, Eve acts as a passive attacker and listens to the communicating terminals' broadcasted messages to deduce any useful information about the symmetric key Sk_{i-k}. In this scenario, Eve attempts to deduce the shared key either during the initial authentication phase (case 1) or during the message signing and verification phase (case 2). In case 1, Sk_{i-k} is calculated using the Diffie-Hellman key exchanging protocol. This makes it difficult for Eve to compute Sk_{i-k} due to the difficulty of solving the ECDLP. In case 2, Eve has difficulty deducing the value of Sk_{i-k} from the PHY-layer signature $\sigma_{V_i}^{PHY}$ due to the following: 1) The signature generation step is dependent on the dynamically updated parameters $\langle T_i, A_i, PID_{V_i}, m\rangle$, which results in different outputs, Φ_a and Φ_b, under the same shared key Sk_{i-k}. In addition, The received $\sigma_{V_i}^{PHY}$ in (5.3) is dependent on the spatially and temporally varying channel phase responses ξ_i and ξ_i' that masks $\phi_{a,i}$ and $\phi_{b,i}$, respectively. 2) For $y = H_2(x)$, it is difficult for Eve to determine the input variable x from the hashed variable $y : \{0,1\}^{N_2}$. In this scenario, Eve is considered an active attacker capable of constructing the following types of attacks:

   - *Modification resistance*: In this attack, Eve tries to modify the message payload either during the initial authentication phase (case 1) or during the message signing and verification phase (case 2). In case 1, the recipient R_k/V_i verifies the received tuple $\langle TID_{V_i/R_k}, T_i, Cert_{V_i/R_k}, \sigma_{V_i/R_k}\rangle$ for its integrity based on the attached signature σ_{V_i/R_k}. For this attack to be successful, Eve must modify the message contents and forge a valid signature, which is computationally intractable due to the difficulty of solving the ECDLP. In case 2, Eve must modify the message contents $\langle m, PID_{V_i}, A_i, T_i\rangle$ and forge a valid signature $\sigma_{V_i}^{PHY}$. Without any knowledge of the shared key Sk_{i-k}, Eve is unable to correctly estimate the values of Φ_a and Φ_b needed to generate a valid signature. Accordingly, this type of attack can be easily detected.
   - *Impersonation resistance*: In this attack, Eve tries to impersonate the communicating vehicle V_i during the initial authentication phase. For this attack to be successful, Eve must generate a valid signature σ_{V_i} using the V_i's secret key Sk_{V_i}, which cannot be forged due to the difficulty of solving the ECDLP. Accordingly, it is hard to compute a valid shared key Sk_{i-k} used for generating $\sigma_{V_i}^{PHY}$ during the message signing and verification phase. Hence, the proposed scheme is resistant to this type of attack.
   - *Replay resistance*: In this attack, Eve repeats the transmission of a previously captured message either during the initial authentication phase (case 1) or during the message signing and verification phase (case 2). In both cases, each transmission

is accompanied by a fresh timestamp T_i that helps the recipient detect this type of attack by testing whether $T_r - T_i \leq T_\Delta$ holds. Hence, the proposed scheme is resistant to replay attacks.

5.2.2 Security proof using BAN-logic formal analysis

The Burrows–Abadi–Needham (BAN) security proof is a formal methodology that offers a rigorous approach to evaluate the security of authentication protocols. The BAN approach is grounded in a formal model of authentication protocols and employs inference rules to analyse the knowledge and beliefs of principals involved in the protocol. Due to its effectiveness, the BAN methodology has been extensively adopted for analysing and verifying the security of authentication protocols in diverse settings such as computer networks, web communications, smart cards, and mobile devices. This study employs the BAN-logic analysis to scrutinise the security of the proposed method against various types of attacks, such as replay, man-in-the-middle, and impersonation attacks.

1. *Notations*: In BAN-logic, security properties are expressed and argued using the following symbols.
   - $A \mid\equiv X$: A believes that the proposition of X is true.
   - $A \triangleleft X$: A sees X denotes that principal A has received a message that includes the value X.
   - $A \mid\sim X$: X has been transmitted to A at some point, and A has subsequently believed the proposition X.
   - $A \mid\Longrightarrow X$: A has control over the value X and has the authority or jurisdiction to manipulate or modify it.
   - $A \overset{k}{\longleftrightarrow} B$: A and B share a secret key k, which they use to securely communicate with each other.
   - $A \overset{k}{\longrightarrow} B$: k denotes the public key attributed to A.
   - $\{X\}_k$: The shared key k is used to encrypt X.
   - $\#(X)$: It represents a fresh message X.
2. *Rules*: A set of deductive rules are used to analyse initial beliefs and protocol messages exchanged between participants and make inferences about the security properties of the protocol. These rules are listed and defined in Table 7.4.
3. *Goals*: The primary objective of BAN-logic is to demonstrate the validity of the proposed scheme by accomplishing the following set of goals.
   - *Goal* 1: $R_k \mid\equiv (R_k \overset{Sk_{i-k}}{\longleftrightarrow} V_i)$.

- *Goal* 2: $R_k \mid\equiv (V_i \mid\equiv M_1)$.
- *Goal* 3: $V_i \mid\equiv (V_i \overset{Sk_{i-k}}{\longleftrightarrow} R_k)$.
- *Goal* 4: $V_i \mid\equiv (R_k \mid\equiv M_2)$.
- *Goal* 5: $R_k \mid\equiv (M_3)$.

4. *Idealised forms*: The following points formulate the idealised messages for the proposed method.
   - M_1: $V_i \rightarrow R_k$: $\{TID_{V_i}, T_1, Cert_{V_i}\}_{SK_{V_i}}$, where $Cert_{V_i} = \{PK_{V_i}, T_R\}_\beta$.
   - M_2: $R_k \rightarrow V_i$: $\{TID_{R_k}, T_2, Cert_{R_k}\}_{SK_{R_k}}$, where $Cert_{R_k} = \{PK_{R_k}, T_R\}_\beta$.
   - M_3: $V_i \rightarrow R_k$: $\{m, PID_{V_i}, A_1, T_3, \sigma_{V_i}^{PHY}\}$, where $\sigma_{V_i}^{PHY} = \{m, PID_{V_i}, A_1, T_3\}_{Sk_{i-k}}$.
5. *Assumptions*: The fundamental assumptions underlying the BAN-logic security proof are as follows.
   - A_1: $R_k \mid\equiv \#(T_1)$.
   - A_2: $V_i \mid\equiv \#(T_2)$.
   - A_3: $R_k \mid\equiv \#(T_3)$.
   - A_4: $R_k \mid\equiv (TA \overset{K_{TA}}{\longrightarrow} R_k)$.
   - A_5: $V_i \mid\equiv (TA \overset{K_{TA}}{\longrightarrow} V_i)$.
   - A_6: $\dfrac{R_k \mid\equiv (TA \overset{PK_{TA}}{\longrightarrow} R_k), R_k \triangleleft \{Pk_{V_i}, T_R\}_\beta}{R_k \mid\equiv (V_i \overset{Pk_{V_i}}{\longrightarrow} R_k)}$.
   - A_7: $\dfrac{V_i \mid\equiv (TA \overset{PK_{TA}}{\longrightarrow} V_i), V_i \triangleleft \{Pk_{R_k}, T_R\}_\beta}{V_i \mid\equiv (R_k \overset{Pk_{R_k}}{\longrightarrow} V_i)}$.
   - A_8: $R_k \mid \equiv (V_i \Longrightarrow M_3)$.
6. *Implementation*: The security proof of the proposed protocol is presented as follows.
   - *Step* 1: Upon receipt of message M_1 from V_i, R_k applies A_4 and $Cert_{V_i} \in M_1$ to A_6, resulting in the following outcome: $O_1 : R_k \mid\equiv (V_i \overset{Pk_{V_i}}{\longrightarrow} R_k)$. Accordingly, R_k computes $SK_{i-k} = PK_{V_i}.SK_{R_k}$ and have $O_2 : R_k \mid\equiv (R_k \overset{SK_{i-k}}{\longleftrightarrow} V_i)$, achieving *Goal* 1.
   - *Step* 2: By applying O_1 and M_1 to R_2 from Table 7.4, the outcome is $O_3 : R_k \mid\equiv (V_i \mid\sim M_1)$. Next, applying A_1 and M_2 to R_5 from Table 7.4 yields $O_4 : R_k \mid\equiv \#(M_1)$. Accordingly, applying O_4 and O_3 to R_3 from Table 7.4 yields $R_k \mid\equiv (V_i \mid\equiv M_1)$, achieving *Goal* 2.
   - *Step* 3: Upon receipt of message M_2 from R_k, V_i applies A_5 and $Cert_{R_k} \in M_2$ to A_7, resulting in the following outcome: $O_5 : V_i \mid\equiv (R_k \overset{Pk_{R_k}}{\longrightarrow} V_i)$. Accordingly, V_i computes $SK_{i-k} = SK_{V_i}.PK_{R_k}$ and have $O_6 : V_i \mid\equiv (V_i \overset{SK_{i-k}}{\longleftrightarrow} R_k)$, achieving *Goal* 3.

- *Step* 4: By applying O_5 and M_2 to R_2 from Table 7.4, the outcome is $O_7 : V_i \mid\equiv (R_k \mid\sim M_2)$. Next, applying A_2 and M_2 to R_5 from Table 7.4 yields $O_8 : V_i \mid\equiv \#(M_2)$. Accordingly, applying O_8 and O_7 to R_3 from Table 7.4 yields $V_i \mid\equiv (R_k \mid\equiv M_2)$, achieving *Goal* 4.
- *Step* 5: Upon receipt of message M_3 from V_i, R_k applies O_2 and $\sigma_{V_i}^{PHY} \in M_3$ to R_1 from Table 7.4 to get $O_9 : R_k \mid\equiv (V_i \mid\sim M_3)$. Next, applying A_3 and M_3 to R_5 from Table 7.4 yields $O_{10} : R_k \mid\equiv \#(M_3)$. Then, applying O_{10} and O_9 to R_3 from Table 7.4 yields $O_{11} : R_k \mid\equiv (V_i \mid\equiv M_3)$. Finally, applying A_8 and O_{11} to R_4 from Table 7.4 yields $O_{12} : R_k \mid\equiv (M_3)$, achieving *Goal* 5.

5.3 Performance evaluation

This section analyses the theoretical and practical aspects of RIS-assisted PHY-layer authentication performance, followed by detailed computation and communication comparisons.

5.3.1 Theoretical analysis of the PHY-layer authentication

In order to evaluate the ROCs of the proposed method, it is crucial to evaluate the PDF for the phase estimate (Θ) of $\mathrm{C} = \mathbf{R}_1' \odot \mathbf{R}_2'^*$, where $\mathbf{R}_1'$ and $\mathbf{R}_2'$ denote the equalised received PHY-layer signature, given by the element-wise multiplication of $\mathbf{R}_1$ in (5.3) and $\Phi_a'^*$, and $\mathbf{R}_2$ in (5.3) and $\Phi_b'^*$, respectively. In the case of $\{\Phi_a, \Phi_b\}$ at the transmitting side V_i are equivalent to $\{\Phi_a', \Phi_b'\}$ at the receiving side R_k, the phase distribution of C for varying SNR values can be formulated according to [20] as follows.

$$P(\Theta \mid \Gamma) = \frac{1}{2\pi} e^{-\Gamma} + \frac{1}{\sqrt{\pi}} (\sqrt{\Gamma}\cos\Theta) \cdot e^{-\Gamma \sin^2\Theta} [1 - \mathbb{Q}(\sqrt{2\Gamma}\cos\Theta)] \tag{5.10}$$

where

$$\begin{aligned} \Gamma &= \frac{|h_i|^2 \cdot {E_S}^2}{\sigma_n^2}, \\ \mathbb{Q}(x) &= \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-t^2/2} dt \end{aligned} \tag{5.11}$$

where E_S is the symbol energy. Fig. 5.5 presents $P(\Theta \mid \Gamma)$ for different SNR values (i.e., $\Gamma \in [0, 25]$ dB). As indicated in (5.4), the circular variance of $\angle(\mathbf{C})$ with a specific order of N_2 is denoted as v, and this quantity satisfies the CLT. Therefore, v's distribution $\mathscr{F}(x)$ follows a normal distribution with a mean (μ_{H_0}) equal to the variance of $P(\Theta)$ for a given Γ value and a variance equal to $\sigma_{H_0}^2$. Thus, the following formulation can express v's CDF for both hypotheses.

$$\phi\left(x \mid \mu_{H_i}, \sigma_{H_i}^2\right) = \frac{1}{2}\left[1 + \mathrm{erf}\left(\frac{x - \mu_{H_i}}{\sqrt{2\sigma_{H_i}^2}}\right)\right], \forall i \in \{0, 1\} \tag{5.12}$$

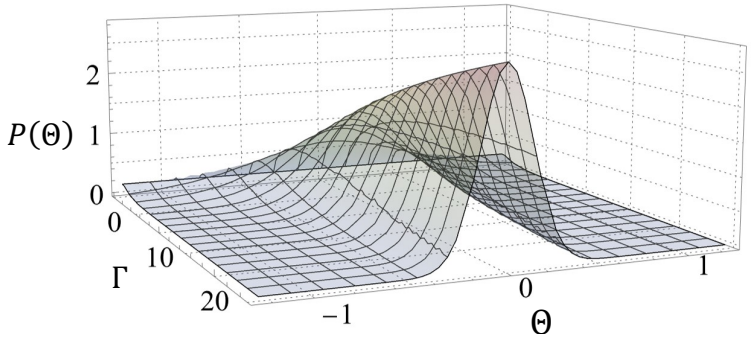


Figure 5.5: $P(\Theta \mid \Gamma)$ in (5.10) at different given $\Gamma \in [0,25]$ dB.

In this context, P_d is defined as $\phi(x \mid \mu_{H_0}, \sigma^2_{H_0})\Big|_{x=\tau_1}$, and P_{fa} is defined as $\phi(x \mid \mu_{H_1}, \sigma^2_{H_1})\big|_{x=\tau_1}$ for a threshold value τ_1 of the hypothesis testing problem in (5.6). As illustrated in (5.11), the channel fading coefficient, represented by $|h_i|$, is a critical factor in determining the value of Γ while maintaining a constant value of E_s and noise variance σ^2_n. Generally, the received signal at the recipient side comprises various multipath components originating from distinct scatterers. Nonetheless, this study considers only the RIS path connecting the communicating terminals, as the impact of the remaining scatterers is consistent regardless of whether the RIS is switched ON or OFF. The channel components of the i^{th} subcarrier in both scenarios, considering the RIS turned ON and OFF, have been expressed in (5.3) and (5.9), respectively. Accordingly, the presence of the RIS can improve the SNR towards the communicating vehicle by configuring the reflective elements in a way that constructively interferes in a specific direction. This can be achieved by controlling the RIS electromagnetic behaviour by optimising ω_θ in (5.9) to maximise the Γ value in (5.11). By doing so, the system's performance at a certain SNR value, denoted as $\Gamma = X$ dB, without the RIS can be equal to its performance at a lower SNR value, $\Gamma = X - \Delta X$ dB, with the RIS. A higher Γ value signifies a decrease in the overlapping between the distributions of both hypotheses, $\mathscr{F}(x)|_{H_0}$ and $\mathscr{F}(x)|_{H_1}$, due to a lower value of μ_{H_0} for $\mathscr{F}(x)|_{H_0}$ relative to μ_{H_1} for $\mathscr{F}(x)|_{H_1}$. This improvement enhances the detection performance while maintaining an acceptable false alarm probability (a_1). Hence, the optimisation of the system's threshold value (τ_1 in (5.6)) can be computed by utilising the following formula [20].

$$\tau_1 = \arg\max_{\tau_1'} \operatorname{erf}\left(\frac{\tau_1' - \mu_{H_1}}{\sqrt{2\sigma^2_{H_1}}}\right) \leq 2a_1 - 1 \tag{5.13}$$

5.3.2 Practical experimentation of the RIS-assisted method

In order to demonstrate the practicality of the proposed RIS-assisted PHY-layer authentication method, the hardware implementation is conducted using a 1-bit RIS consisting of 4096 reflective elements arranged in a two-dimensional 64×64 grid, along with a USRP equipped with two channels (denoted as Ch_0 and Ch_1) that functioned as the transmitter (T_x) and receiver (R_x),

representing R_k and V_i, respectively. The antennas used for T_x and R_x are of the two-horn type, with the T_x antenna beam adjusted perpendicular to the RIS reflecting surface and located 3 meters away from the centre. On the other hand, the R_x antenna was situated 9 meters away from the RIS, with an NLoS path between it and the T_x antenna, and its beam set at a 45-degree angle from the line connecting the T_x antenna to the RIS. Different views of the experimental setup are presented in Fig. 5.6 while Table 5.3 shows the experimental settings.

The carrier frequency is set to 3.75 GHz for 5G-V2I communication. The gains T_x and R_x are set to 20 dB and 5 dB, respectively. The sampling rates for both channels are set to 200 KHz. A range of OFDM systems with different numbers of subcarriers, including 64, 128, and 256, and CP lengths of 16, 32, and 64, are implemented. The optimal configuration associated with the location of the receiving antenna is determined by utilizing the Hadamard codebook. The Hadamard codebook comprises a number of Hadamard matrices that provide a set of binary and orthogonal phase shift states ($\omega_l, \forall l \in [1, L]$) that can be used to modify the reflection of incoming electromagnetic waves in a desired direction or with a preferred phase shift by applying these values to the reflective elements. Accordingly, the proposed re-authentication method is implemented by transmitting two consecutive OFDM symbols with the same structure presented in Fig. 5.3, representing the PHY-layer signature $\sigma_{V_i}^{PHY}$.

Fig. 5.7 shows the received OFDM symbol in the frequency domain following the removal of the CP and applying the FFT. This figure presents the received power in dB for each subcarrier when the RIS is ON and OFF. It can be seen that the power of the subcarriers carrying data has

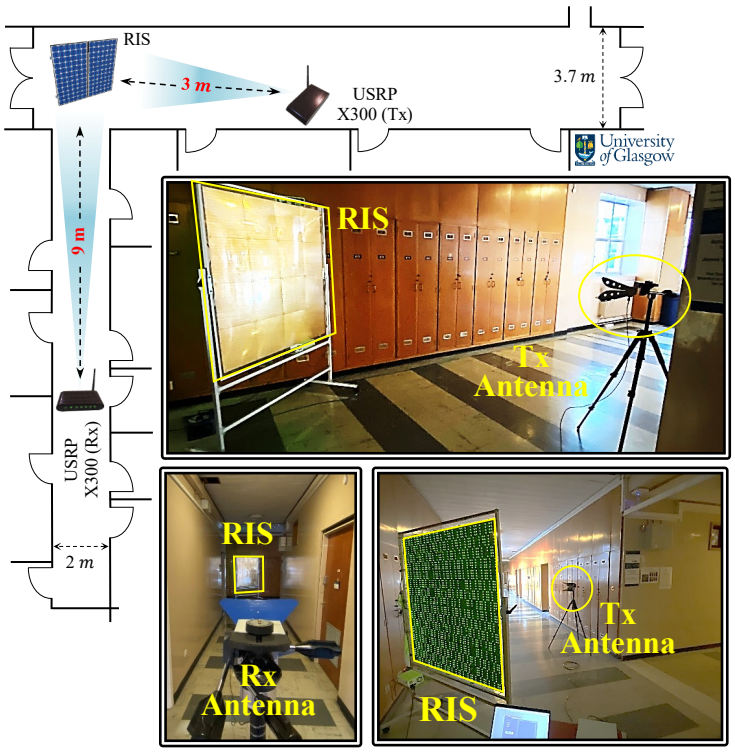


Figure 5.6: Experiment setup of the RIS-assisted method.

Table 5.3: Experimental settings

Par.	**Value**	**Description**
F_c	3.75 GHz	Carrier frequency
T_x $(Gain)$	20 dB	The transmitter gain
R_x $(Gain)$	5 dB	The receiver gain
N	64, 128, 256	Number of subcarriers
CP length	16, 32, 64	The cyclic prefix length
SR	200 KHz	The sampling rate for the T_x and R_x
$Antennas\ types$	Horn	T_x and R_x antennas types
$T_x \leftrightarrow RIS$	3 meters	The distance between the T_x and RIS
$RIS \leftrightarrow R_x$	9 meters	The distance between the RIS and R_x

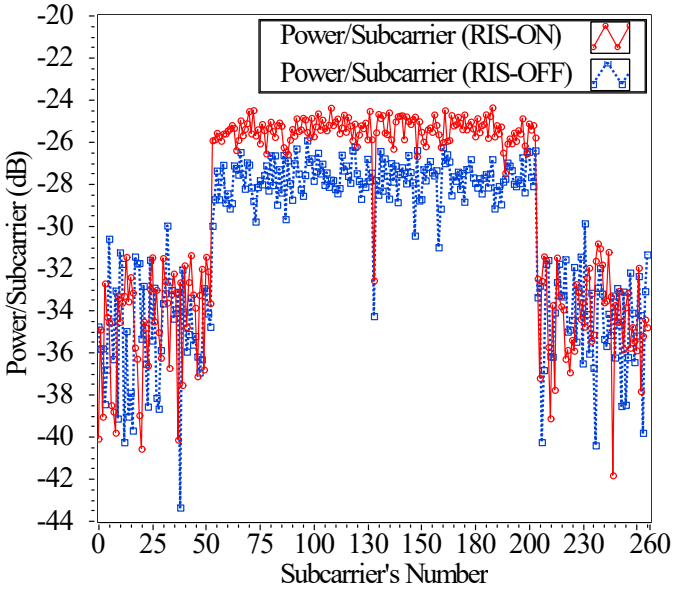


Figure 5.7: The received symbol's power for each subcarrier at $N = 256$ subcarriers.

increased by approximately 2 dB with the activation of the RIS. This improvement is significant, especially for NLoS scenarios. Fig. 5.8 shows the PDF for hypothesis H_0 when the RIS is ON and OFF and for hypothesis H_1 for $N = 64$ subcarriers and SNR = 5 dB. The figure demonstrates that the activation of the RIS results in a reduction of the mean value for PDF$|_{H_0}$ compared to when the RIS is off. This reduction leads to a decrease in the overlap between PDF$|_{H_0}$ and PDF$|_{H_1}$, providing superior ROC curves under low SNR conditions.

Fig. 5.9 illustrates the ROC curve for varying SNR values SNR $\in \{0, -3, -6\}$ dB, $N = 64$ subcarriers, and with and without the use of the RIS. The figure demonstrates that decreasing the SNR value reduces P_d for a given P_{fa}. This result arises from the increasing overlap between both hypotheses as the SNR decreases. Furthermore, the figure indicates that activating the RIS leads to improved ROC curves. For example, when the RIS is off, the P_d is approximately 0.92,

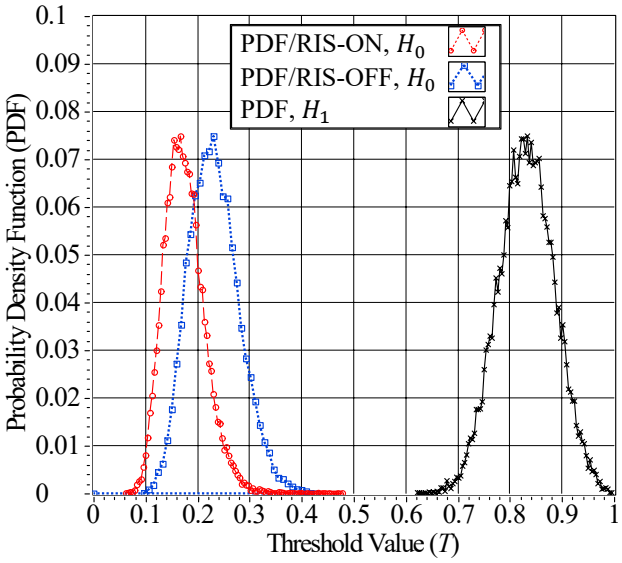
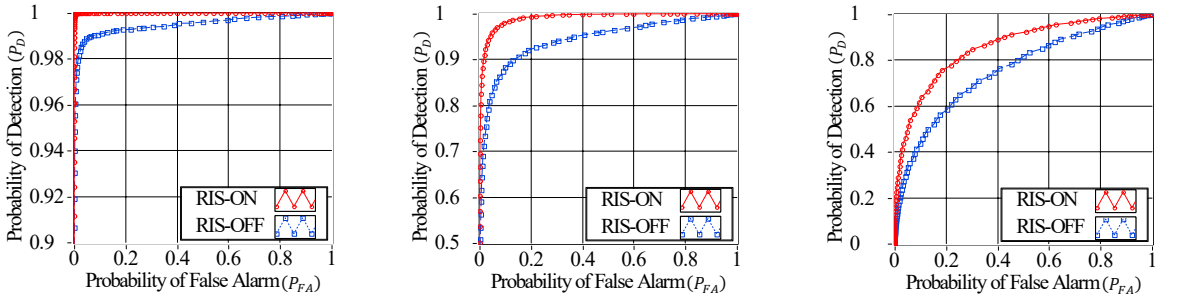


Figure 5.8: Distributions of both hypotheses $H_{0,1}$ with and without the RIS for $N = 64$ subcarriers and SNR = 5 dB.

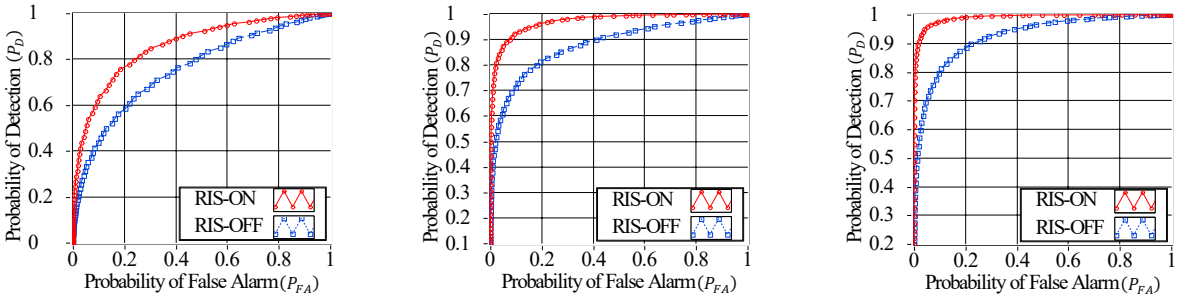


(a) ROC with/without RIS at SNR = 0 dB (b) ROC with/without RIS at SNR = -3 dB (c) ROC with/without RIS at SNR = -6 dB

Figure 5.9: The ROCs with and without the RIS at different SNRs and $N = 64$ subcarriers.

as shown in Fig. 5.9(b). However, with the RIS enabled, the P_d increases to approximately 0.99 for $P_{fa} \sim 0.2$, thereby demonstrating the ability of the RIS to enhance the authentication performance.

Additionally, the ROC for different numbers of subcarriers $N = \{64, 128, 256\}$ is evaluated for a fixed SNR value of -6 dB, as presented in Fig. 5.10. Since v in (5.4) represents the circular variance of a specific number of $N_2 = \frac{3N}{4}$ values, it follows the CLT. Hence, increasing the number of subcarriers results in an increase in N_2, which reduces the variance of $\mathscr{F}(x)|_{H_0}$ and minimises the overlap with $\mathscr{F}(x)|_{H_1}$, thereby improving the authentication performance. The enhanced ROC curves obtained in Fig. 5.10 affirm the effectiveness of increasing the number of subcarriers. The increase in the number of subcarriers (N) directly correlates with the elevation of the value of N_2. As v in (5.4) relies on the circular variance derived from a finite set of N_2 samples, it adheres to the CLT. Consequently, as the count of N_2 increases, it reduces the variance within its Gaussian distribution for H_0, thereby minimising its overlap with the distribution for H_1. This, in turn, enhances the ROC. Moreover, activating the RIS leads to an increase in the P_d



(a) ROC with/without RIS at $N = 64$ (b) ROC with/without RIS at $N = 128$ (c) ROC with/without RIS at $N = 256$

Figure 5.10: The ROCs with and without the RIS at different numbers of subcarriers and SNR $= -6$ dB.

for a given P_{fa}. As shown in Fig. 5.10(b), when the RIS is off, the P_d is roughly 0.82. However, with the RIS enabled, the P_d increases to approximately 0.96 for $P_{fa} \sim 0.2$, thus demonstrating the beneficial impact of the RIS in enhancing authentication performance.

5.3.3 Comparison of computation and communication costs

This subsection presents the computation and communication analyses of the proposed method and shows that it outperforms traditional approaches.

Comparison of computation cost

This part provides a detailed analysis of the computation comparison. Table 5.4 provides a summary of the running time for various crypto-based operations measured in [156] using the MIRACL cryptographic library [126] and a device equipped with an Intel Core $I7-6700$ processor. In Table 5.4, the notations $\{T_{sm}^{BP}, T_{pa}^{BP}\}$ and $\{T_{sm}^{ECC}, T_{pa}^{ECC}\}$ denote the computational time for the BP-based and ECC-based scale multiplication and point addition, respectively. Furthermore, the computational time for the mapping operation $T_{\mathscr{M}}$ and the circular variance operation in (5.5), denoted as $T_{c.var}$, is evaluated. The latter is insignificant compared to the values presented in Table 5.4. Consequently, these results have been incorporated to accurately quantify

Table 5.4: The time required for various crypto operations

Symbol	The operation definition	Run time
T_{sm}^{BP}	BP-based scale multiplication in $\mathbb{G}_1$	0.6940
T_{pa}^{BP}	BP-based point addition in $\mathbb{G}_1$	0.0018
T_{sm}^{ECC}	ECC-based scale multiplication in $\mathbb{G}$	0.3218
T_{pa}^{ECC}	ECC-based point addition in $\mathbb{G}$	0.0024
T_h	One way hashing operation	0.0010

Table 5.5: Computation and communication comparisons

Schemes	Computation cost (*msec*)		Communication cost (*bytes*)
	Signature generation	Signature verification of n messages	
Cui et al. [54]	$3T_{sm}^{ECC}+3T_h$	$(n+2)T_{sm}^{ECC}+(n-1)T_{pa}^{ECC}+(2n)T_h$	$124n$
Wang et al. [61]	$2T_{sm}^{BP}+2T_{pa}^{BP}+T_h$	$(3n+2)T_{sm}^{BP}+(2n)T_{pa}^{BP}+(n)T_h$	$300n$
Li et al. [60]	$3T_{sm}^{BP}+2T_{pa}^{BP}+T_h$	$(3n+2)T_{sm}^{BP}+(3n)T_{pa}^{BP}+(n)T_h$	$408n$
Proposed	$2T_{sm}^{ECC}+\lceil\frac{n}{Q}\rceil(2T_{sm}^{ECC}+T_h)+n(T_h+T_{\mathscr{M}})$	$2T_{sm}^{ECC}+\lceil\frac{n}{Q}\rceil(T_{sm}^{ECC}+T_h)+n(T_h+T_{\mathscr{M}}+T_{c.var})$	$148+112n$

the total computation cost of the proposed method and ensure a fair comparison, as listed in Table 5.5.

In the proposed scheme, the EC signature generation process incurs a cost of approximately $1T_{sm}^{ECC}$, while the verification process costs $2T_{sm}^{ECC}$. Based on this, the computation cost of transmitting n messages from a single vehicle using the proposed method can be expressed as $[2T_{sm}^{ECC}+\lceil\frac{n}{Q}\rceil(2T_{sm}^{ECC}+T_h)+n(T_h+T_{\mathscr{M}})]$. The first term accounts for the signature generation and the secret key agreement, the second term accounts for the dynamically updating pseudo-identity after every Q transmitted messages, and the third term accounts for generating $\sigma_{V_i}^{PHY}$. On the other hand, the verification time can be expressed as $[2T_{sm}^{ECC}+\lceil\frac{n}{Q}\rceil(T_{sm}^{ECC}+T_h)+n(T_h+T_{\mathscr{M}}+T_{c.var})]$. The first term corresponds to the initial signature verification, and the second and third terms verify the pseudo-identity for every Q transmitted message and $\sigma_{V_i}^{PHY}$, respectively. Thus, the total computation cost can be expressed as $(0.6436+0.3228\lceil\frac{n}{q}\rceil+0.001n)$ *msec*.

In Cui et al. [54], the computation cost for verifying n received messages is $[(n+2)T_{sm}^{ECC}+(n-1)T_{pa}^{ECC}+(2n)T_h]=(0.6412+0.3262n)$ *msec*, while for Wang et al. [61] and Li et al. [60], this value is $[(3n+2)T_{sm}^{BP}+(2n)T_{pa}^{BP}+(n)T_h]=(1.388+2.0866n)$ *msec* and $[(3n+2)T_{sm}^{BP}+(3n)T_{pa}^{BP}+(n)T_h]=(1.388+2.0884n)$ *msec*, respectively. To illustrate the comparison, Fig. 5.11 displays the computation cost required to verify 1000 received messages from a single user. The proposed scheme exhibits the lowest computation cost compared to its best competitors.

Comparison of communication cost

This part provides a detailed comparison of communication costs. For the 80-bit security level of the proposed scheme, the elliptic curve group is denoted as $\mathbb{G}$, where $|\mathbb{G}|=40$ bytes and $Z_q^*=20$ bytes. For the same security level, the bilinear pairing is denoted as $\bar{E}:\mathbb{G}_1\times\mathbb{G}_1\rightarrow\mathbb{G}_T$, where $\bar{P}$ is the generator of the elliptic curve $\bar{E}:y^2=x^3+x \bmod \bar{p}$, with $|\mathbb{G}_1|=128$ bytes and $Z_{\bar{q}}^*=20$ bytes. Moreover, the size of hashed values using the SHA-1 hashing operation is 20 bytes, and the timestamp has a size of 4 bytes.

In the proposed scheme, the communication cost of transmitting n messages is determined by the size of the tuple $\langle TID_{V_i},T_1,(PK_{V_i},T_R,\sigma_{TA}),\sigma_{V_i}\rangle$ during the first transmission slot, as well as the size of the tuple $\langle PID_{V_i},A_1,T_3,\sigma_{V_i}^{PHY}\rangle$ for n subsequent transmissions. Specifically, $\{PK_{V_i},A_1\}\in\mathbb{G}$, and the length of TID_{V_i} and PID_{V_i} is 20 bytes each. The sizes of

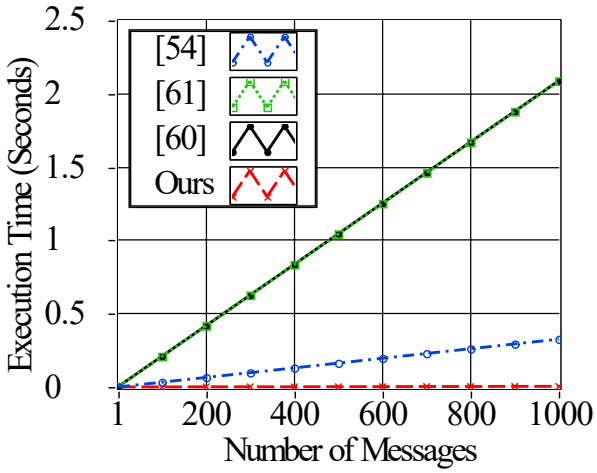


Figure 5.11: The computation cost of verifying 1000 messages at $Q = 100$.

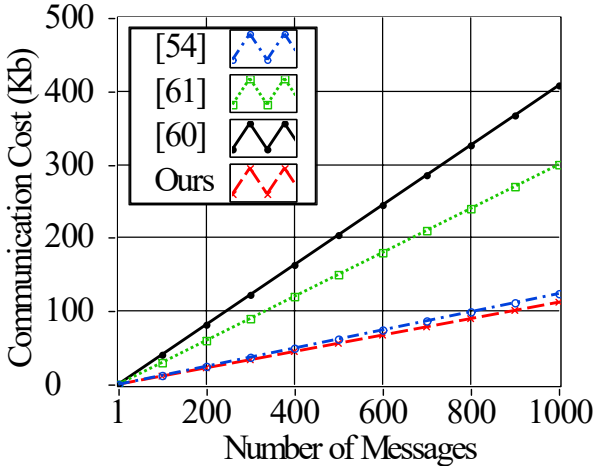


Figure 5.12: The communication cost of sending 1000 messages.

σ_{TA} and σ_{V_i} are 40 bytes each, while the lengths of T_R, T_1, and T_3 are 4 bytes each. The size of $\sigma_{V_i}^{PHY}$ is 48 bytes. Therefore, the total communication cost for transmitting n messages is $[(20+2\times4+3\times40)+(20+40+4+48)n] = (148+112n)$ bytes. In Cui et al. [54], the signature is represented by the tuple $\langle PID_j^1, PID_j^2, \delta_j, D_j, T_j\rangle$, where $\{PID_j^1, D_j\} \in \mathbb{G}$, $\{PID_j^2, \delta_j\} \in Z_q^*$, and T_j denotes the timestamp. Thus, the total size of the signature is $(2\times40+2\times20+4) = 124$ bytes. In Wang et al. [61], the signature is represented by the tuple $\langle R_{u_i}, T'_{u_i}, \rho_{u_i}, PK_{TA}, t_i\rangle$, where $\{R_{u_i}, T'_{u_i}\} \in \mathbb{G}_1$, $\{\rho_{u_i}, PK_{TA}\} \in Z_{\bar{q}}^*$, and t_i represents the timestamp. Thus, the total size of the signature is $(2\times128+2\times20+4) = 300$ bytes. Similarly, Li et al. [60] represent a signature as $\langle R_{u_i}, K'_{u_i}, KG'_{u_i}, \rho_{u_i}, t_i\rangle$, where $\{R_{u_i}, K'_{u_i}, KG'_{u_i}\} \in \mathbb{G}_1$, $\rho_{u_i} \in Z_{\bar{q}}^*$, and t_i denotes the timestamp. The total size of this signature is $(3\times128+20+4) = 408$ bytes. Fig. 5.12 shows the communication cost required for transmitting 1000 messages received from a single user. The figure reveals that the proposed scheme exhibits the lowest communication cost compared to its best rivals.

5.4 Summary

This chapter proposes an authentication scheme that utilises the RIS to enhance the detection probability of the PHY-layer authentication in NLoS conditions while still adhering to the security and privacy requirements of VANETs. The theoretical and experimental results demonstrate the effectiveness of the RIS in improving authentication performance. Informal and formal (BAN-logic) analyses are conducted to verify the scheme's security resistance against typical attacks. Additionally, computation and communication comparisons are conducted to demonstrate that the proposed method effectively reduces the overheads, resulting in a computation cost savings of over 98% compared to existing methods in [54, 60, 61], and communication cost savings of approximately 10%, 62%, and 72% compared to [54], [61], and [60], respectively. The next chapter examines the advantages of integrating RIS technology into the PHY-layer secret key extraction process, with a particular focus on enhancing the key generation rate for NLoS communication scenarios, while also strengthening the system's ability to resist DoS attacks.

Chapter 6

RIS Enabled Secret Key Generation

In recent years, some researchers have applied RISs to the physical layer security of wireless communications with the goal of improving the secrecy data rate under the wiretap channel, a concept referred to as keyless information theory security [161]. The application of RISs to phase-based key extraction mechanisms has not yet been explored. The effective extraction of keys with the aid of the RIS and the utilisation of channel phase responses to generate shared keys remains an open issue in the field of physical layer security. Given the ability of RISs to configure the wireless channel in real-time through passive reflection, they have the potential to improve secret key capacity significantly. Furthermore, flooding attacks are a potential hazard, whereby the attacker floods the network with a substantial volume of simultaneous communication requests, thus constituting a DoS attack [162]. By strengthening the signal from a certain side (i.e., a legitimate user) while weakening it from another (i.e., the adversary), the RIS can help mitigate the effect of such attacks. An effective way to accomplish this is to configure the RIS elements in a way that can destructively interfere in one direction, and constructively interfere in another.

The following summarises the contributions of this chapter which are published in [23], fulfilling the outlined thesis objectives (2, 3, 4, 5, 6, 7) detailed in Subsection 1.4.3:

1. This study extends the work introduced in Chapter 4 by proposing a RIS-assisted key extraction method that enhances the signal strength for the designated user's location while reducing that from the active attacker's location. Hence, the proposed method improves the key extraction performance for designated users while mitigating the impact of DoS attacks within the network.

2. To accomplish this, a RIS configuration optimisation algorithm is designed using the Hadamard matrix codebook. This algorithm optimises the measurement quantisation order based on the optimal configuration's average SNR estimate.

3. The proposed RIS optimisation algorithm is practically implemented using a 1-bit RIS with 64×64 elements and two USRPs operating in the 5G communication frequency

range (3.75 GHz). Finally, the statistical randomness of the extracted keys is measured to demonstrate the extracted keys' suitability for use as cryptographic keys.

The structure of this chapter is as follows: Section 6.1 presents the preliminary concepts required for this research. Section 6.2 presents the proposed RIS-assisted key extraction method. Section 6.3 analyses the hardware implementation of the method. Finally, Section 6.4 summarises the findings and contributions of this work.

6.1 Preliminaries and theoretical concepts

This section provides a brief overview of the secret key extraction process in Chapter 4. A thorough discussion of the considered system model is also provided. The notations used in this chapter are summarised in Table 6.1 for better readability.

6.1.1 Review of the PHY-layer secret key extraction scheme

The work introduced in Chapter 4 proposes a novel Diffie-Hellman channel probing mechanism that utilises the extended Chebyshev chaotic mapping operation to exchange probing signals in an interleaved fashion. Specifically, the extended Chebyshev mapping operation for the OFDM system of N subcarriers is formulated as:

$$T'_{n_i}(\theta_i) = \begin{cases} n_i \cdot \theta_i \bmod p, & \theta_i \in [0, 2\pi) \\ n_i \cdot \cos^{-1}(x_i) \bmod p, & x_i = \cos(\theta_i) \end{cases} \quad \text{for } i = 1, \ldots, N, \tag{6.1}$$

Table 6.1: List of notations for the proposed RIS-assisted key extraction technique

Symbol	Definition
θ_i	The generator of the cyclic group $\mathbb{G}$ for the i^{th} subcarrier of the OFDM symbol
n_i, m_i	The private integer numbers at the sides of Alice and Bob, respectively
Δt	The transmission time interval between two subsequent OFDM symbols
$\lvert h_i \rvert, \xi_i$	The wireless channel amplitude and phase responses, respectively
$\hat{T}'_{m_i}(\theta_i), \hat{T}'_{n_i}(\theta_i)$	The equalised phase estimates at the sides of Alice and Bob, respectively
$\mathscr{M}^{-1}$	The Gray code mapping operation that converts final estimates into bit streams
r	The order of the generator θ_i and the mapping operation $\mathscr{M}^{-1}$
$\phi(x), \mathrm{erf}(z)$	The cumulative distribution function and the error function
P_e	The probability of error in the extracted key between two parties
I	The secret key capacity
HD	The Hadamard codebook used for optimising the RIS configuration
$\overline{\mathrm{SNR}}_i^{Bob}, \overline{\mathrm{SNR}}_i^{Eve}$	The average SNR of the signals transmitted from Bob and Eve, respectively
H_{opt}	The optimal configuration for the RIS's reflecting units

where $p = 2\pi$, n_i is a large integer number, and $\theta_i = \frac{2\Pi}{2^r}$ for $r \in \{1,2,3\}$ is the primitive root of the i^{th} subcarrier. The primitive root θ_i is a generator of the group $\mathbb{G}$ such that its multiples generate the entire group. For example, let $r = 2$, then $\theta_i = \frac{\Pi}{2}$. Thus, the cyclic group elements are $\mathbb{G}_2 = \{0, \frac{\Pi}{2}, \Pi, \frac{3\Pi}{2}\}$. While for $r = 3$, $\theta_i = \frac{\Pi}{4}$. Thus, the cyclic group elements are $\mathbb{G}_3 = \{0, \frac{\Pi}{4}, \frac{\Pi}{2}, \frac{3\Pi}{4}, \Pi, \frac{5\Pi}{4}, \frac{3\Pi}{2}, \frac{7\Pi}{4}\}$. A scenario has been considered where two parties (Alice and Bob) are in the same communication range and want to establish a secure communication link. In this context, Alice and Bob exchange authenticated probing packets at times t_0 and t_1, respectively. Based on the received probing packets, both terminals can extract a high entropy secret key, which is used to secure subsequent transmissions using the upper layer's crypto-based approaches. Fig. 6.1 reviews the steps involved in the secret key extraction process. Generally, the extraction process comprises channel probing and quantisation, information reconciliation, and privacy amplification. In the former, Alice sends the probing packet in the form of two OFDM symbols of N subcarriers, which can be represented in a simplified form as:

$$\begin{aligned} s_a(t_0) &= \sum_{i=1}^{N} \sqrt{\frac{2E_S}{T}} e^{j\left(T'_{2n_i}(\theta_i)\right)} = \sum_{i=1}^{N} \sqrt{\frac{2E_S}{T}} e^{j(2n_i\theta_i)} \\ s_a(t_0 + \Delta t) &= \sum_{i=1}^{N} \sqrt{\frac{2E_S}{T}} e^{j\left(T'_{n_i}(\theta_i)\right)} = \sum_{i=1}^{N} \sqrt{\frac{2E_s}{T}} e^{j(n_i\theta_i)}, \end{aligned} \tag{6.2}$$

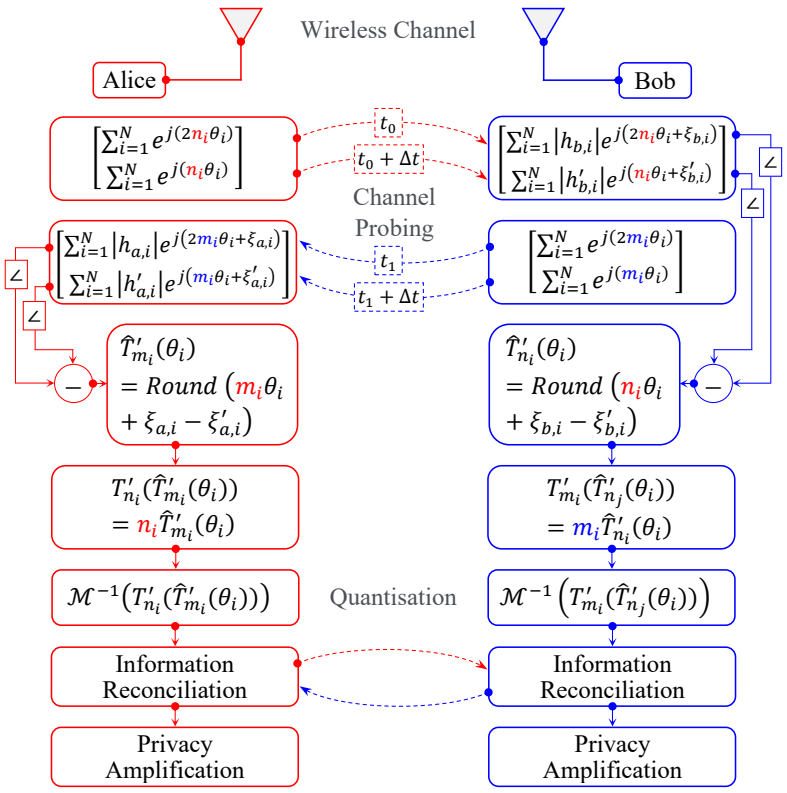


Figure 6.1: The PHY-layer secret key extraction scheme in a noiseless channel.

where the transmission time interval between both OFDM symbols is $\Delta t \leq T_c$. Thus, Bob's received signal can be expressed as

$$\begin{aligned} r_b\left(t_0'\right) &= \sum_{i=1}^{N} \sqrt{\frac{2\left|h_i\right|^2 E_s}{T}} e^{j\left(T'_{2n_i}(\theta_i)+\xi_{b,i}\right)} + N_i \\ r_b\left(t_0'+\Delta t\right) &= \sum_{i=1}^{N} \sqrt{\frac{2\left|h_i'\right|^2 E_s}{T}} e^{j\left(T'_{n_i}(\theta_i)+\xi'_{b,i}\right)+} N_i', \end{aligned} \tag{6.3}$$

where $\{|h_i|, |h_i'|\}$ and $\{\xi_i, \xi_i'\}$ are the channel fading coefficients and phase responses of the i^{th} subcarrier at times $\{t_0', t_0'+\Delta t\}$, respectively and $\{N_i, N_i'\}$ are complex additive Gaussian noises $\mathscr{CN}\left(0, \sigma_n^2\right)$ with zero means and σ_n^2 variances. It is noteworthy to mention that the channel responses $\{|h_i|, \xi_i\}$ are highly correlated with $\{|h_i'|, \xi_i'\}$ for $\Delta t \leq T_c$. Similarly, Bob replies by sending an authenticated probing packet as in (6.2) with phases $\{T'_{2m_i}(\theta_i), T'_{m_i}(\theta_i)\}$ at times $\{t_1', t_1'+\Delta t\}$. Then, both terminals, Alice and Bob, equalise their received signals by computing $e_a(t) = r_a(t_1')\, r_a(t_1'+\Delta t)^*$ and $e_b(t) = r_b\left(t_0'\right) r_b\left(t_0'+\Delta t\right)^*$, respectively. Hence, the phases of $e_a(t)$ and $e_b(t)$ of the i^{th} subcarrier can be formulated as

$$\begin{aligned} \angle e_{a,i}(t) &= m_i\theta_i + \left(\xi_{a,i} - \xi'_{a,i}\right) + \left(\omega_{a,i} - \omega'_{a,i}\right) \\ \angle e_{b,i}(t) &= n_i\theta_i + \left(\xi_{b,i} - \xi'_{b,i}\right) + \left(\omega_{b,i} - \omega'_{b,i}\right), \end{aligned} \tag{6.4}$$

where $\{\omega_{a,i}, \omega'_{a,i}\}$ and $\{\omega_{b,i}, \omega'_{b,i}\}$ are the noisy added estimates result from $\{N_i, N_i'\}$ in (6.3) at the sides of Alice and Bob, respectively with Gaussian distributions $\mathscr{N}\left(0, \sigma^2\right)$. Accordingly, both terminals use the *Round* function to obtain $\hat{T}'_{m_i}(\theta_i)$ and $\hat{T}'_{n_i}(\theta_i)$ as

$$\begin{aligned} \hat{T}'_{m_i}(\theta_i) &= Round\left(\angle e_{a,i}(t)\right) = Round\left(m_i\theta_i + \left(\xi_{a,i} - \xi'_{a,i}\right) + \left(\omega_{a,i} - \omega'_{a,i}\right)\right) \\ \hat{T}'_{n_i}(\theta_i) &= Round\left(\angle e_{b,i}(t)\right) = Round\left(n_i\theta_i + \left(\xi_{b,i} - \xi'_{b,i}\right) + \left(\omega_{b,i} - \omega'_{b,i}\right)\right), \end{aligned} \tag{6.5}$$

where the function $Round(x)$ is used to round x to the nearest multiple of $2\pi/2^r$ for $r \in \{1,2,3\}$. Then, Alice and Bob compute $T'_{n_i m_i}(\theta_i)\,|_{Alice} = T'_{n_i}\left(\hat{T}'_{m_i}(\theta_i)\right)$ and $T'_{n_i m_i}(\theta_i)\,|_{Bob} = T'_{m_i}\left(\hat{T}'_{n_i}(\theta_i)\right)$, respectively. The use of the *Round* function in the context is important to avoid the significant error results from multiplying the negligible value of $((\xi - \xi') + (\omega - \omega'))$ by the large integer number n_i or m_i. Finally, both terminals quantise their estimates to convert them into bit streams using a mapping operation $\mathscr{M}^{-1}(.)$ of order r. For clarity, a Gray code mapping operation of

order 2 can be expressed as

$$\mathscr{M}^{-1}\left(T'_{n_i m_i}(\theta_i)\right) = \begin{cases} 00 & T'_{n_i m_i}(\theta_i) \in [-\frac{\pi}{4}, \frac{\pi}{4}) \\ 01 & T'_{n_i m_i}(\theta_i) \in [\frac{\pi}{4}, \frac{3\pi}{4}) \\ 11 & T'_{n_i m_i}(\theta_i) \in [\frac{3\pi}{4}, -\frac{3\pi}{4}) \\ 10 & T'_{n_i m_i}(\theta_i) \in [-\frac{3\pi}{4}, -\frac{\pi}{4}) \end{cases} \quad \text{for } i = 1, \ldots, N. \tag{6.6}$$

Note that the higher the variance σ^2 of the phase noisy estimates in (6.4), the lower the quantisation order r, and vice versa.

6.1.2 System modelling

In this study, the vehicular communication network comprises the following entities, as shown in Fig. 6.2.

1. *The RSU*: RSUs are stationary devices located along roads that facilitate wireless communication between themselves and surrounding vehicles within a particular range. Each RSU acts as a relay between vehicles, extending the communication range and improving the network's reliability. It is equipped with wireless communication capabilities and can support various applications, such as traffic management, safety warnings, and entertainment services. It also has a reliable communication link with the RIS's intelligent controller, so configurations of reflecting units can be optimised. Through this mechanism, the RSU effectively manages the RIS to enhance the transmission of signals towards a designated direction while simultaneously reducing the strength of signals toward potential unauthorised interceptors, commonly referred to as "Eve."

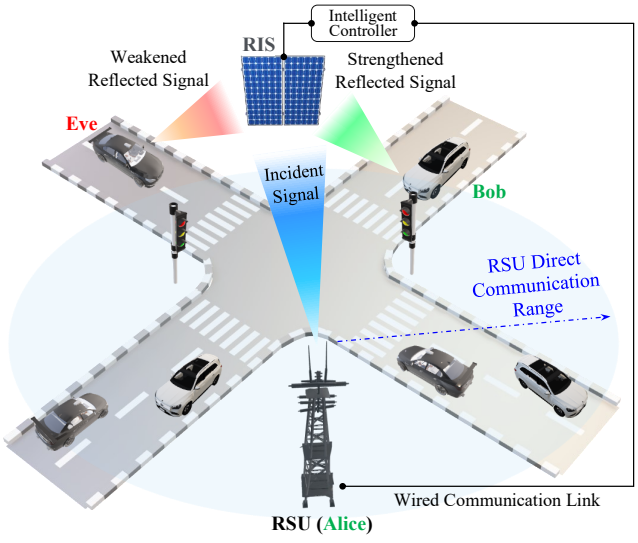


Figure 6.2: System modelling for the proposed RIS-assisted key extraction technique.

2. *The vehicle's OBUs*: OBU is a communication device installed within each vehicle in the network. It can communicate with other OBUs and RSUs within range, facilitating the exchange of traffic-related messages in 100-300 *msec* intervals based on the dedicated short-range communication protocol [137]. In this way, OBUs play a crucial role in the functioning of the vehicular network.

3. *The RIS*: RISs are intelligent surfaces that can dynamically change their electromagnetic behaviours to improve the performance of wireless networks. RISs can be used to manipulate the propagation of radio signals, allowing for better signal quality, increased network coverage, and improved energy efficiency. The intelligent controller is an integral component of each RIS. It manages and configures the multiple meta-surface reflecting units (RUs) of order N elements that make up the RIS. It plays a crucial role in optimising the performance of the RIS in the network.

4. *The adversary Eve*: "Eve" is an active attacker who overloads the network with excessive traffic, causing it to become unavailable to legitimate users. In this attack, the adversary overwhelms the target's resources and prevents it from functioning properly, thereby denying service to its intended users. By constructing and launching a flooding DoS attack, the attacker aims to disrupt the system's normal functioning and cause inconvenience or harm to its users.

6.2 RIS-assisted secret key extraction method

This section shows how the RIS improves the key extraction performance and reduces the impact of potential flooding-based DoS attacks on the network.

6.2.1 Performance optimisation

Three critical evaluation metrics must be considered while optimising the key extraction performance, namely the BGR, the BMR, and the SBGR. The BGR is a measure of the efficiency of this process and typically represents the number of generated bits per channel sample, expressed as:

$$BGR = \frac{\text{Total extracted bits}}{\text{Channel sample}}, \tag{6.7}$$

A high valuation of the BGR indicates a more efficient extraction process and a higher rate of secret bit generation, resulting in improved security and faster key establishment for the communication system. On the other hand, the BMR represents the number of mismatched bits extracted from each channel sample, expressed as:

$$BMR = \frac{\text{No. of mismatched bits}}{\text{Channel sample}}, \tag{6.8}$$

The SBGR is defined as the number of matched bits, which is represented as $SBGR = BGR - BMR$. Hence, the SBGR considers both the BGR and the BMR in the process of secret key extraction. For negligible channel phase decorrelation $(\xi - \xi') \approx 0$, the phase distribution of the equalised signal $\angle e(t)$ in (6.4) is normally distributed with means $\{T'_{n_i}(\theta_i) = n_i\theta_i, T'_{m_i}(\theta_i) = m_i\theta_i\}$ and variance $2\sigma^2$ for {Alice, Bob}, respectively. Thus, its CDF is approximated as:

$$\begin{aligned} \phi(x) &= \frac{1}{2}\left[1 + \operatorname{erf}\left(\frac{x - T'_{n_i(m_i)}(\theta_i)}{2\sigma}\right)\right], \\ \operatorname{erf}(z) &= \frac{2}{\sqrt{\pi}}\int_0^z e^{-t^2}dt \end{aligned} \tag{6.9}$$

where $\operatorname{erf}(z)$ is the error function. Thus, the probability of error P_e is the probability of the estimated $\angle e(t)$ in (6.4) to be out of the interval $\left[T'_{n_i(m_i)} + \frac{\pi}{2^r}, T'_{n_i(m_i)} - \frac{\pi}{2^r}\right)$, which can be represented by:

$$P_e = 2\phi\left(T'_{n_i(m_i)}(\theta_i) - \frac{\pi}{2^r}\right). \tag{6.10}$$

Accordingly, the communicating terminals can agree on the optimum quantisation order $r \in \{1,2,3\}$ for an acceptable $P_e \leq a_1$ as:

$$x = \arg\max_{x'} \operatorname{erf}\left(\frac{x' - T'_{n_i(m_i)}(\theta_i)}{2\sigma}\right) \leq a_1 - 1. \tag{6.11}$$

Based on x, r is optimised as:

$$r = \arg\max_{r'} 2^{r'} \leq \frac{\pi}{x} \quad \text{for } r' = 1,2,3. \tag{6.12}$$

6.2.2 Channel modelling

The scenario depicted in Fig. 6.3 involves the concurrent processes of communication establishment between Bob and Alice, and Eve's deliberate disruption of network integrity through the inundation of the network with excessive communication requests. In this scenario, the RSU has the capability to manage the RIS and optimise its configuration to reinforce the signal in the direction of the intended recipient "Bob", while simultaneously mitigating the strength of the signals received from the adversary "Eve". Hence, the signals received by Alice from both Bob

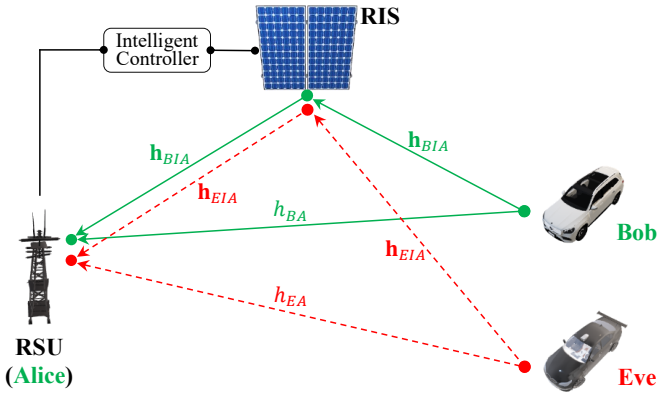


Figure 6.3: RIS assisted channel modelling.

and Eve can be theoretically formulated as follows:

$$\begin{aligned} y_A|Bob &= (h_{BA} + h_{BIA})\,x + N_A \\ &= \left(h_{BA} + \sum_{i=1}^{N} h_{BIA}^{i}\beta_i\Psi_i\right)x + N_A \\ y_A|Eve &= (h_{EA} + h_{EIA})\,x + N_A \\ &= \left(h_{EA} + \sum_{i=1}^{N} h_{EIA}^{i}\beta_i\Psi_i\right)x + N_A, \end{aligned} \tag{6.13}$$

where N_A is the complex additive Gaussian noise $\mathscr{CN}\left(0,\sigma_n^2\right)$, $\{h_{BA},h_{EA}\}$ are the channel responses in the complex form of the direct link from (Bob $\rightarrow$ Alice) and (Eve $\rightarrow$ Alice), respectively, and $\{h_{BIA},h_{EIA}\}$ are the superposition of the N channel multipath components of the RIS's elements of the indirect link from (Bob $\rightarrow$ RIS $\rightarrow$ Alice) and (Eve $\rightarrow$ RIS $\rightarrow$ Alice), respectively. Additionally, the configuration of the RIS is represented by the variable $H = [\beta_1\Psi_1,\beta_2\Psi_2,\ldots,\beta_N\Psi_N]^T$, where $\{\beta_i,\Psi_i\}$ defines the state of each RIS element. An example of a 1-bit RIS can be described as follows: the phase shift applied by each unit cell, denoted by Ψ_i, is equal to Π, and the reflection coefficient, represented by β_i, is a binary variable that can take on values of either 0 or 1.

The use of the RIS helps increase the secret key capacity I, which refers to the maximum amount of information that can be securely extracted from the physical layer of a communication system and used as a secret key. By properly designing and controlling the phase shifts applied by the RIS, the RIS can counter the effects of fading and interference in the channel, which can also result in higher secret key capacities. Therefore, the RIS can be seen as a valuable tool for improving the secret key capacity I in the presence of an eavesdropper and ensuring secure communication. The work in [161] provides a theoretical formulation for the secret key capacity denoted by:

$$I = \log_2 \left(1 + \frac{\left(\sigma^2_{h_{BA}} + \sum_{i=1}^{N} \beta_i^2 \sigma^2_{h^i_{BIA}} \right)^2 / \sigma_n^4}{1 + 2 \left(\sigma^2_{h_{BA}} + \sum_{i=1}^{N} \beta_i^2 \sigma^2_{h^i_{BIA}} \right) / \sigma_n^2} \right). \tag{6.14}$$

The RIS can adjust signal directionality, consequently reducing the signal strength from Eve's direction and enhancing the signal coming from Bob. This can be achieved by adjusting the phase shifts applied by each unit cell of the RIS so that the reflection coefficients of the unit cells constructively interfere in certain directions and destructively interfere in others. Therefore, the goal is to optimise the RIS configuration H to maximise the secret key capacity I in (6.14) while concurrently reducing any interference from Eve.

6.2.3 Optimising the best RIS configuration (H_{opt})

The use of the Hadamard matrix in the configuration of the RIS offers several advantages, including low complexity, high efficiency, and improved performance. This makes the Hadamard matrix effective for scenarios where reducing interference, enhancing privacy, and increasing energy efficiency are critical objectives in wireless communication systems [163]. The Hadamard matrix offers a suite of orthogonal and binary phase shift values that can be applied to the elements of the RIS to influence the reflection of incoming electromagnetic waves in a specific direction or with a preferred phase shift. The flexibility and efficacy of the Hadamard matrix in configuring the RIS to achieve these objectives while minimizing complexity makes it a promising solution for wireless communication challenges. This study involves the measurement of the average signal-to-noise ratio ($\overline{\text{SNR}}$) for every configuration (H) of the OFDM system. Based on these measurements, An optimisation method for the RIS configuration is developed, presented in Algorithm 1. This method encompasses four phases: initialisation, scanning toward Bob, scanning toward Eve, and configuration optimisation.

- *Initialisation*: Alice initialises the Hadamard codebook $HD = \sum_{i=1}^{N_x \times N_y} H_i$, where N_x and N_y are the number of elements in the RIS's x and y coordinates, respectively.

- *Scanning toward Bob*: Alice scans the average SNR value for the received OFDM symbols from Bob, denoted as $\overline{\text{SNR}}_i^{Bob}$, for each configuration H_i within the set of all possible configurations, HD, where $i = 1, 2, \ldots, N_x \times N_y$.

- *Scanning toward Eve*: Alice scans the average SNR value of the received OFDM symbols from Eve, denoted as $\overline{\text{SNR}}_i^{Eve}$, for each configuration H_i within the set of all possible configurations, HD, where $i = 1, 2, \ldots, N_x \times N_y$.

- *Configuration optimisation*: Alice computes the ratio of the average SNR for Bob $\overline{\text{SNR}}_i^{Bob}$ over the average SNR for Eve $\overline{\text{SNR}}_i^{Eve}$, denoted as C_i, for $i = 1, 2, \ldots, N_x \times N_y$. The maximum value of C_i, referred to as C_{max}, is then determined from the set of all values of C_i. The optimum configuration, denoted as H_{opt}, is identified as the configuration H_i that

corresponds to the maximum value of C_{max}. This calculation maximises Bob's average SNR while minimising Eve's average SNR.

Algorithm 1 Optimising the Best RIS Configuration (H_{opt})

Initialisation

The Hadamard codebook $HD = \sum_{i=1}^{N_x \times N_y} H_i$ for the $(N_x \times N_y)$ RIS reflecting units
Two empty variables, SNR^{Bob} and SNR^{Eve}, used to store the measured SNRs
An empty variable C

Alice is communicating with the legitimate terminal (Bob)

for $i = 1 : (N_x \times N_y)$ **do**
  Measuring the average SNR value $(\overline{\text{SNR}}_i^{Bob})$ for each Hadamard matrix (H_i)
  Appending the measured $\overline{\text{SNR}}_i^{Bob}$ to SNR^{Bob}
end for

Alice is communicating with the illegitimate terminal (Eve)

for $i = 1 : (N_x \times N_y)$ **do**
  Measuring the average SNR value $(\overline{\text{SNR}}_i^{Eve})$ for each Hadamard matrix (H_i)
  Appending the measured $\overline{\text{SNR}}_i^{Eve}$ to SNR^{Eve}
end for

Optimising the best configuration

for $i = 1 : (N_x \times N_y)$ **do**
  Computing $C_i = \frac{\overline{\text{SNR}}_i^{Bob}}{\overline{\text{SNR}}_i^{Eve}}$
  Appending the computed C_i to C
end for
Finding the best configuration $(H_{opt} = H_i)$ corresponding to $C_{max} = \max(C_i \in C)$

6.3 Hardware implementation analysis

In this section, the hardware-based experimental results for the proposed RIS-assisted secret key extraction method are presented, and the effectiveness of the optimisation approach for configuring the RIS is evaluated.

6.3.1 Experimental setup and the RIS configuration analysis

This part describes the experimental parameters/settings and then evaluates the proposed method. As depicted in Fig. 6.4, the experimental setup consists of two USRPs version Ettus X300 and a 1-bit RIS with 64×64 elements. One USRP serves as the transmitter, positioned 3 meters from the RIS, while the other USRP is equipped with two channels with horn antennas and serves as two separate receivers, representing Bob and Eve, positioned 5 meters from the RIS and situated at 45° degrees on either side of the line connecting the RIS and the first USRP. In this exper-

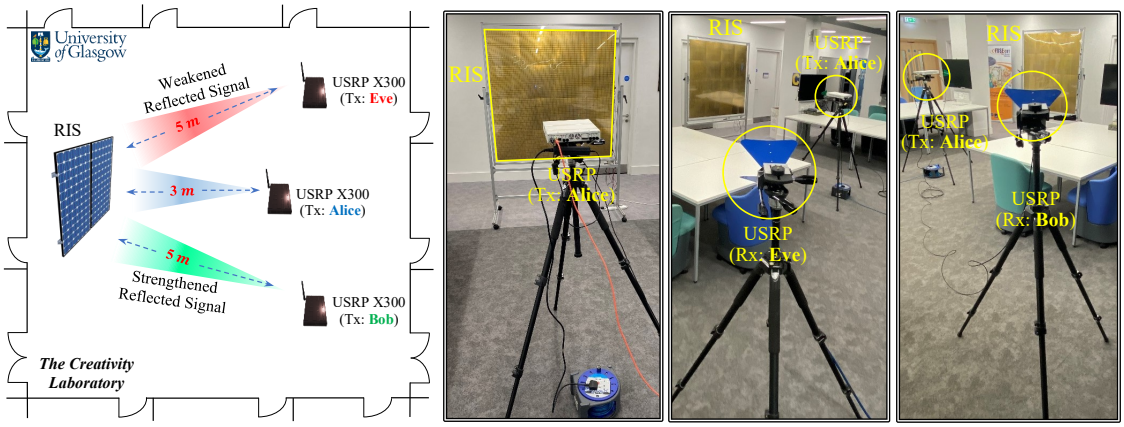
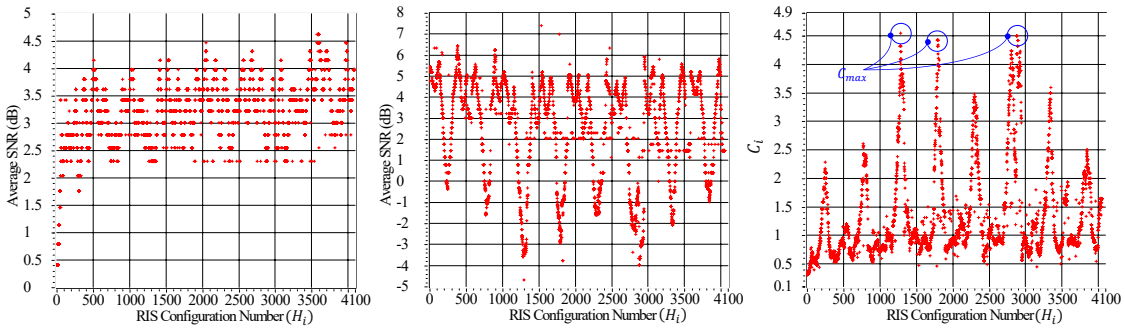


Figure 6.4: Experiment setup for the secret key generation scheme.

iment, a single antenna is installed on all terminals. The carrier frequency is set to 3.75 GHz, and the sampling rate is configured at 200 KHz for an OFDM system with 256 subcarriers.

The $\overline{\text{SNR}}_i^{Bob}$ and $\overline{\text{SNR}}_i^{Eve}$ are calculated for each configuration matrix $H_i \in HD$, where HD is the Hadamard codebook of order $|HD| = 64 \times 64 = 4096$ configurations. Fig. 6.5(a) illustrates the relationship between $\overline{\text{SNR}}_i^{Bob}$ and H_i, while Fig. 6.5(b) presents the relationship between $\overline{\text{SNR}}_i^{Eve}$ and H_i, for $i = 1, \ldots, 4096$. It can be observed from Fig. 6.5(b) that some configurations enhance the transmitted signals' received power, while others result in a reduction ranging from -3.5 dB to 6.5 dB. Algorithm (1) is applied to the estimated measurements to compute $C_i = \frac{\overline{\text{SNR}}_i^{Bob}}{\overline{\text{SNR}}_i^{Eve}}$ for each configuration, as shown in Fig. 6.5(c). This figure shows that the configurations associated with the top three peaks are good candidates for H_{opt}. Consequently, the value of C_i is maximised to determine the optimal configuration matrix H_{opt}.

Fig. 6.6 displays the impact of the RIS on the received OFDM symbols at the sides of Bob and Eve. When the RIS is activated using the optimised configuration H_{opt}, it is evident that the received power at Bob's side is boosted by approximately 2 dB compared to the scenario when the RIS is turned off. Additionally, the figure highlights the effectiveness of the RIS in reducing



(a) $\overline{SNR}_i^{Bob}$ for each configuration H_i. (b) $\overline{SNR}_i^{Eve}$ for each configuration H_i. (c) $C_i = \frac{\overline{SNR}_i^{Bob}}{\overline{SNR}_i^{Eve}}$ for each configuration H_i.

Figure 6.5: The average SNR values for different configurations and their optimised value.

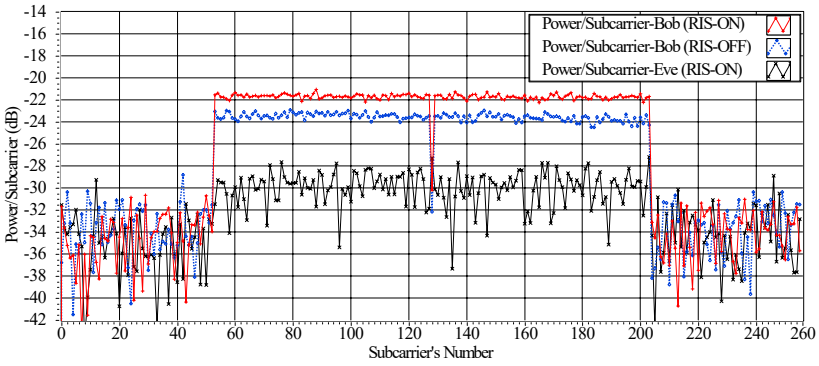


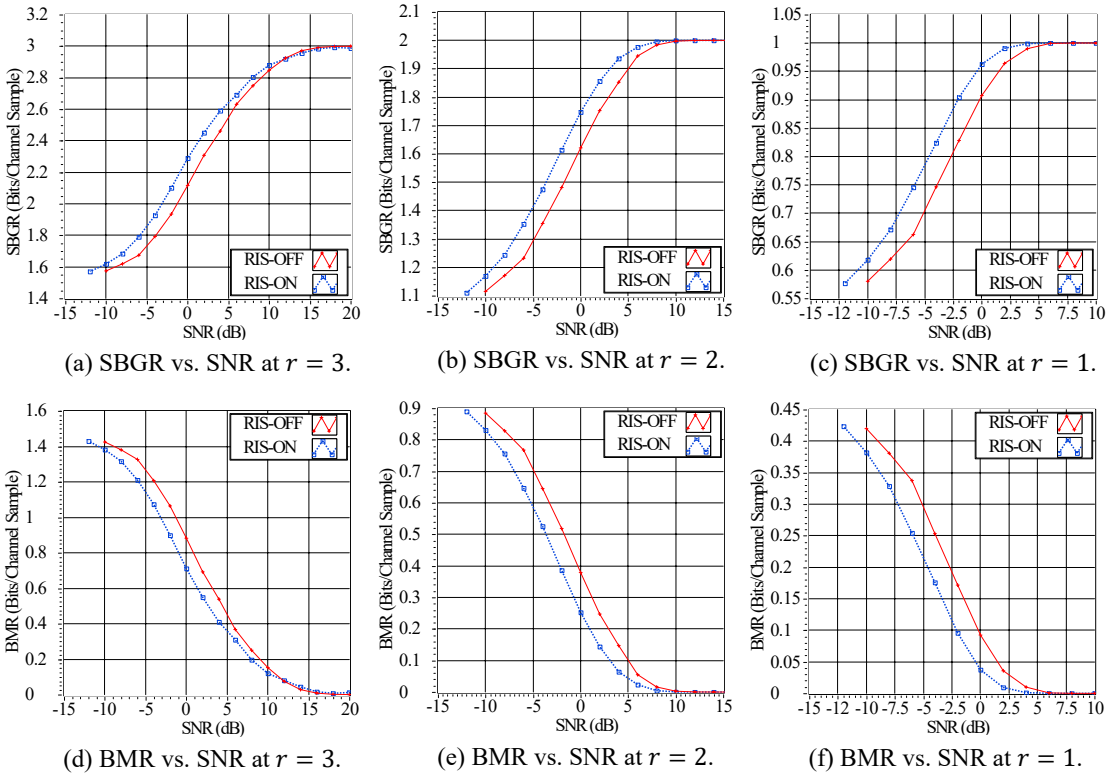
Figure 6.6: The power/subcarrier for $N = 256$ at the side of Bob and Eve, with/without the RIS.

the received power at Eve's side. This reduced received power at Eve's side effectively reduces the impact of DoS attacks carried out by Eve.

6.3.2 Implementation results and analysis of the key extraction process

The secret key extraction performance is compared under two scenarios: when the RIS is activated with the optimal configuration (H_{opt}) and when the RIS is turned off. The performance evaluation is based on the SBGR metric from (6.7) and the BMR metric from (6.8), at various SNR values and $r = \{1,2,3\}$. As presented in Fig. 6.7(a), Fig. 6.7(b), and Fig. 6.7(c), the results indicate that the SBGR improves when the RIS is activated. For instance, at an SNR of 0 dB, the SBGR increases from approximately 1.62 $bits/sample$ when the RIS is off to approximately 1.75 $bits/sample$ when the RIS is activated (see Fig. 6.7(b)). Conversely, the BMR decreases when the RIS is activated as compared to when it is kept off. For instance, at an SNR of 0 dB, the BMR drops from approximately 0.38 $bits/sample$ when the RIS is off to approximately 0.25 $bits/sample$ when the RIS is activated (see Fig. 6.7(e)). These results demonstrate the efficacy of the RIS in enhancing secret key extraction performance.

The quantisation order, r, can be optimised based on the estimated average SNR at the side of Bob, $\overline{\text{SNR}}^{Bob}_{H_{opt}}$, corresponding to the optimal configuration H_{opt}, where $\overline{\text{SNR}}^{Bob}_{H_{opt}} \in \overline{\text{SNR}}^{Bob}$ in step (6) from Algorithm (1). The optimisation range for an acceptable BMR ≤ 0.1 $bits/sample$ is presented in Table 6.2 for scenarios where the RIS is both ON and OFF. It can be inferred that the RIS is more effective in improving the system performance in scenarios with lower SNR values than in higher SNR scenarios. This suggests that the impact of the RIS on the SNR may be limited when the SNR is already high, and other factors, such as fading and shadowing, may have a more dominant impact on the system performance. For the terminals to agree on $r = 2$, the estimated average SNR should be within the range of 5 dB $\leq \overline{\text{SNR}}^{Bob}_{H_{opt}} < 12$ dB when the RIS is OFF, and 3 dB $\leq \overline{\text{SNR}}^{Bob}_{H_{opt}} < 12$ dB when the RIS is ON. When the estimated average SNR is below the specified range, both terminals can agree on $r = 1$ if $\overline{\text{SNR}}^{Bob}_{H_{opt}} < 5$ dB when the RIS is OFF, and $\overline{\text{SNR}}^{Bob}_{H_{opt}} < 3$ dB when the RIS is ON.



(a) SBGR vs. SNR at $r = 3$. (b) SBGR vs. SNR at $r = 2$. (c) SBGR vs. SNR at $r = 1$.

(d) BMR vs. SNR at $r = 3$. (e) BMR vs. SNR at $r = 2$. (f) BMR vs. SNR at $r = 1$.

Figure 6.7: The scheme's performance of the SBGR and the BMR at different SNRs and $r = \{1,2,3\}$.

Table 6.2: The optimised SNRs for $r = \{1,2,3\}$, with/without the RIS, and the BMR ≤ 0.1 bits/sample

Quantisation order	**RIS Status (ON/OFF)**	
	RIS-OFF	**RIS-ON**
$r = 3$	SNR $\geq$ 12 dB	SNR $\geq$ 12 dB
$r = 2$	5 dB $\leq$ SNR $<$ 12 dB	3 dB $\leq$ SNR $<$ 12 dB
$r = 1$	SNR $\leq$ 5 dB	SNR $\leq$ 3 dB

Furthermore, the extracted bit-streams are rigorously evaluated for statistical defects through the application of the well-established randomness test suite developed by the NIST [164]. The results of each test are presented in the form of a p-value for extracted keys with a length of 256 bits, as depicted in Table 6.3. These values are then compared to the predetermined significance level (0.01) to assess the degree of randomness of the extracted bit-streams. It can be observed that the extracted keys exhibit satisfactory randomness properties, as their chaotic characteristics are predominantly determined by the random large integer parameters n_i and m_i of chaotic mapping operation in (6.1), selected by the individual users.

Table 6.3: Statistical randomness analysis of the extracted keys

NIST Statistical Test Suite (256 bits)	**p-value**
Key Entropy	0.299629
Monobit Test	0.59766
Long Runs Test	0.485934
Block Frequency Test	0.486333
Maurer Universal Statistical Test	0.156093
Overlapping Template Matchings Test	0.486245
Discrete Fourier Transform (Spectral) Test	0.507344

6.3.3 Overhead analysis

This part presents a discussion on the execution time required for Algorithm 1 and the identification of the optimal configuration (H_{opt}) to achieve the research objective. The reflecting units of the developed RIS prototype are controlled through positive-intrinsic-negative (PIN) diodes, which switch between two-phase states. The individual control of each unit element allows for operation in the near field and channel estimation. The configuration is generated using a Hadamard codebook in MATLAB, which is transferred over WiFi using a transmission control protocol (TCP)/internet protocol (IP) link to a server program running on the Raspberry Pi-3 (Model B). The clock speed of the Raspberry was optimised at 7.8 MHz, with an operational power consumption of 12-15 $watts$ and a beam switching speed of 8 $msec$s. Based on the updating time, the overall running time for 4096 RIS configurations is calculated as $4096 \times 0.008 = 32.7$ $seconds$ which is acceptable as a prototype RIS with limited performance capabilities. However, this time can be significantly reduced by using a high-speed FPGA that operates at a clock speed of up to 500 MHz. Specifically, this would entail updating the control circuits of the PIN diodes to ensure compatibility with the FPGA's clock speed. This strategy holds the potential to significantly shorten the required running time.

Besides, the security robustness of the proposed secret key extraction scheme depends on the infeasibility of solving the Diffie-Hellman problem through the utilisation of the Chebyshev chaotic mapping operation presented in (6.1). This is facilitated by the straightforward multiplication and modular arithmetic operations involved in the calculation of $T'_{n_i}(\theta_i)$. Hence, the proposed method exhibits significantly reduced computational complexity in comparison to that of the computationally intensive elliptic curve-based Diffie-Hellman key exchanging protocol.

6.4 Summary

This chapter investigates the feasibility of employing the RIS to enhance the PHY-layer secret key extraction performance in the presence of DoS attacks. An optimisation algorithm is proposed that leverages the RIS to boost the signals transmitted by legitimate users while suppressing the interfering signals from malicious adversaries. Furthermore, the effectiveness of the proposed RIS-assisted key extraction method has been experimentally demonstrated using a 1-bit RIS and two USRPs. Experimental results show that this method enhances the performance of the key extraction, as quantified by two performance metrics, the SBGR and BMR. Specifically, An increase in the SBGR from 1.62 to 1.75 $bits/sample$ is observed when the RIS is turned on, and a decrease in the BMR from 0.38 to 0.25 $bits/sample$ is observed when the RIS is enabled at a poor SNR of 0 dB. These findings are particularly significant for future insights into secure and reliable intelligent transportation systems. Additionally, the statistical randomness of the extracted keys is evaluated using the NIST statistical test suite, confirming that the extracted keys are suitable for use as cryptographic keys. In summary, the presented results and analyses offer valuable perspectives on the practical implementation and optimisation of the RISs in enhancing the security and functionality of the PHY-layer secret key extraction for poor SNR and NLoS scenarios. The next chapter explores the use of smart contracts in the process of reconciling the mismatched bits resulting from the channel non-reciprocity components in the PHY-layer key extraction process.

Chapter 7

Smart Contract-based Secret Key Extraction

One of the challenges of the secret key extraction process is the significant communication cost incurred by the reconciliation stage [100]. Furthermore, a reconciliation approach such as the Cascade algorithm exposes 60% of the matched bits to reconcile only 10% of the mismatched bits, posing a security threat [100]. Other reconciliation approaches, such as low-density parity-check [113] and turbo [114] codes, suffer from high computation complexities [115]. Accordingly, this chapter addresses these limitations by designing a blockchain-based reconciliation technique that allows a trusted third party (TTP) to serve as a referee between the communicating vehicles by publishing a transaction containing the correction sequence (*CS*) of the mismatched bits using smart contract-based blockchain technology. The published transaction allows the vehicles to obtain the *CS* while the transaction address serves as temporary proof of trustworthiness for the entire session rather than transmitting a certificate every time, thereby saving communication costs and storage capacity.

In terms of key extraction, several theoretical approaches have been published, see Fig. 2.14. However, the complexities involved in their practical integration with symmetric key cryptography (*SKC*)-based applications are typically overlooked. This study presents a blockchain-based authentication scheme in which a PKI-based approach is used for handshaking between communicating vehicles and the exchanging of authenticated probing packets. After the quantisation of the channel estimates, the *CS* is published by the TTP in the blockchain to address the discrepancies and have a secret shared key to be used for subsequent transmissions. Accordingly, *SKC*, the advanced encryption standard (AES) algorithm is used for re-authentication, resulting in significant computation cost-savings compared to public key cryptography (*PKC*)-based methods.

The following summarises the contributions of this chapter which are published in [24], fulfilling the outlined thesis objectives (1, 2, 3, 5, 6) detailed in Subsection 1.4.3:

1. A blockchain-based secret key extraction (BCSKE) scheme is proposed for authentica-

tion in VANETs. The BCSKE scheme incorporates the channel phase response-based secret key extraction algorithm [99] for key agreement between communicating terminals. Accordingly, the proposed scheme uses *PKC* and *SKC*-based signatures at first and subsequent transmissions, respectively, mitigating the significant costs of using *PKC*-based signatures for each transmission.

2. In addition, this study demonstrates how the smart contract can be used to establish and publish the relationship between *CS* and vehicles' related information via a transaction, hence, leveraging the immutable and memorable properties of blockchain technology to map the transaction address to vehicles' related information.

3. The correctness and security robustness of the BCSKE scheme is demonstrated through BAN-logic analysis and the automated validation of internet security protocols and applications (AVISPA) simulation tool. The discussion also covered the scheme's resistance to various attacks.

4. Finally, the scheme's performance is analysed, and a comprehensive evaluation is conducted in terms of computation and communication costs, authentication delay, and packet loss ratio using the OMNeT++ network simulator.

The remainder of this chapter is organised as follows. Section 7.1 demonstrates the preliminary aspects of the BCSKE scheme. Section 7.2 details the steps involved in developing the proposed scheme. Sections 7.3 and 7.4 evaluate the scheme's security strength and performance, respectively. Finally, Section 7.5 presents the conclusions generated from this work.

7.1 Key extraction and system model

This section reviews the key extraction algorithm in [99]. Then, the system modelling is discussed in detail. Table 7.1 lists the notations used in this section.

7.1.1 Review of the channel phase response-based secret key extraction algorithm in [99]

The pairwise key extraction process between communicating vehicles, V_1 and V_2, consists of the following steps.

- *Step 1*: V_1 sends V_2 a probing packet PP_{V_1} at time T_1 in the following simplified form.

$$PP_{V_1}(T_1) = e^{j(w_c T_1 + \phi_1)} \tag{7.1}$$

Table 7.1: List of notations for the proposed blockchain-based authentication scheme

Symbol	Definition
Sk_{V_i}	The private key of the vehicle V_i
Pk_{V_i}	The public key of the vehicle V_i
T_R	The expiry date of the digital certificate
$Cert_{V_i}$	The digital certificate of the vehicle V_i
σ_i	The generated signature of the content x
T_i	The timestamp of the generated signature
T_r	The signature receiving time
T_Δ	The timestamp expiry period [00:00:59]
$T_{Session}$	The session expiry period [00:04:59]
$\hat{k}_{V_i}$	The extracted secret key by vehicle V_i
Sk_{RSU_j}	The private key of the RSU_j
Pk_{RSU_j}	The public key of the RSU_j
Sk_{RTA}	The private key of the region trust authority (RTA)
Pk_{RTA}	The public key of the RTA
Sk_{TA}	The private key of the TA
Pk_{TA}	The public key of the TA
$CS_{V_{1-2}}$	The correction sequence of $\hat{k}_{V_1}$ and $\hat{k}_{V_2}$
$TxID$	The transaction ID in the blockchain
T_{Tx}	The transaction publishing timestamp
$\|\|$	Concatenation between two variables

where ϕ_1 is a uniformly distributed random phase chosen by V_1 within the interval $[0, 2\pi)$. So that V_2's received signal can be formulated as

$$R_{V_{12}}(t) = \alpha_{12} e^{j(w_c t + \phi_1 + \theta_{12})} + \eta_{12}(t) \tag{7.2}$$

where $\eta_{12}(t)$ is the additive white Gaussian noise (AWGN) and α_{12} and θ_{12} are the forward link channel gain and phase responses, respectively. At last, V_2 obtains the noisy phase estimate $\hat{\phi}_{12} \approx \phi_1 + \theta_{12}$.

- *Step 2*: Similarly, V_2 sends V_1 a probing packet PP_{V_2} at time T_2 in the following simplified form.

$$PP_{V_2}(T_2) = e^{j(w_c T_2 + \phi_2)} \tag{7.3}$$

where ϕ_2 is a uniformly distributed phase chosen by V_2 within the interval $[0, 2\pi)$. So that V_1's received signal can be formulated as

$$R_{V_{21}}(t) = \alpha_{21} e^{j(w_c t + \phi_2 + \theta_{21})} + \eta_{21}(t) \tag{7.4}$$

where $\eta_{21}(t)$ is the AWGN and α_{21} and θ_{21} are the reverse link channel gain and phase

responses, respectively. At last, V_1 obtains the noisy phase estimate $\hat{\phi}_{21} \approx \phi_2 + \theta_{21}$.

- *Step 3*: Both vehicles compute the final phase components used for the key extraction as follows.

$$\begin{aligned} V_1 : \Phi_1 &= \hat{\phi}_{21} + \phi_1 \bmod 2\pi \\ V_2 : \Phi_2 &= \hat{\phi}_{12} + \phi_2 \bmod 2\pi \end{aligned} \tag{7.5}$$

  Note that, $\theta_{12} \approx \theta_{21}$ for $T_2 - T_1 \leq T_c$.

- *Step 4*: Finally, both vehicles map Φ_1 and Φ_2 into the quantisation region to get $\hat{k}_{V_1}$ and $\hat{k}_{V_2}$ by applying the following formula.

$$Q(x) = k \quad \text{if } x \in \left[\frac{2\pi(k-1)}{q}, \frac{2\pi k}{q}\right) \tag{7.6}$$

  for $k = 1, 2, \ldots, q$. See reference [99] for more details.

7.1.2 System modelling

In the proposed BCSKE scheme, six entities are involved: the TA, the RTAs, the RSUs, the vehicles' OBUs, and the smart contract using blockchain technology - see Fig. 7.1. The following defines the role of the network entities.

- *TA*: The TA initialises the scheme's public parameters and registers the network terminals in the system. It has the authority to reveal the real identities of the network terminals in case of malicious behaviours. Furthermore, it distributes the CRL of the misbehaving vehicles between terminals.

- *RTA*: In each region, there is an RTA that provides vehicles with efficient authentication services and reduces the TA's computational overhead. The RTA's regional centralised servers are responsible for reconciling the mismatched bits of the extracted keys between vehicles by triggering the smart contract and publishing the computed correction sequences in the blockchain through transactions in an orderly fashion.

- *RSU*: In both directions of the road, the RSUs are deployed with high storage and computation capacities. It functions as a cooperative relay between vehicles and the RTA, allowing wireless and wired communication between itself and surrounding vehicles as well as between itself and the RTA, respectively.

- *OBUs*: It is a vehicle-mounted processing unit with constrained computation capabilities and a tamper-proof property. It also has the availability to trigger the smart contract's **Get** function and retrieve the transaction information from the blockchain.

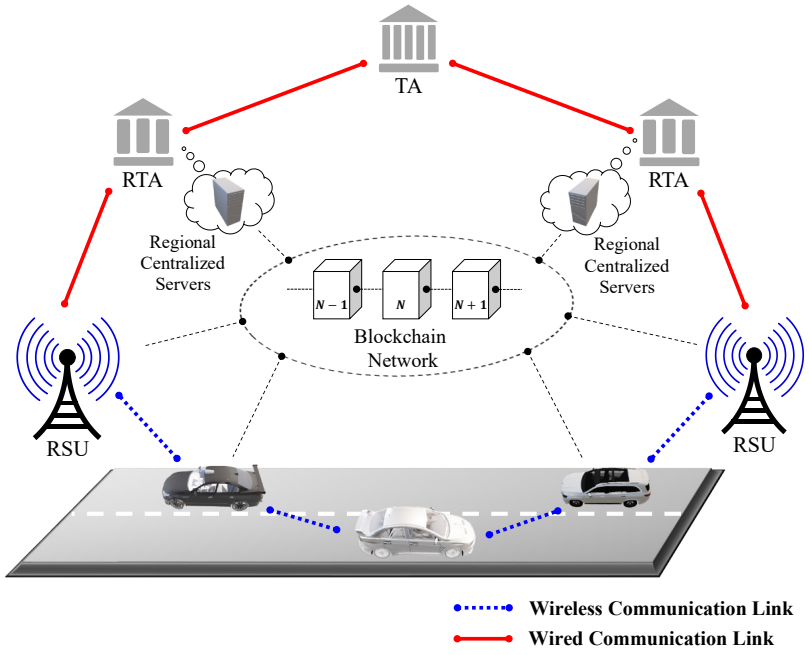


Figure 7.1: VANET architecture using blockchain technology.

- *Blockchain network*: The blockchain network is a decentralised distributed database system that offers immutable, undeniable, and verifiable data storage through transactions [140]. Recently, researchers have been contributing to the development of blockchain technology with horizontal and vertical scaling expansion, such as Ethereum and Hyperledger, respectively [139]. Horizontal expansion in blockchain refers to scaling solutions aimed at widening participation without compromising security. For instance, Ethereum's approach enables a broader user base by allowing anyone to participate in its network, fostering a diverse ecosystem of applications and users [139]. In contrast, vertical expansion, as seen in Hyperledger, emphasises restricted access controls, catering to enterprises and consortiums that prioritise privacy and permission access over a larger, open network. These scaling strategies play a pivotal role in shaping the blockchain landscape, accommodating different trust models and use cases while ensuring the integrity and functionality of the underlying technology. Both types maintain an immutable and chronological sequence of chaining using proof-of-work (PoW) and proof-of-stack (PoS) consensus mechanisms. This approach entails embedding the correction sequence of the shared key into the transaction such that vehicles can reconcile the mismatched bits of the shared key from the transaction content. RTAs' activities in VANET are transparent and verifiable, so the transaction contents in this study function as a short-term digital certificate for a specific period called the session time $T_{Session}$.

- *Smart contract (SC) using blockchain technology*: *SC*s are self-executing transactions-based contracts whose terms and conditions are written in the form of codes using the

Algorithm 1: Smart Contract for **KeyAgreement**

Given: function name, parameter settings

Require: Setting up functions

```
struct V2V {uint pk1; uint pk2; uint CS;
} //Define the input parameter types
address RTA = 0xcbB3012b86b594223E43FB9c56624F357463b;
mapping (uint → uint256) public PK2TX;
function Deployer ( ) public {RTA = msg.sender;
} //Define the deployer as the RTA
modifier onlyowner {require (msg.sender == RTA);
_;
}//Identify the message sender
V2V keyagreementl;
function IssueCS (uint _pk1, uint _pk2, uint _CS)
onlyowner public returns (uint, uint, uint) {
keyagreement1.pk1 = _pk1;
keyagreement1.pk2 = _pk2;
keyagreementl.CS = _CS;
return (keyagreement1.pk1, keygreement1.pk2, keyagreement1.CS)
} //Generate a transaction for CS and get TxID
function Update (uint pk1, uint pk2, uint256 TxID)
onlyowner public {
PK2TX [pk1 ∧ pk2] = TxID;
} //Mapping the inserted pair of public keys to TxID
function Get (uint pk1, uint pk2) public view returns
(uint256) {
return PK2TX [pk1 ∧ pk2];
} //Retrieve the TxID by vehicles
```

Turing complete scripting language (i.e., *Solidity* for Ethereum smart contracts). These codes are distributed throughout the decentralised blockchain network. Based on the conditions assigned, the *SC*'s built-in functions can be triggered. This chapter uses the public Ethereum blockchain as the platform for the creation of the *SC*. The main reasons behind the use of the *SC* in Algorithm (1) are to (I) publish the correction sequence of the extracted key's mismatched bits between the communicating vehicles, V_1 and V_2, via a transaction Tx, retrieving its address $TxID$, and (II) use the retrieved $TxID$ as a proof of trustworthiness for subsequent transmissions. There are four functions to be provided in the involved *SC*. The **Deployer** function is used to specify the address of the RTA (owner) that is authorised to deploy the *SC* in the blockchain. The **IssueCS**(Pk_{V_1}, Pk_{V_2}, CS) function can only be invoked by the RTA (only owner), which is used to publish the correction sequence along with the communicating vehicles' public keys (Pk_{V_1}, Pk_{V_2}) in the blockchain and get $TxID$. The **Update**$(Pk_{V_1}, Pk_{V_2}, TxID)$ function is used to map $TxID$ to (Pk_{V_1}, Pk_{V_2}), and also can only be invoked by the RTA. The **Get**(Pk_{V_1}, Pk_{V_2}) function is a view function that can be invoked by network terminals to retrieve $TxID$ without incurring any gas fees.

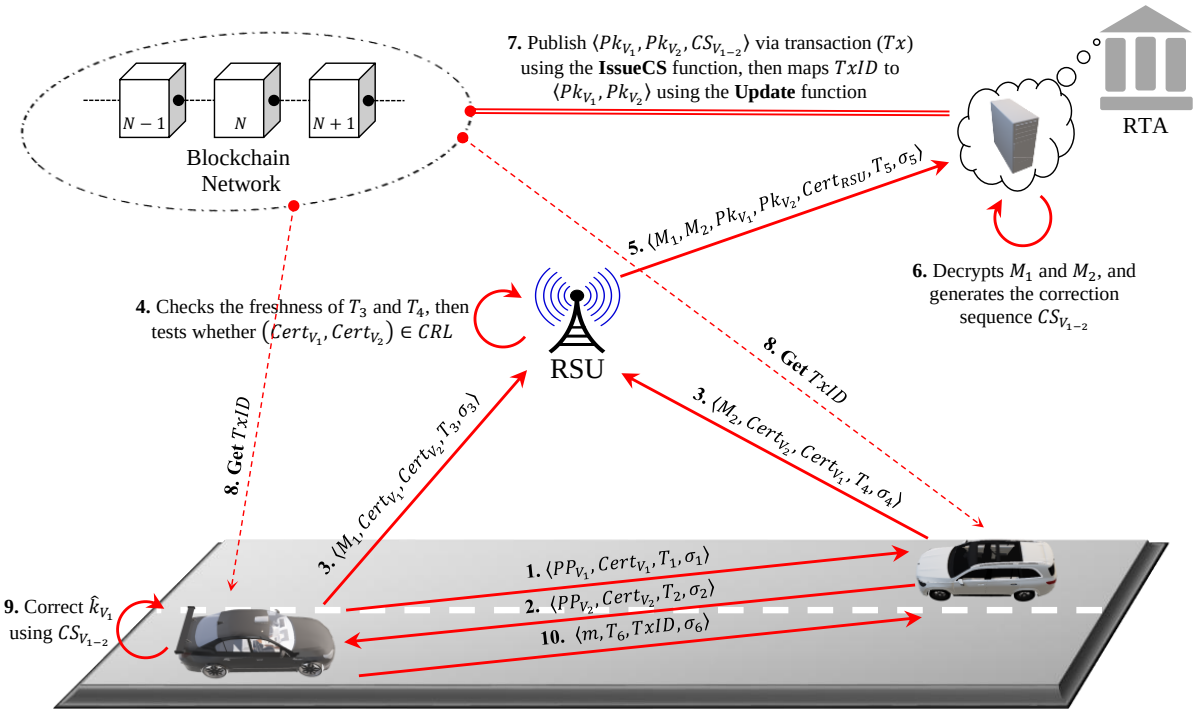


Figure 7.2: The proposed blockchain-based authentication model for VANETs.

7.2 The proposed scheme

This section describes the BCSKE scheme implemented on the public blockchain (e.g., Ethereum), see Fig. 7.2. In this scheme, each network terminal (i.e., vehicles and RSUs) possesses a long-term digital public key certificate that is used for initial legitimacy detection using PKI-based authentication. Taking advantage of the short-term reciprocal properties of the channel phase responses and employing its unpredictable behaviour as a source of randomness, both vehicles, $V_1(Alice)$ and $V_2(Bob)$, can exchange authenticated and time-stamped probing packets (PP) within the coherence interval T_c to extract high entropy secret keys (Subsection 7.2.3: Step 1∼2). Due to the use of the half-duplex mode when probing the channel, the extracted bit sequences, $\hat{k}_{V_1}$ and $\hat{k}_{V_2}$, have some discrepancies. The BMR is used to define the number of mismatched bits to the total number of channel samples, formulated as

$$BMR = \frac{\bar{I}(\hat{k}_{V_1}, \hat{k}_{V_2})}{No.\ Channel\ Samples} \tag{7.7}$$

where $\bar{I}(\hat{k}_{V_1}, \hat{k}_{V_2})$ is the number of mismatched/incorrect bits between $\hat{k}_{V_1}$ and $\hat{k}_{V_2}$. While the BGR is defined as the order/length of the extracted bit sequence to the total number of channel samples, denoted by

$$BGR = \frac{Bit\ Length\ (\hat{k}_{V_{1(2)}})}{No.\ Channel\ Samples} \tag{7.8}$$

In order to avoid the communication cost and security flaws associated with the information reconciliation stage, both vehicles encrypt $\hat{k}_{V_1}$ and $\hat{k}_{V_2}$ and send them to the RTA within the same region (Subsection 7.2.3: Step 3∼6). In the proposed scheme, the RTA acts as a referee (TTP) between the communicating vehicles, correcting the mismatched bits and generating the correction sequence. After the RTA deploys the *SC*, it establishes the relationship between the pair of public keys of the communicating vehicles and its associated correction sequence (Subsection 7.2.3: Step 7). Finally, both vehicles can obtain an identical shared key (Subsection 7.2.4) used for symmetric key cryptography at subsequent transmissions (Subsection 7.2.5: Step 1, 2). In general, the BCSKE scheme consists of five phases, i.e., system initialisation, registration, initial verification and channel probing, key reconciliation, and message signing and verification.

7.2.1 System initialisation phase

Following are the processes by which the TA generates the public parameters of the system.

- The system is set up with an elliptic curve $E : y^2 = x^3 + ax + b \bmod p$, where $a, b \in Z_q^*$ with a condition $\Delta = 4a^3 + 27b^2 \neq 0$ and p is a large prime number. For 80-bit security, the recommended domain parameters of the 160-bit elliptic curve "secp160k1" from [141] are used, see Table 7.2.

- Using the base point g, the TA creates the cyclic additive group $\mathbb{G}$ of order q comprising all the points on E and the point of infinity $\mathcal{O}$.

- The TA selects its own private key $Sk_{TA} \in Z_q^*$ and computes its associated public key $Pk_{TA} = Sk_{TA}.g$.

- The TA selects a unique private key for all the RTAs $Sk_{RTA} \in Z_q^*$ and computes its associated public key $Pk_{RTA} = Sk_{RTA}.g$.

- The hash function $H_1 : \{0,1\}^* \rightarrow \{0,1\}^{N_1}$.

- The TA deploys the *SC* on behalf of the RTA (i.e., using the RTA's address), then obtains the *SC*'s unique identity/address *SCID*. The network terminals use the obtained *SCID* to call the **Get** function from the deployed *SC* to attain their *CS*.

- Finally, the public parameters of the scheme are $PPs = \langle a, b, p, q, g, Pk_{TA}, H_1, SCID \rangle$.

7.2.2 Registration phase

The TA is responsible for registering all the terminals before being part of the network by doing the steps below.

- For registering a vehicle V_i, the TA checks V_i's real identity RID_{V_i}, picks up at random V_i's private key $Sk_{V_i} \in Z_q^*$, then computes its associated public key $Pk_{V_i} = Sk_{V_i}.g$. At last, the TA creates $Cert_{V_i} = \langle Pk_{V_i}, T_R, \sigma_{TA} \rangle$ in which $\sigma_{TA} = \text{sign}_{Sk_{TA}}(Pk_{V_i} \| T_R)$ and T_R is the expiry date of the certificate. This is also done for all registered RTAs and RSUs in the network.

- As a final step, the TA preloads $\langle PPs, Pk_{RTA}, Sk_{V_i}, Cert_{V_i} \rangle$ onto the registered V_i, $\langle PPs, Pk_{RTA}, Sk_{RSU_j}, Cert_{RSU_j} \rangle$ onto the registered RSU_j, and $\langle PPs, Sk_{RTA}, Cert_{RTA} \rangle$ onto the registered RTA. Note that, only the TA has the link between the RID_{V_i} and $Cert_{V_i}$ to reveal V_i's real identity in case of malicious activity.

Fig. 7.3 shows the top-level description flowchart of the proposed scheme's phases following the registration phase.

7.2.3 Initial verification and channel probing phase

In this phase, both terminals exchange authenticated probing packets (PP_{V_i}) along with digital certificates used for establishing a shared key and mutual authentication. This phase comprises the following steps:

- *Step 1*: During the first transmission slot, V_1 sends V_2 a communication request in the form of $\langle PP_{V_1}, Cert_{V_1}, T_1, \sigma_1 \rangle$, where $\sigma_1 = \text{sign}_{Sk_{V_1}}(PP_{V_1} \| Cert_{V_1} \| T_1)$ generated at timestamp T_1.

- *Step 2.1*: V_2 replies by sending the tuple $\langle PP_{V_2}, Cert_{V_2}, T_2, \sigma_2 \rangle$ to V_1, for $T_2 - T_1 \leq T_c$, where $\sigma_2 = \text{Sign}_{Sk_{V_2}}(PP_{V_2} \| Cert_{V_2} \| T_2)$ generated at timestamp T_2. After that, both vehicles check if $(Cert_{V_1}, Cert_{V_2}) \in$ CRL. If not, they check the freshness of the received timestamp by finding out if $T_r - T_i \leq T_\Delta$ holds or not, defending against replay attacks. Then, both verify the received signatures as $\text{verf}_{Pk_{V_{1(2)}}}(\sigma_{1(2)})$.

- *Step 2.2*: Based on the short-term channel reciprocity, both vehicles obtain their channel phase response estimates, specified in (7.5), and quantise them using (7.6) to get the bit

Table 7.2: The recommended domain parameters of the 160-bit elliptic curve "secp160k1" in the Hexadecimal form [141]

Par.	**Recommended value**
a	00000000 00000000 00000000 00000000 00000000
b	00000000 00000000 00000000 00000000 00000007
p	*FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFE FFFFAC*73
g	04 3*B*4*C*382*C E*37*AA*192 *A*4019*E*76 3036*F*4*F*5 *DD*4*D*7*EBB* 938*CF*935 318*FDCED* 6*BC*28286 531733*C*3 *F*03*C*4*FEE*
q	01 00000000 00000000 0001*B*8*FA* 16*DFAB*9*A CA*16*B*6*B*3

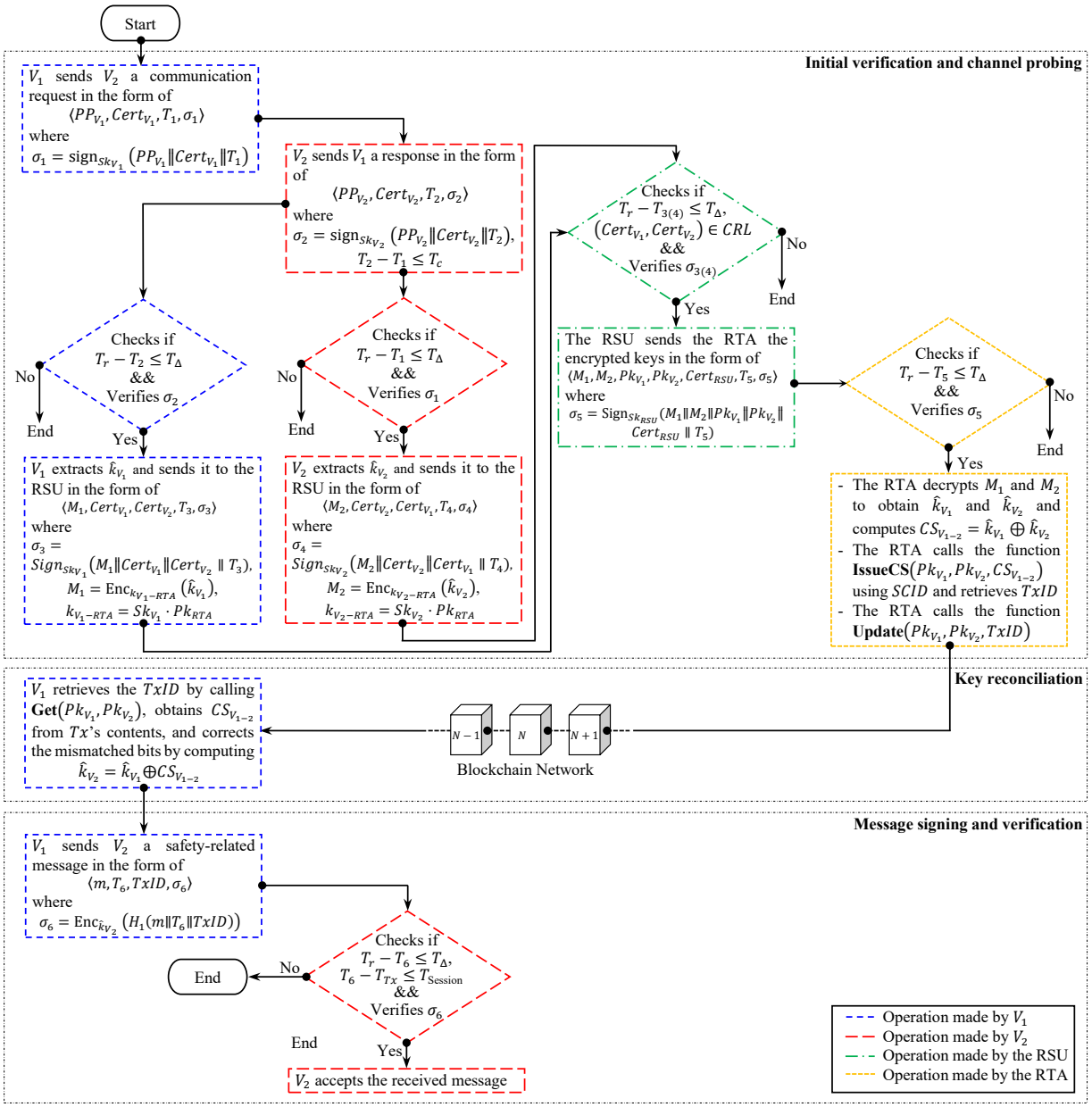


Figure 7.3: The top-level description flowchart of the proposed scheme.

streams $\hat{k}_{V_1}$ and $\hat{k}_{V_2}$ at the side of V_1 and V_2, respectively. However, $\hat{k}_{V_1}$ and $\hat{k}_{V_2}$ hold some mismatched bits resulting from the channel non-reciprocity components.

- *Step 3*: Accordingly, both vehicles encrypt their secret keys to get $M_{1(2)} = \text{Enc}_{k_{V_{1(2)}-RTA}}(\hat{k}_{V_{1(2)}})$ in which $k_{V_{1(2)}-RTA} = Sk_{V_{1(2)}}.Pk_{RTA}$ and send it to the RSU in the form of

$$\begin{aligned} V_1 &\rightarrow RSU : \langle M_1, Cert_{V_1}, Cert_{V_2}, T_3, \sigma_3 \rangle, \\ V_2 &\rightarrow RSU : \langle M_2, Cert_{V_2}, Cert_{V_1}, T_4, \sigma_4 \rangle \end{aligned} \tag{7.9}$$

where $\sigma_{3(4)} = \text{Sign}_{Sk_{V_{1(2)}}}(M_{1(2)} \| Cert_{V_{1(2)}} \| Cert_{V_{2(1)}} \| T_{3(4)})$ generated at $T_{3(4)}$ timestamp.

- *Step 4*: The RSU, in turn, checks the freshness of the received timestamp $T_{3(4)}$, checks if $(Cert_{V_1}, Cert_{V_2}) \in$ CRL, and then verifies the received signatures $\text{verf}_{Pk_{V_{1(2)}}}(\sigma_{3(4)})$.

- *Step 5*: The RSU forward the encrypted secret keys to the RTA as

$$\langle M_1, M_2, Pk_{V_1}, Pk_{V_2}, Cert_{RSU}, T_5, \sigma_5 \rangle \tag{7.10}$$

  where $\sigma_5 = \text{Sign}_{Sk_{RSU}}(M_1 \| M_2 \| Pk_{V_1} \| Pk_{V_2} \| Cert_{RSU} \| T_5)$ generated at T_5 timestamp.

- *Step 6*: The RTA checks T_5, verifies the received signature $\text{verf}_{Pk_{RSU}}(\sigma_5)$, then decrypts M_1 and M_2 to get $\hat{k}_{V_1}$ and $\hat{k}_{V_2}$, respectively, as $\hat{k}_{V_{1(2)}} = \text{Dec}_{k_{V_{1(2)}-RTA}}(M_{1(2)})$ in which $k_{V_{1(2)}-RTA} = Sk_{RTA}.Pk_{V_{1(2)}}$.

- *Step 7*: Accordingly, if the RTA finds a sufficient matching percentage between $\hat{k}_{V_1}$ and $\hat{k}_{V_2}$, it computes the correction sequence $CS_{V_{1-2}} = \hat{k}_{V_1} \bigoplus \hat{k}_{V_2}$, and then records $CS_{V_{1-2}}$ into the blockchain by calling **IssueCS**$(Pk_{V_1}, Pk_{V_2}, CS_{V_{1-2}})$ using *SCID*. Once the miners chain the transaction into the blockchain at time T_{Tx} and the RTA obtains the transaction identity $TxID$, it maps the pair of public keys (Pk_{V_1}, Pk_{V_2}) to $TxID$ by calling **Update**$(Pk_{V_1}, Pk_{V_2}, TxID)$ in the *SC* using the *SCID*.

7.2.4 Key reconciliation

In this phase, V_1 reconciles the mismatched bits in $\hat{k}_{V_1}$ by performing the following steps.

- V_1 obtains $TxID$ by calling **Get**(Pk_{V_1}, Pk_{V_2}) using *SCID* to get $CS_{V_{1-2}}$ from the blockchain and agrees on a shard key with V_2 by computing $\hat{k}_{V_2} = \hat{k}_{V_1} \bigoplus CS_{V_{1-2}}$.

7.2.5 Message signing and verification phase

In this phase, V_1 signs a time-stamped safety-related message m using symmetric key cryptography and sends it to V_2. Using the key obtained, V_2 verifies the received signature and accepts the received message.

- *Step 1*: V_1 sends V_2 the tuple $\langle m, T_6, TxID, \sigma_6 \rangle$, where $\sigma_6 = \text{Enc}_{\hat{k}_{V_2}}(H_1(m \| T_6 \| TxID))$ generated at T_6 timestamp.

- *Step 2*: V_2 checks the freshness of T_6 timestamp, invokes the $TxID$ data from the blockchain to verify the session's continuity by checking if $T_6 - T_{Tx} \leq T_{Session}$ holds or not, then decrypts σ_6 to verify the integrity of the attached data by testing whether $\text{Dec}_{\hat{k}_{V_2}}(\sigma_6) \stackrel{?}{=} H_1(m \| T_6 \| TxID)$. If true, the message is accepted. Otherwise, it will be discarded.

Table 7.3: BAN-Logic symbols and their equivalent scheme notations

BAN-Logic variable	Scheme notation
$k_{V_i}^{-1}$	Sk_{V_i}
k_{V_i}	Pk_{V_i}
T_R	T_R
$\{k_{V_i}, T_R\}_{k_{TA}^{-1}}$	$Cert_{V_i}$
$\{x\}_{k^{-1}}$	σ_i
T_i	T_i
$\hat{k}_{V_i}$	$\hat{k}_{V_i}$
$k_{RSU_j}^{-1}$	Sk_{RSU_j}
k_{RSU_j}	Pk_{RSU_j}
k_{RTA}^{-1}	Sk_{RTA}
k_{RTA}	Pk_{RTA}
k_{TA}^{-1}	Sk_{TA}
k_{TA}	Pk_{TA}
,	$\|\|$

7.3 Security proofs and analysis

This section proves the correctness of the BCSKE scheme using BAN-logic, analyses its security strength and proves its robustness using the AVISPA simulation tool.

7.3.1 BAN-logic security proof

The BAN-logic is a proof of correctness technique used to verify the validity of the authentication scheme [142]. The proposed BCSKE scheme is analysed using BAN-logic analysis, demonstrating successful key agreement and authentication processes. Table 7.3 shows the BCSKE scheme used notations and their corresponding BAN-logic symbols.

1. *Notations*: The following notations are used for the BAN-logic security proof:

Table 7.4: The rules involved in the BAN-logic analysis

No.	Rule	BAN-logic representation	Definition
R_1	MMR for a shared key	$\frac{A\|\equiv(A\overset{K}{\longleftrightarrow}B), A\triangleleft\{X\}_K}{A\|\equiv(B\|\sim X)}$	If A believes in K and A received X encrypted by K, then A believes B said X
R_2	MMR for a public key	$\frac{A\|\equiv(B\overset{K}{\longrightarrow}A), A\triangleleft\{X\}_{k^{-1}}}{A\|\equiv(B\|\sim X)}$	If A believes K is B's public key and receives X encrypted with B's private key, then A believes B said X
R_3	nonce verification rule (NVR)	$\frac{A\|\equiv\#(X), A\|\equiv(B\|\sim X)}{A\|\equiv(B\|\equiv X)}$	If A believes X is fresh and that B said X, then A believes B believes X
R_4	jurisdiction rule (JR)	$\frac{A\|\equiv(B\Longrightarrow X), A\|\equiv(B\|\equiv X)}{A\|\equiv X}$	If A believes B has jurisdiction over X and that B believes X, then A believes X
R_5	freshness rule (FR)	$\frac{A\|\equiv\#(X)}{A\|\equiv\#(X,Y)}$	Freshness of one part ensures the freshness of the entire formula

(a) $A \mid\equiv X$: A believes X and accepts it as true.

(b) $A \triangleleft X$: A sees X, indicating A received a message containing X.

(c) $A \mid\sim X$: X has once transmitted and believed by A at one time.

(d) $A \mid\Longrightarrow X$: A controls X and has jurisdiction over it.

(e) $A \xleftrightarrow{k} B$: A and B use k as a shared key for communication.

(f) $A \xrightarrow{k} B$: k represents the public key of A.

(g) $\{X\}_k$: X is encrypted using the shared key k.

(h) $\#(X)$: X is a fresh message.

2. *Rules*: A set of beliefs can be generated by manipulating the protocol according to the rules listed in Table 7.4.

3. *Goals*: In BAN-logic, the aim is to prove the correctness of the proposed scheme by satisfying the following goals.

(a) *Goal 1*: $V_1 \mid\equiv (V_1 \xleftrightarrow{\hat{k}_{V_2}} V_2)$.

(b) *Goal 2*: $V_2 \mid\equiv (V_1 \xleftrightarrow{\hat{k}_{V_2}} V_2)$.

(c) *Goal 3*: $V_2 \mid\equiv (V_1 \mid\sim m)$.

4. *Idealised forms*: Following are the outlines of the idealised messaging forms for the proposed protocol.

(a) $Msg_1 : V_1 \rightarrow V_2 : \{PP_{V_1}, Cert_{V_1}, T_1\}_{k_{V_1}^{-1}}$, where $Cert_{V_1} = \{k_{V_1}, T_R\}_{k_{TA}^{-1}}$.

(b) $Msg_2 : V_2 \rightarrow V_1 : \{PP_{V_2}, Cert_{V_2}, T_2\}_{k_{V_2}^{-1}}$, where $Cert_{V_2} = \{k_{V_2}, T_R\}_{k_{TA}^{-1}}$.

(c) $Msg_3 : V_1 \rightarrow$ RSU: $\{M_1, Cert_{V_1}, Cert_{V_2}, T_3\}_{k_{V_1}^{-1}}$, where $M_1 = \{\hat{k}_{V_1}\}_{k_{V_1-RTA}}$ and $k_{V_1-RTA} = k_{V_1}^{-1}.\, k_{RTA} = k_{RTA}^{-1}.k_{V_1}$ using Diffie-Hellman protocol.

(d) $Msg_4 : V_2 \rightarrow$ RSU: $\{M_2, Cert_{V_2}, Cert_{V_1}, T_4\}_{k_{V_2}^{-1}}$, where $M_2 = \{\hat{k}_{V_2}\}_{k_{V_2-RTA}}$ and $k_{V_2-RTA} = k_{V_2}^{-1}.\, k_{RTA} = k_{RTA}^{-1}.k_{V_2}$ using Diffie-Hellman protocol.

(e) $Msg_5 : RSU \rightarrow$ RTA: $\{M_1, M_2, k_{V_1}, k_{V_2}, T_5, Cert_{RSU}\}_{k_{RSU}^{-1}}$, where $Cert_{RSU} = \{k_{RSU}, T_R\}_{k_{TA}^{-1}}$.

(f) $Msg_6 : V_1 \rightarrow V_2 : \{m, T_6, TxID\}_{\hat{k}_{V_2}}$.

5. *Assumptions*: Following are the basic assumptions that underlie the BAN logic security proof.

(a) A_1: $V_2 \mid\equiv \#(T_1)$.

(b) A_2: $V_1 \mid\equiv \#(T_2)$.

(c) A_3: $RSU \mid\equiv \#(T_3)$.

(d) A_4: $RSU \mid\equiv \#(T_4)$.

(e) A_5: $RTA \mid\equiv \#(T_5)$.

(f) A_6: $V_2 \mid\equiv \#(T_6)$.

(g) A_7: $RTA \mid\equiv RTA \overset{k_{V_1-RTA}}{\longleftrightarrow} V_1$.

(h) A_8: $RTA \mid\equiv RTA \overset{k_{V_2-RTA}}{\longleftrightarrow} V_2$.

(i) A_9: $V_2 \mid\equiv (TA \overset{K_{TA}}{\longrightarrow} V_2)$.

(j) A_{10}: $\dfrac{V_2 \mid\equiv (TA \overset{K_{TA}}{\longrightarrow} V_2), V_2 \triangleleft \{k_{V_1}, T_R\}_{k_{TA}^{-1}}}{V_2 \mid\equiv (V_1 \overset{k_{V_1}}{\longrightarrow} V_2)}$.

(k) A_{11}: $V_1 \mid\equiv (TA \overset{K_{TA}}{\longrightarrow} V_1)$.

(l) A_{12}: $\dfrac{V_1 \mid\equiv (TA \overset{K_{TA}}{\longrightarrow} V_1), V_1 \triangleleft \{k_{V_2}, T_R\}_{k_{TA}^{-1}}}{V_1 \mid\equiv (V_2 \overset{k_{V_2}}{\longrightarrow} V_1)}$.

(m) A_{13}: $RSU \mid\equiv (TA \overset{K_{TA}}{\longrightarrow} RSU)$.

(n) A_{14}: $\dfrac{RSU \mid\equiv (TA \overset{K_{TA}}{\longrightarrow} RSU), RSU \triangleleft \{k_{V_1}, T_R\}_{k_{TA}^{-1}}}{RSU \mid\equiv (V_1 \overset{k_{V_1}}{\longrightarrow} RSU)}$.

(o) A_{15}: $\dfrac{RSU \mid\equiv (TA \overset{K_{TA}}{\longrightarrow} RSU), RSU \triangleleft \{k_{V_2}, T_R\}_{k_{TA}^{-1}}}{RSU \mid\equiv (V_2 \overset{k_{V_2}}{\longrightarrow} RSU)}$.

(p) A_{16}: $RTA \mid\equiv (TA \overset{K_{TA}}{\longrightarrow} RTA)$.

(q) A_{17}: $\dfrac{RTA \mid\equiv (TA \overset{K_{TA}}{\longrightarrow} RTA), RTA \triangleleft \{k_{RSU}, T_R\}_{k_{TA}^{-1}}}{RTA \mid\equiv (RSU \overset{k_{RSU}}{\longrightarrow} RTA)}$.

6. *Implementation*: Following is the BAN-logic security proof to the proposed protocol.

   - *Step 1*: V_2 receives Msg_1 from V_1.
   - *Step 2*: Applying A_9 and $Cert_{V_1}$ from Msg_1 into A_{10}, then the result is $R_1 : V_2 \mid\equiv (V_1 \overset{k_{V_1}}{\longrightarrow} V_2)$. Substituting R_1 and Msg_1 into the message meaning rule (MMR) in the public key form, then $R_2 : V_2 \mid \equiv (V_1 \mid\sim Msg_1)$. Applying A_1 and Msg_1 into the FR, then $R_3 : V_2 \mid\equiv \#(Msg_1)$. By combining R_2 and R_3 into the NVR, then $R_4 : V_2 \mid\equiv (V_1 \mid\equiv Msg_1)$.
   - *Step 3*: V_1 receives Msg_2 from V_2.
   - *Step 4*: Applying A_{11} and $Cert_{V_2}$ from Msg_2 into A_{12}, then $R_5 : V_1 \mid\equiv (V_2 \overset{k_{V_2}}{\longrightarrow} V_1)$. Substituting R_5 and Msg_2 into the MMR in the public key form, then $R_6 : V_1 \mid \equiv (V_2 \mid\sim Msg_2)$. Applying A_2 and Msg_2 into the FR, then $R_6 : V_1 \mid\equiv \#(Msg_2)$. By combining R_6 and R_7 into the NVR, then $R_8 : V_1 \mid\equiv (V_2 \mid\equiv Msg_2)$.

- *Step 5*: RSU receives Msg_3 from V_1.
- *Step 6*: Applying A_{13} and $Cert_{V_1}$ from Msg_3 into A_{14}, then $R_9 : RSU \mid\equiv (V_1 \xrightarrow{k_{V_1}} RSU)$. Substituting R_9 and Msg_3 into the MMR in the public key form, then $R_{10} : RSU \mid \equiv (V_1 \mid\sim Msg_3)$. Applying A_3 and Msg_3 into the FR, then $R_{11} : RSU \mid\equiv \#(Msg_3)$. By combining R_{10} and R_{11} into the NVR, then $R_{12} : RSU \mid\equiv (V_1 \mid\equiv Msg_3)$.
- *Step 7*: RSU receives Msg_4 from V_2.
- *Step 8*: Applying A_{13} and $Cert_{V_2}$ from Msg_4 into A_{15}, then $R_{13} : RSU \mid\equiv (V_2 \xrightarrow{k_{V_2}} RSU)$. Substituting R_{13} and Msg_4 into the MMR in the public key form, then $R_{14} : RSU \mid \equiv (V_2 \mid\sim Msg_4)$. Applying A_4 and Msg_4 into the FR, then $R_{15} : RSU \mid\equiv \#(Msg_4)$. By combining R_{14} and R_{15} into the NVR, then $R_{16} : RSU \mid\equiv (V_2 \mid\equiv Msg_4)$.
- *Step 9*: RTA receives Msg_5 from RSU.
- *Step 10*: Applying A_{16} and $Cert_{RSU}$ from Msg_5 into A_{17}, then $R_{17} : RTA \mid\equiv (RSU \xrightarrow{k_{RSU}} RTA)$. Substituting R_{17} and Msg_5 into the MMR in the public key form, then $R_{18} : RTA \mid \equiv (RSU \mid\sim Msg_5)$. Applying A_5 and Msg_5 into the FR, then $R_{19} : RTA \mid\equiv \#(Msg_5)$. By combining R_{18} and R_{19} into the NVR, then $R_{20} : RTA \mid\equiv (RSU \mid\equiv Msg_5)$.
- *Step 11*: Applying A_7 and M_1 from Msg_5 into the MMR in the shared key form, then $R_{21} : RTA \mid\equiv (V_1 \mid\sim M_1)$. Similarly, by applying A_8 and M_2 from Msg_5 into MMR in a symmetric shared key form, then $R_{22} : RTA \mid\equiv (V_2 \mid\sim M_2)$. Based on R_{21} and R_{22}, the RTA can calculate $CS_{V_{1-2}} = \hat{k}_{V_1} \bigoplus \hat{k}_{V_2}$ and record it into the blockchain. Accordingly, V_1 retrieves the $CS_{V_{1-2}}$ from the blockchain using $TxID$ and agrees with V_2 on $\hat{k}_{V_2}$. Now, $V_1 \mid\equiv (V_1 \overset{\hat{k}_{V_2}}{\longleftrightarrow} V_2)$ (***Goal 1***).
- *Step 12*: V_2 receives Msg_6 from V_1.
- *Step 13*: Once $V_2 \triangleleft TxID$ in the blockchain, then $V_2 \mid\equiv (V_2 \overset{\hat{k}_{V_2}}{\longleftrightarrow} V_1)$ (***Goal 2***). Thus, by applying ***Goal 2*** and Msg_6 into the MMR in the shared key form, then $V_2 \mid\equiv (V_1 \mid\sim Msg_6)$ (***Goal 3***).

7.3.2 Security analysis

Throughout this subsection, the security requirements fulfilled through this methodology are discussed, which are primarily determined by the digital signatures and blockchain system adopted.

1. *Message authentication*: The proposed scheme's security strength mainly depends on the infeasibility of solving the ECDLP for the first transmission slot and the sufficient security level provided by the symmetric key-based cryptography with a key length of order $|\hat{k}_{V_2}| \simeq 128$, 192, or 256 bits during subsequent transmission slots. Furthermore, the certificate $Cert_{V_i}$ signed by the TA allows the recipient to authenticate the sender's public key

Pk_{V_i}. Hence, the recipient is able to authenticate the received message by verifying the received signatures $\text{verf}_{Pk_{V_i}}(\sigma_i)$ and $\text{Dec}_{\hat{k}_{V_i}}(\sigma_i) \stackrel{?}{=} H_1(m\|T_i\|TxID)$ for first and subsequent transmission slots, respectively.

2. *Conditional privacy preservation*: Since all the transmitted messages have no information about V_i's real identity RID_{V_i}, no network terminal is able to expose RID_{V_i} except for TA, as it's the only terminal that stores the link between RID_{V_i} and V_i's issued long-term digital certificate $Cert_{V_i}$. Hence, preserving privacy under certain conditions.

3. *Unlinkability*: Since the key extraction process depends on the reciprocal features and the spatially and temporally correlated wireless channel responses within T_c, the adversary cannot establish a rational relationship between the extracted keys from different sessions, supporting forward and backward secrecy. In addition, the dynamically updated $TxID$ between vehicles prevents adversaries from linking messages from different sessions.

4. *Resistance to birthday collisions*: Ethereum's consensus mechanism depends on the proof of work (Ethereum 1.0) and proof of stake (Ethereum 2.0). This consensus mechanism helps prevent forking; thus, the likelihood of a block's birthday colliding is effectively reduced.

5. *Resistance to Hijacking*: All Ethereum transactions are signed using the digital signatures of the elliptic curve (secp256k1). The ECDSA's security ensures that no probabilistic polynomial time adversary can alter the signature of a transaction message, resisting this type of attack.

6. *Resistance to active attacks*: In order for the proposed scheme to be effective, it must be immune to the following types of active attacks.

   (a) *Resistance to modification*: The design of the BCSKE scheme ensures message integrity since V_j can detect modification attempts in the received message m from V_i by checking if $\text{Dec}_{\hat{k}_{V_i}}(\sigma_i) \stackrel{?}{=} H_1(m\|T_i\|TxID)$, where $\hat{k}_{V_i}$ is the extracted shared key based on the unpredictable channel randomness between V_i and V_j.

   (b) *Resistance to replay*: The freshly extracted shared key $\hat{k}_{V_i}$ between V_i and V_j in each session allows for avoiding replaying attacks from different sessions. In addition, the attached timestamp T_i helps the recipient to check the freshness of the received message during the same session interval. Hence, resisting such attacks.

   (c) *Resistance to impersonation*: To impersonate a legitimate vehicle V_i, the adversary needs to generate a valid signature at the first transmission slot to extract a secret shared key used for subsequent transmissions. In this sense, the adversary must deduce V_i's private key $Sk_{V_i} \in Z_q^*$ from $Pk_{V_i} = Sk_{V_i}.g$ under the difficulty of solving the ECDLP. Hence, protecting against impersonation attacks.

7.3.3 Security proof based on AVISPA simulation

In this subsection, the AVISPA tool is used to analyse the security robustness of the proposed BCSKE scheme.

1. *Preliminaries*: In [143], Armando et al. developed the AVISPA toolkit, which is a widely used security protocol animator to validate and evaluate the security aspect of applications and internet security protocols. In AVISPA, the high-level protocol specification language (HLPSL) specifies the role played by each network terminal referred to by the agent, which verifies the security features regarding authentication and data secrecy of the exchanged messages between different agents in the presence of an intruder. Security properties are predefined in a separate section called "goals," based on which the security protocol is classified as SAFE or UNSAFE. As part of AVISPA's toolset, the HLPSL2IF translator is utilised to translate the HLPSL code into the intermediate format (IF), which is integrally crucial for offering and serving adequate input to the various back-ends of the tool. There are four back-ends provided by AVISPA: tree automata-based on automatic approximations for analysis of security protocol (TA4SP), SAT-based model checker (SATMC), on-the-fly model checker (OFMC), and constraint logic-based attack searcher (CL-AtSe). In this study, the simulation result of the BCSKE scheme is determined using the CL-AtSe back-end, which determines the protocol's resistance to MITM and replay attacks. Table 7.5 presents the BCSKE scheme used notations and their associated HLPSL scripting symbols.

Table 7.5: AVISPA symbols and their equivalent scheme notations

HLPSL variable	**Scheme notation**
inv(KVi)	Sk_{V_i}
KVi	Pk_{V_i}
TR	T_R
KVi.TR.{KVi.TR}_inv(KTA)	$Cert_{V_i}$
{x}_inv(k)	σ_i
Ti	T_i
Ki	$\hat{k}_{V_i}$
inv(KRSU)	Sk_{RSU_j}
KRSU	Pk_{RSU_j}
inv(KRTA)	Sk_{RTA}
KRTA	Pk_{RTA}
inv(KTA)	Sk_{TA}
KTA	Pk_{TA}
CS	$CS_{V_{1-2}}$
KV1.KV2.CS.{KV1.KV2.CS}_inv(KRTA)	$TxID$
.	$\|\|$

2. *Specifications for simulation*: As a first step, the security goals for the BCSKE simulation are specified, which include the secrecy of the extracted keys K1 and K2 between different agents referred to by sec_1, sec_2, sec_3, and sec_4, along with authenticating the broadcasted messages by the intended agent described by auth_1, auth_2, auth_3, auth_4, auth_5, auth_6, and auth_7. In the simulation, there are four agents' roles role_V1, role_V2, role_RSU, and role_RTA played by V1, V2, RSU, and RTA, respectively. During the role session, all the agents' declarations are defined, and in the role environment, all the variables and functions associated with different agents are denoted. Fig. 7.4 shows the protocol simulation in the form of transitions between different agents in the BCSKE scheme. A full explanation of these roles is presented in Appendix B in the form of HLPSL codes. Note that $/\backslash$ means a conjunction between two operations.

Code 1 in Appendix B shows the role played by V1 in the network. The knowledge of V1 includes all the protocol's agents (V1, V2, RSU, and RTA), their public keys (KV1, KV2, KRSU, and KRTA), TA's public key KTA, the symmetric key KV1Rta between RTA and V1 (equals $Sk_{V_1}.Pk_{RTA}$ using the Diffie-Hellman key exchanging protocol), and the send (SND) and receive (RCV) Dolev-Yao (dy) channels. The local variables part defines the role's initial state (State:=0), the certificate expiry date TR, the timestamps (T1, T2, T4, and T6), the probing packets of V1 and V2 (PPV1 and PPV2), the correction sequence CS, the extracted keys by V1 and V2 (K1 and K2), and the message M. The following are the three transitions that describe the role of V1.

- For State=0, and if V1 receives the start signal "RCV(start)" to execute the protocol, then the current state is increased by 1 (State':=1) and V1 sends the communication request containing the probing packet PPV1, a fresh timestamp T1', V1's certificate, and the message signature signed by V1's private key inv(KV1). Note that {x}_inv(y) represents the signature of the contents x using the agent's private key inv(y). Finally, V1 expects

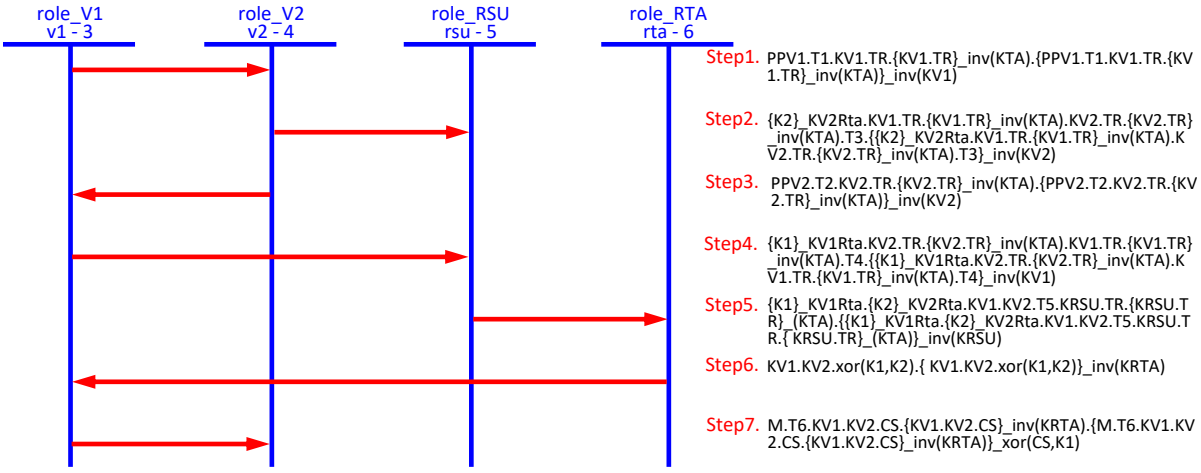


Figure 7.4: AVISPA protocol simulation.

that V2 authenticates PPV1 through a process named “auth_1”.

- For State=1, and if V1 receives a message from V2 containing the probing packet PPV2, a fresh timestamp T2’, V2’s certificate, and the message signature signed by V2’s private key inv(KV2), then the current state is increased by 1 (State’:=2) and V1 sends a message containing the encrypted key {K1’}_KV1Rta, V1’s certificate, V2’s certificate, a fresh timestamp T4’, and the message signature signed by V1’s private key inv(KV1). Finally, V1 verifies the received PPV2 from V2 through a process named “auth_2”, believes in the secrecy of the transmitted K1’ to the RSU through a process named “sec_1”, and expects that RSU authenticates {K1’}_KV1Rta through a process named “auth_4”.

- In the 3^{rd} transition, the obtained correction sequence from the blockchain is referred to as a received message from the RTA containing KV1, KV2, and CS’ signed by the RTA’s private key as {KV1.KV2.CS’}_inv(KRTA). For State=2, and if V1 receives a signed CS’ from the RTA, then V1 reconciles the mismatched bits by computing K2’:=xor(CS’,K1) and securely sends a message to V2 containing the safety-related message M, a fresh timestamp T6’, the CS’, and the message signature signed by the symmetric key K2’. Finally, V1 verifies the received CS’ from the RTA through a process named “auth_6”, and hopes that M will be authenticated by V2 through a process named “auth_7”.

Code 2 in Appendix B shows the role played by V2 in the network. The knowledge of V2 includes all the protocol’s agents (V2, V1, RSU, and RTA), their public keys (KV2, KV1, KRSU, and KRTA), TA’s public key KTA, the symmetric key KV2Rta between V2 and RTA (equals $Sk_{V_2}.Pk_{RTA}$ using the Diffie-Hellman key exchanging protocol), and the SND/RCV channels. The local variables part defines the role’s initial state (State:=0), the certificate expiry date TR, the timestamps (T1, T2, T3, and T6), the probing packets of V1 and V2 (PPV1 and PPV2), the correction sequence CS, the extracted keys by V1 and V2 (K1 and K2), and the message M. The following are the two transitions that describe the role of V2.

- For State=0, and if V2 receives the communication request from V1, then the current state is increased by 1 (State’:=1) and V2 sends

- a reply message to V1 containing the probing packet PPV2, a fresh timestamp T2’, V2’s certificate, and the message signature signed by V2’s private key inv(KV2). Finally, V2 verifies the received PPV1 from V1 through a process named “auth_1” and expects that V1 authenticates PPV2 through a process named “auth_2”.
- a message to the RSU containing the encrypted key {K2’}_KV2Rta using the symmetric key KV2Rta, V1’s certificate, V2’s certificate, a fresh timestamp T3’, and the message signature signed by V2’s private key inv(KV2). Finally, V2 believes in the secrecy of the transmitted K2’ to the RSU through a process named “sec_2” and

expects that RSU authenticates {K2'}_KV2Rta through a process named "auth_3".

- For State=1, and if V2 receives a message from V1 containing the safety-related message M, a fresh timestamp T6', the CS', and message signature signed by the symmetric key K2, then the current state is increased by 1 (State':=2). Finally, V2 verifies the received M from V1 through a process named "auth_7".

Code 3 in Appendix B shows the role played by the RSU in the network. The knowledge of the RSU includes all the protocol's agents (RSU, V1, V2, and RTA), their public keys (KRSU, KV1, KV2, and KRTA), TA's public key KTA, the SND/RCV channels. The local variables part defines the role's initial state (State:=0), the certificate expiry date TR, the timestamps (T3, T4, and T5), and the symmetric keys (K1, K2, KV1Rta, and KV2Rta). The following are the two transitions that describe the role of the RSU.

- For State=0, and if the RSU receives a message from V2 containing the encrypted key {K2'}_KV2Rta using the symmetric key KV2Rta, V1's certificate, V2's certificate, a fresh timestamp T3', and the message signature signed by V2's private key inv(KV2), then the current state is increased by 1 (State':=1). Finally, the RSU verifies the received {K2'} _KV2Rta' from V2 through a process named "auth_3".

- For State=1, and if the RSU receives a message from V1 containing the encrypted key {K1'}_KV1Rta, V1's certificate, V2's certificate, a fresh timestamp T4', and the message signature signed by V1's private key inv(KV1), then the current state is increased by 1 (State':=2) and the RSU sends a message to the RTA containing the encrypted keys {K1'}_KV1Rta' and {K2'}_KV2Rta', V1's certificate, V2's certificate, a fresh timestamp T5', and the message signature signed by the RSU's private key. Finally, the RSU verifies the received {K1'} _KV1Rta' from V1 through a process named "auth_4", believes in the secrecy of the transmitted K1' and K2' to the RTA through a process named "sec_3" and "sec_4", respectively, and expects that the RTA authenticates {K1'}_KV1Rta and {K2'}_KV2Rta through a process named "auth_5".

Code 4 in Appendix B shows the role played by the RTA in the network. The knowledge of the RTA includes all the protocol's agents (V2, V1, RSU, and RTA), their public keys (KV2, KV1, KRSU, and KRTA), TA's public key KTA, the symmetric key KV1Rta and KV2Rta, the SND/RCV channels. The local variables part defines the role's initial state (State:=0), the certificate expiry date TR, the timestamps T5, and the symmetric keys K1 and K2. There is a single transition played by the RTA denoted by

- For State=0, and if the RTA receives a message from the RSU containing the encrypted keys, then the current state is increased by 1 (State':=1), the RTA computes CS':=xor(K1', K2') and sends a message to V1 containing (KV1, KV2, CS') and the message signature signed by the RTA's private key inv(KRTA). Finally, the RTA verifies the received {K1'}

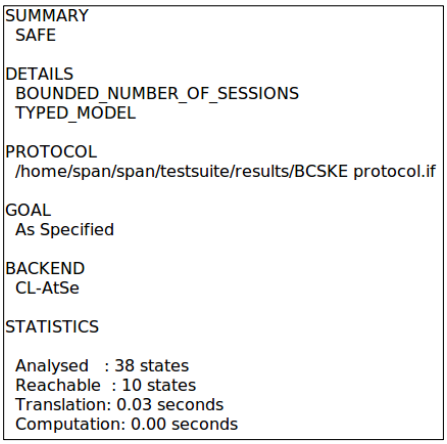


Figure 7.5: AVISPA simulation result using CL-AtSe.

_KV1Rta' and {K1'} _KV1Rta' from the RSU through a process named "auth_5", and expects that V1 authenticates CS' through a process named "auth_6".

As a final substep, **Code 5** in Appendix B defines the protocol variables and the intruder knowledge of all the network agents and their associated public keys. In addition, the same code outlines the protocol goals mentioned above.

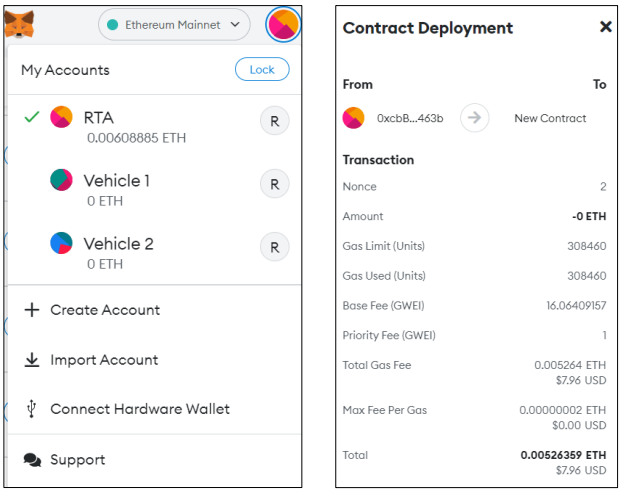
3. *Simulation results*: Based on the AVISPA security analysis, Fig. 7.5 summarises the simulation result of the specified goals using the CL-AtSe back-end checker. As can be seen, the CL-AtSe model takes 0.03 seconds for IF translation. According to the summary, the conclusion is that the BCSKE protocol is SAFE from potential MITM and replay attacks.

7.4 Performance analysis

7.4.1 Implementation and transaction fees

This part discusses the functionality of the BCSKE scheme by implementing it on the Ethereum main network, i.e., *Ethereum MainNet*. For triggering and deploying the proposed *SC*, the integration of the *Ethereum MainNet* and *Remix* 0.25.1 is achieved using *MetaMask* (a Microsoft Edge plug-in extension). For all *Metamask* wallets, *Ethereum MainNet* is the default network that is used by developers to develop and examine the actual performance of various types of decentralised applications. Following are the details of the BCSKE implementation.

1. As a first step, three accounts are created in the *MetaMask* corresponding to RTA, V_1, and V_2, denoted by 0xcbB3012b86b594223E43FB9c50176624F357463b, 0xa30C281D2Cf62 52e7524f38341fb6d1a8b68876B, and 0xA510aEe2869D2A2608C6D96d454d322BC67d 327A, respectively, as shown in Fig. 7.6(a). Then switched to the RTA's account and



(a) Network terminals. (b) Smart contract deployment.

Figure 7.6: Blockchain terminals and the *SC* deployment process in MetaMask.

charged it from the *Ethereum MainNet* such that the RTA can deploy and interact with the *SC*'s functions, i.e., **Deployer**, **IssueCS**, **Update**, and **Get**. Afterwards, the proposed *SC* is deployed into the blockchain and retrieved its address (*SCID*), denoted by 0x91feA69D12 8a4C82AffA49a70E95b35774e04738, as shown in Fig. 7.6(b). An overview of the *SC* deployment process is given in Fig. 7.7, which includes information on transaction fees, gas costs (Ethereum unit of measurement, i.e., ETH and Wei), etc.

2. Following that, the **IssueCS** function is invoked to deploy the correction sequence (*CS*) of the mismatched bits in a transaction (*Tx*) and obtain its related address (*TxID*). Next, the *TxID* is mapped to both of the communicating vehicles' public keys (Pk_{V_1}, Pk_{V_2}) using the **Update** function.

3. Finally, the switch to V_1 is made to obtain the *CS* of the transaction (*Tx*) by invoking the **Get** function to attain *TxID*.

An example of how the network terminals interact with the *SC*'s different functions is explained in a four-step process using the *Remix Virtual Machine* - see Fig. 7.8, as follows.

- *Step 1*: Using the **Deployer** function, the *SC* is deployed by the RTA.
- *Step 2*: Assuming Pk_{V_1} = 847932647234870754 345, Pk_{V_2} = 234354679832720343455, and *CS* = 1054342 37269874323123, the *CS* is published via a transaction and retrieved its related address, e.g., *TxID* = 782353723486592342345 using the **IssueCS** function.
- *Step 3*: The **Update** function maps the obtained *TxID* to Pk_{V_1} and Pk_{V_2}.
- *Step 4*: The **Get** function is used to call the *TxID* corresponding to the input public keys Pk_{V_1} and Pk_{V_2}.

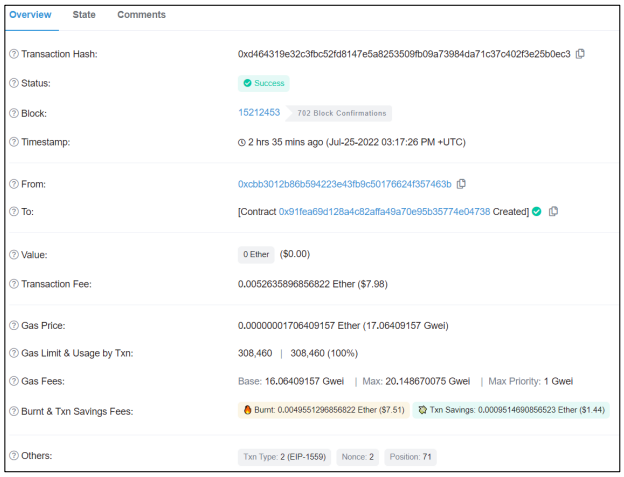


Figure 7.7: The *SC* deployment details.

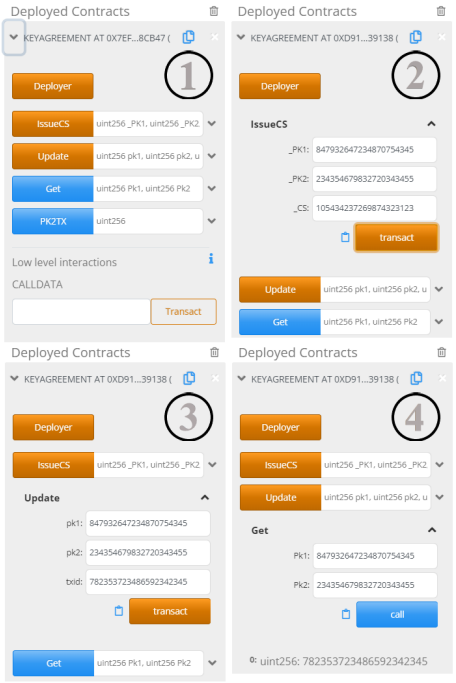


Figure 7.8: An example of the *SC* functionality.

In addition, the gas costs associated with the *SC*'s functions are evaluated, as shown in Table 7.6. As can be seen, the *SC* deployment is the most costly phase, but it only needs to be performed once. In contrast to all the other functions, the **IssueCS** and **Update** functions are invoked at the beginning of each session. While the **Get** function is invoked by the recipient for every received safety-related message. A comparison of transaction time and fee costs for various blockchains is shown in Table 7.7 [144–146]. Among its rivals, it is noteworthy that Aleph Zero is the fastest blockchain ($\sim$ 600 *msec*) with the lowest transaction fees ($\sim$ 0.0003 \$). In each session, based on Aleph Zero statistics, the communicating vehicles lose from two to six safety-related messages since the broadcasting rate is a message every 100 - 300 *msec*.

Table 7.6: Gas costs for different *SC*'s functions (1 ETH = 1,859.89 $)

Function	**Gas used (Gwei)**	**Actual cost (ETH)**
Deployer	17.06409157	0.0052635896856822
IssueCS	15.260894856	0.00139812688223244
Update	15.465276944	0.001416851346989
Get	No fees	No fees

Table 7.7: A comparison of blockchains' transaction fees and costs ($) [144–146]

Blockchain	Tx**'s time**	Tx**'s cost**
Aleph Zero	$\sim$ 0.6 *sec*	$\sim$ 0.0003 $
Avalanche	$\sim$ 1-2 *sec*	1 $ for transaction fee $\leq$ 20 $
		5% for transaction fee $>$ 20 $
Digibyte	$\sim$ 2-3 *min*	$\sim$ 0.0005 $
Dash	$\sim$ 6 *min*	$\sim$ 0.2 - 0.3 $
Litecoin	$\sim$ 30 *min*	$\sim$ 0.007 $
Tezos	$\sim$ 30 *min*	$\sim$ 0.01 $

Fortunately, blockchain technology is rapidly developing, which can help mitigate this loss in the future.

7.4.2 Comparative analysis of computation cost

This subsection conducts a performance evaluation, comparing it with the methodologies presented in [57–60, 140]. Specifically, [57, 58] introduced CPPA schemes utilising ECC-based scalar multiplication and addition operations for signature generation and verification. Meanwhile, [59, 60, 140] introduced certificate-less authentication schemes, aiming to minimise authentication overheads and promote privacy. In this context, the same evaluation of the time consumed by different cryptographic operations is invoked from [147], see Table 7.8. These operations are measured using the MIRACL library [126] running on quad-core i7-4790 CPU, 16 GB RAM, and Ubuntu 20.04-desktop-amd64. From Table 7.8, T_{mul}^{ECC} and T_{add}^{ECC} are the scalar multiplication and addition operations in the ECC-based group $\mathbb{G}$. While T_{mul}^{BP}, T_{add}^{BP}, and T_{BP} are the scalar multiplication, addition, and bilinear pairing operations in BP-based group $\mathbb{G}_1$. Finally, T_{exp}, T_h, T_{enc}^{AES}, and T_{dec}^{AES} are the exponentiation, SHA-256 hashing, encryption, and decryption operations using the AES algorithm.

Table 7.9 presents a comparison of computation and communication costs between the BCSKE scheme and those described in [57–60, 140]. As can be seen, [57] demonstrates that the vehicle needs two ECC-based scalar multiplication and two hashing operations to sign a single message. Thus, the total run time for the signature generation process is $2T_{mul}^{ECC} + 2T_h \approx$

Table 7.8: The cost of computing different cryptographic operations in *msec*

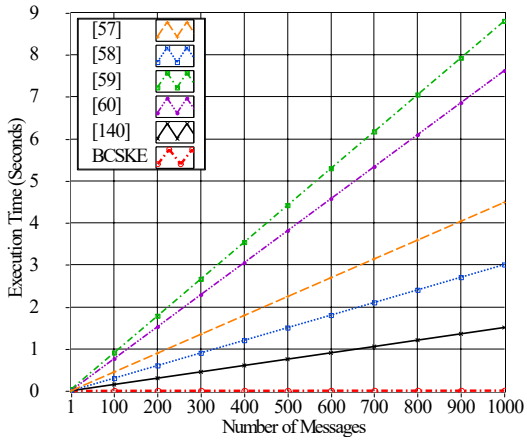
Definition	**Symbol**	**Time**
ECC-based scalar multiplication in $\mathbb{G}$	T_{mul}^{ECC}	1.489
ECC-based Point addition in $\mathbb{G}$	T_{add}^{ECC}	0.008
BP-based scalar multiplication in $\mathbb{G}_1$	T_{mul}^{BP}	2.521
BP-based point addition in $\mathbb{G}_1$	T_{add}^{BP}	0.018
BP operation in $\mathbb{G}_1$	T_{BP}	13.44
Exponentiation operation	T_{exp}	1.864
Hash function operation using SHA-256	T_h	0.003
The AES encryption operation	T_{enc}^{AES}	0.002
The AES decryption operation	T_{dec}^{AES}	0.001

Table 7.9: Comparative analyses of computation and communication costs

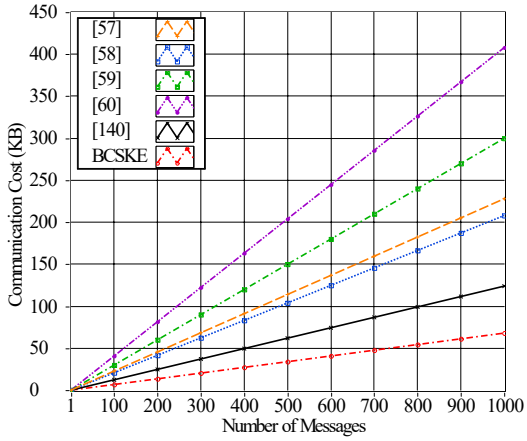
Schemes	**Computation cost in *msec***		**Communication cost in *bytes***
	Signature generation	**Signature verification of n messages**	
Sutrala et al. [57]	$2T_{mul}^{ECC}+2T_h$	$(3n)T_{mul}^{ECC}+(2n)T_{add}^{ECC}+(2n)T_h$	$228n$
Ming et al. [58]	$3T_{mul}^{ECC}+2T_h$	$(2n+2)T_{mul}^{ECC}+(2n+1)T_{add}^{ECC}+(3n)T_h$	$208n$
Tan et al. [59]	$6T_{mul}^{BP}+2T_{add}^{BP}+3T_h$	$2T_{BP}+(2n+2)T_{mul}^{BP}+(2n)T_{exp}+(2n+3)T_h$	$300n$
Li et al. [60]	$3T_{mul}^{BP}+2T_{add}^{BP}+1T_h$	$(3n+2)T_{mul}^{BP}+(3n)T_{add}^{BP}+(n)T_h$	$408n$
Ogundoyin et al. [140]	$T_{mul}^{ECC}+T_{add}^{ECC}+2T_h$	$(n+1)T_{mul}^{ECC}+(2n)T_{add}^{ECC}+(n)T_h$	$124n$
BCSKE	$1T_{\text{enc}}^{AES}+1T_h$	$(n)T_{dec}^{AES}+(n)T_h$	$68n$

2.984 *msec*. While the time cost to verify a number of n received signatures comprises $3n$ ECC-based scalar multiplication, $2n$ ECC-based addition, and $2n$ hashing operations, so the total run time for the signature verification process is $(3n)T_{mul}^{ECC}+(2n)T_{add}^{ECC}+(2n)T_h \approx 4.489n$ *msec*. The same goes for [58–60, 140].

The calculations for the BCSKE are based on the time consumed to send n safety-related messages, ignoring the key agreement process since it only occurs once per session. The time consumed to sign a single message includes one hashing operation and one AES-based encryption operation, so the total run time for the signature generation is $1T_{enc}^{AES}+1T_h \approx 0.005$ *msec*. While the time consumed to verify n messages includes n hashing operations and n AES-based decryption operations, the total run time for the signature verification is $(n)T_{dec}^{AES}+(n)T_h \approx 0.004n$ *msec*. Therefore, the computation costs for generating and verifying signatures in the BCSKE scheme are lower than those in [57]. Similarly, the computation costs in the BCSKE scheme (i.e., signature generation and verification) are lower than those of [58–60, 140], as shown in Fig. 7.9(a).



(a) The computation cost for different numbers of messages.



(b) The communication cost for different numbers of messages.

Figure 7.9: Comparison of computation and communication costs.

7.4.3 Comparative analysis of communication cost

This subsection evaluates the communication cost of the proposed scheme. For comparison, the curve equation $E : y^2 = x^3 + x \; mod \; p$ is used for the bilinear pairing map $e : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_T$, where $\mathbb{G}_1$ and $\mathbb{G}_T$ represent additive group and multiplicative group with order q_1, respectively. For the 80-bit security level, the curve type "*SS*512" is adopted [148], where p is a large prime number of order 512 bits (64 bytes) [66]. Thus, the size of the element in the BP-based group $\mathbb{G}_1$ is $2 \times 512 = 1024$ bits (128 bytes). While the size of the element in the ECC-based group $\mathbb{G}$ described in Table 7.2 is 320 bits (40 bytes). As for the element size in Z_q^* and $Z_{q_1}^*$, the timestamp, and the hash function output, they are respectively 160 bits (20 bytes; the security level of the 1024-bit RSA key length [149]), 32 bits (4 bytes), and 160 bits (20 bytes). The lengths of the ECC and BP parameters are summarised in Table 7.10 [46]. This calculation only considers the size of the signatures attached to the safety-related messages since all the schemes have the same message size. In [57], the transmitted signature is the set of elements $\{\{f_i, g_i\}, B_i, K_i, R_i, \{pid_{1i}, pid_{2i}, T_i\}, T_1\}$ in which $\{pid_{1i}, K_i, B_i, R_i\} \in \mathbb{G}$, $\{f_i, g_i, pid_{2i}\} \in Z_q^*$, and T_1 and T_i are the timestamp and the expiry time of the pseudo-

Table 7.10: Parameters of ECC and bilinear pairing [46]

Scheme	Curve type	Pairing	Cyclic group	Length of p	Length of a group point
ECC	$E: y^2 = x^3 + ax + b \ mod \ p$	Pairing Free	$\mathbb{G}(p)$	160 bits	$\|\mathbb{G}\| = 320$ bits
BP	$E: y^2 = x^3 + x \ mod \ p$	$\mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_T$	$\mathbb{G}_1(p)$	521 bits	$\|\mathbb{G}_1\| = 1024$ bits

identity, respectively. Hence, the signature size is $4 \times 40 + 4 \times 20 + 2 \times 4 = 228$ *bytes*. In [58], the transmitted signature is the set of elements $\{\{PID_{i,1}, PID_{i,2}, T_i\}, t_i, P_i, D_i, R_i, \sigma_i\}$ in which $\{PID_{i,1}, P_i, D_i, R_i\} \in \mathbb{G}$, $\{PID_{i,2}, \sigma_i\} \in Z_q^*$, and t_i and T_i are the timestamp and the expiry time of the pseudonym, respectively. Thus, the signature size is $4 \times 40 + 2 \times 20 + 2 \times 4 = 208$ *bytes*. In [59], the transmitted signature is the set of elements $\{ID_i, TS_2, R_i, A_i, \mathfrak{L}_i\}$ in which $\{R_i, \mathfrak{L}_i\} \in \mathbb{G}_1$, $\{ID_i, A_i\} \in Z_{q1}^*$, and TS_2 is the timestamp. Thus, the signature size is $2 \times 128 + 2 \times 20 + 4 = 300$ *bytes*. In [60], the transmitted signature is the set of elements $\{R_{ui}, K_{ui}{}', KG_{ui}{}', t_i, \rho_{ui}\}$ in which $\{K_{ui}{}', KG_{ui}{}', R_{ui}\} \in \mathbb{G}_1$, $\rho_{ui} \in Z_{q_1}^*$, and t_i is the timestamp. Hence, the signature size is $3 \times 128 + 20 + 4 = 408$ *bytes*. In [140], the transmitted signature is the set of elements $\{PID_i^k, R_i, \vartheta_i PK_i, t_i^{cur}\}$ in which $\{R_i, PK_i\} \in \mathbb{G}$, $\{PID_i^k, \vartheta_i\} \in Z_q^*$, and t_i^{cur} is the timestamp. Hence, the signature size is $2 \times 40 + 2 \times 20 + 4 = 124$ *bytes*. For the BCSKE scheme, the transmitted signature is the set of elements $\{T_6, TxID, \sigma_6\}$ in which the size of $TxID$ and σ_6 are each equal to 32 *bytes*. Therefore, the signature size is $2 \times 32 + 4 = 68$ *bytes*. Based on this analysis, it can be concluded that the communication cost in the BCSKE scheme is lower than that of [57–60, 140], as shown in Fig. 7.9(b).

7.4.4 Simulation analysis

In this subsection, the discrete event simulator OMNeT++ 5.6.2 [150] is used in conjunction with a network simulator (i.e., Sumo 1.8.0 [151], INET 4.2.5 [152], and Veins 5.2 [153]) to carry out the vehicular simulation to analyse the network performance of the BCSKE scheme. The simulation uses the IEEE 802.11p standard in a $2500 \times 2500\ m^2$ area, and all the other simulation parameters are listed in Table 7.11. The performance of the BCSKE scheme is investigated from the standpoint of average packet delay and packet loss ratio and compared to those of [57–60, 140].

1. The average authentication delay (AAD): It is also called the end-to-end packet delay, which consists of the sum of the message transmission and verification delays, which can be computed using the following formula.

$$AAD = \frac{1}{N} \sum_{i=1}^{N} \left(\frac{1}{n_i} \sum_{j=1}^{n_i} (T_{rec}^j - T_{send}^j) \right) \tag{7.11}$$

where n_i is the number of received messages by the vehicle V_i, N is the total number of

Table 7.11: OMNeT++ simulation parameters

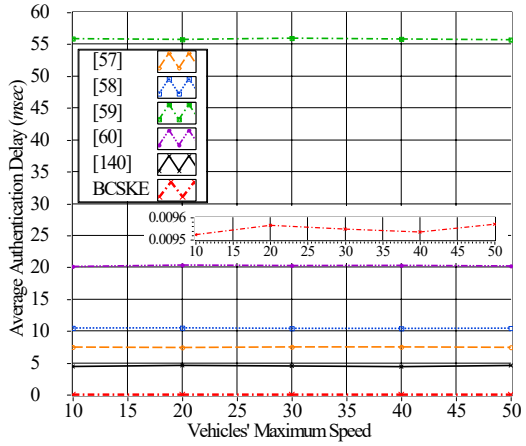
Parameter	**Value**
Simulation area	$2500 \times 2500\ m^2$
Duration of simulation	200 *sec*
Number of vehicles	10, 20, 30, 40, and 50
Vehicles' maximum speed (m/s)	10, 20, 30, 40, and 50
MAC layer protocol	802.11p
Transmission power	50 *mW*
Data transmission rate	6 Mbps
Broadcasting rate	100 *msec*
Receiver sensitivity	-110 dBm
Noise floor, e.g., thermal noise, etc	-98 dBm
Maximum interference distance	2500 *m*
Antenna type	Monopole antenna
Used channel	The control channel (CCH)
Vehicle's length	2.5 *m*
Minimum gap between vehicles	2 *m*
Vehicles' acceleration	$2\ m/s^2$
Vehicles' deceleration	$3\ m/s^2$

vehicles inside the network, and T^j_{send} and T^j_{rec} are the sending and receiving time of the message m_j. The simulation scenario is established such that a fixed number of moving vehicles (i.e., 20 vehicles) periodically send emergency traffic messages every 100 *msec*. In Fig. 7.10(a), the authentication delay for moving vehicles is shown at different speeds (i.e., 10, 20, 30, 40, 50 m/s). As can be seen from the figure, the authentication delay is approximately constant at different speeds. The varying vehicles' speeds, accelerations, and decelerations cause a slight fluctuation in the distance between the communicating vehicles, resulting in subtle variations in the average authentication delay caused by the negligible propagation delay. The BCSKE scheme demonstrates the lowest average authentication delay among its competitors, approximately ~ 0.0095 *msec*.

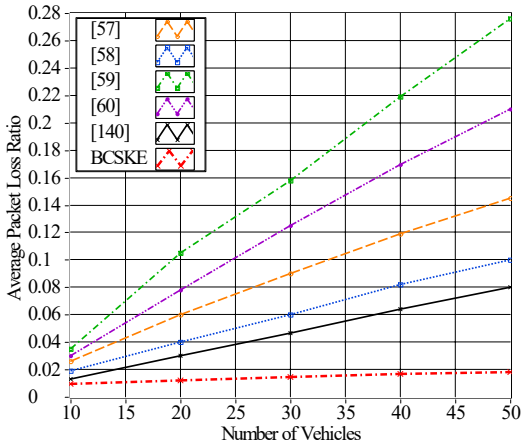
2. The average packet loss ratio (APLR): The ratio between lost packets and transmitted packets reflects the packet loss ratio, which can be computed using the following formula.

$$APLR = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{N^i_{lost}}{N^i_{rec}+N^i_{lost}}\right) \tag{7.12}$$

where N^i_{rec} and N^i_{lost} are the number of successfully received and lost data packets from the vehicle V_i, respectively. In Fig. 7.10(b), the packet loss ratio for a different number of vehicles (i.e., 10, 20, 30, 40, 50 vehicles) is shown at a maximum speed of moving vehicles of 20 m/s. The figure clearly shows that the packet loss ratio of all the schemes (i.e., the



(a) Average authentication delay at different speeds.



(b) Average packet loss ratio at different densities.

Figure 7.10: OMNeT++ simulation results.

number of dropped packets) increases as the number of vehicles increases. However, it is worth noting that the slopes of [59, 60] are much higher than that of [57, 58, 140] because [59, 60] have bilinear pairing-based verification processes and the largest sizes of transmitted messages. While the slopes of [57, 58, 140] are higher than that of the BCSKE scheme because [57, 58, 140] have ECC-based verification processes. Based on these results, it can be concluded that the BCSKE scheme outperforms its competitors in [57–60, 140].

7.5 Summary

In accordance with the contributions listed, this study introduces a lightweight *SKC*-based re-authentication method that supports forward and backward secrecy (1^{st} contribution). For key agreement, an efficient secret key extraction-based authentication scheme is proposed for VANETs, leveraging the immutability of blockchain technology to design a lightweight smart contract for reconciling the mismatched bits incurred by the channel non-reciprocity components (2^{nd} contribution). In addition, the correctness and security robustness of the BCSKE scheme is proven using the BAN-logic and AVISPA simulator, respectively, demonstrating the scheme's resistance to common adversarial attacks (3^{rd} contribution). The effectiveness of the BCSKE scheme in high-density traffic environments is emphasised, and it has been demonstrated that the required time to verify 1000 messages is improved by $\sim 99\%$ compared to previous studies [57–60, 140]. While the communication cost is improved by 70%, 67%, 77%, 83%, and 45% in comparison to that of [57–60, 140], respectively (4^{th} contribution). Using OMNeT++, it is demonstrated that the proposed scheme offers a lower authentication delay and packet loss ratio than competing approaches. The next chapter explores the possibility of designing a smart contract-based blockchain for an efficient group key distribution of signature-based authentication in VANETs.

Chapter 8

Blockchain-based Group Key Distribution

Existing GS-based schemes suffer from high computation and communication costs for generating and distributing group public and private keys. In addition, these keys must be updated periodically to provide forward and backward secrecy. To overcome these limitations, blockchain-based authentication has emerged in recent studies. In [154], Otoum et al. developed a Federated Learning-based framework for authenticating transactions in a decentralised pattern. Lu et al. [40] employed the blockchain to develop a proof of revocation and issuance of certificates. Son et al. [147] proposed a consortium blockchain-based V2I handover authentication protocol. Nevertheless, developing a reliable group key distribution method remains challenging, particularly in high mobility and dense traffic environments.

The following summarises the contributions of this chapter which are published in [25], fulfilling the outlined thesis objectives (3, 5, 6) detailed in Subsection 1.4.3:

1. A blockchain-based group key distribution method is proposed, allowing the RSU to act as a group manager in distributing and updating the group session key among group members with minimal communication and computation costs using a smart contract. Accordingly, a lightweight GS-based message authentication process is developed.

2. The smart contract's functionality is evaluated by implementing its built-in functions and measuring its associated gas costs using Ethereum's main network (*MainNet*).

3. Besides security analysis, the proposed scheme is extensively compared to conventional approaches to prove its superiority in reducing the computation and communication costs of verifying and transmitting messages.

This chapter is organised as follows. Section 8.1 describes the proposed scheme. Sections 8.2 and 8.3 evaluate the security and performance aspects. Finally, Section 8.4 concludes the chapter.

8.1 The proposed scheme

This section goes into detail about the system and scheme modelling. For simplicity, Table 8.1 lists the scheme's notations.

8.1.1 System modelling

The network comprises the following entities, see Fig. 8.1.

1. *TA*: The TA is a trusted third party that initialises the system's parameters and registers network terminals. It is the only terminal that holds the link between the vehicle's real identity and its digital certificate. It also can expose the vehicle's real identity in case of misbehaving (i.e., constructing attacks or driving an unregistered vehicle).

2. *RSUs*: The RSU_k serves as a group manager responsible for the enrolment/revocation of vehicles getting in/out from its coverage area. In addition, it updates the group session key K_{GS} dynamically to ensure forward and backward secrecy. It is able to deploy and interact with its smart contract SC_{RSU_k} through transactions in the blockchain.

3. *OBUs*: Each vehicle serves as a group member in a specific region and has a wireless communication device to communicate with surrounding vehicles. It is also capable of accessing the blockchain network and invoking the **ViewGSK** function using SC_{RSU_k}.

4. *The proposed smart contract-based blockchain*: Algorithm (1) presents the smart contract of the K_{GS} distribution process with a command-by-command explanation. In the proposed smart contract, four functions are involved: **Deployer**, **IssueGSK**, **UpdateGSK**, and **ViewGSK**. The **Deployer**( ) function is used to define the owner of the smart contract *SC*. In this scenario, the RSU_k in each region acts as the owner and the deployer of the

Table 8.1: List of notations for the proposed group key distribution scheme

Symbol	Definition
Sk_{TA}, Pk_{TA}	The system secret and private keys, respectively
Sk_{RSU_k}, Pk_{RSU_k}	RSU_k's private and public keys, respectively
Sk_{V_i}, Pk_{V_i}	V_i's private and public keys, respectively
K_{GS}, C_{GS}	The group session key and its encrypted parameter
SC_{RSU_k}	The smart contract deployed by RSU_k
$SCID_{RSU_k}$	The smart contract's address
T_i, T_r	σ_i's timestamp and receiving time, respectively
T_Δ	The freshness expiry period [00:00:59]
PID_{V_i}	The pseudo-identity of the vehicle V_i
$Cert$, T_R	The terminal's digital certificate and its expiry date

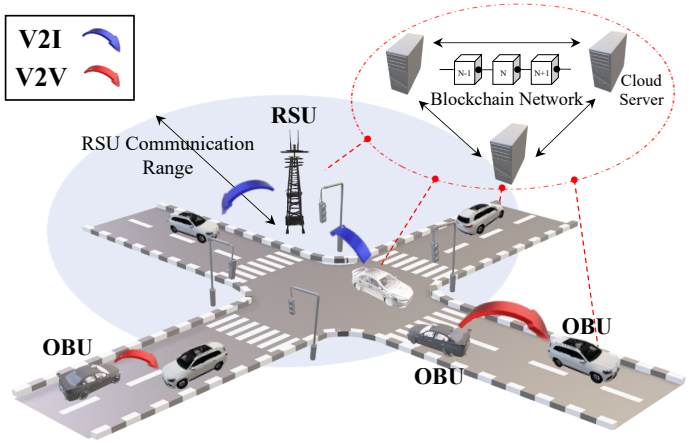


Figure 8.1: System modelling for the proposed group key distribution scheme.

SC_{RSU_k}. The **IssueGSK**$(uint\_PID, uint\_CGS)$ function can only be invoked by the owner RSU_k to publish a transaction Tx contains the encrypted group key C_{GS} associated to the vehicle V_i's pseudo-identity PID_{V_i}, retrieving the published Tx's address $TxID$. Similarly, the **UpdateGSK**$(uint\_PID, uint\_TxID)$ function can only be invoked by the owner RSU_k

Algorithm 1: Smart Contract for **GSKdistribution**

Given: function name, parameter settings

Require: Setting up functions

```
struct V2V {uint PID; uint CGS;
} //Defining the types of the input parameters
address RSU = 0xcbb21012b86b594223E43FB9c5017662
4F357463b; //Defining the address of the RSU
mapping (uint → uint256) private PID2TX;//Defining a local
function "PID2TX" that maps PID_Vi to TxID
function Deployer ( ) public {RSU = msg.sender;
} //Defining the SC's deployer as the RSU
modifier onlyowner {require (msg.sender == RSU);
_;}//Only the RSU can successfully run the Deployer function
V2V GSKdistribution1;
function IssueGSK (uint _PID, uint _CGS)
onlyowner public returns (uint, uint) {
GSKdistribution1.PID = _PID;
GSKdistribution1.CGS = _CGS;
return (GSKdistribution1.PID, GSKdistribution1.CGS)
} //Publishing a transaction Tx by the owner "RSU", which
contains C_GS associated with PID_Vi and retrieving TxID
function UpdateGSK (uint PID, uint256 TxID)
onlyowner public {PID2TX [PID] = TxID;
} //The owner "RSU" maps PID_Vi to TxID.
function ViewGSK (uint PID) public view returns
(uint256) {return PID2TX [PID];
} //This function can be invoked by any vehicle V_i using its
corresponding PID_Vi to retrieve TxID
```

and it is used to map the retrieved $TxID$ to PID_{V_i}. At last, the **ViewGSK**($uint\_PID$) function is invoked by V_i to retrieve $TxID$ related to PID_{V_i}. Using $TxID$, V_i obtains Tx's contents, C_{GS}, from the blockchain.

Note that, this scheme adopts the same blockchain network architecture presented in Chapter 7.

8.1.2 Scheme modelling

The proposed blockchain-based group signature scheme involves four phases, i.e., initialisation, registration, group session key generation, signature generation and verification.

Initialisation phase

The TA performs the following steps to initialise the system's public and private parameters.

- The TA chooses two prime numbers, p and q, with a length of 160 $bits$ used to initialise the elliptic curve $E : y^2 = x^3 + ax + b \; mod \; p$, where $(a,b) \in Z_q^*$ in a condition of $\Delta = 4a^3 + 27b^2 \neq 0$.
- The TA chooses the generator g of length q and creates the cyclic additive group $\mathbb{G}$ that combines all points on E along with the infinity point $\mathscr{O}$.
- The TA randomly chooses the system secret key $Sk_{TA} \in Z_q^*$, then computes its related public parameter $Pk_{TA} = Sk_{TA}.g$. In addition, the TA chooses the SHA-256 hash function $H_1 : \{0,1\}^* \rightarrow \{0,1\}^{N_1}$, where $N_1 = 256$ $bits$.
- Finally, the public parameters are $PPs = \langle a, b, p, q, g, Pk_{TA}, H_1 \rangle$.

Registration phase

The TA performs the following steps to register all network terminals.

- For each RSU, the TA publishes RSU_k's smart contract SC_{RSU_k} and retrieves its associated address $SCID_{RSU_k}$. After that, the TA chooses the RSU_k's private key $Sk_{RSU_k} \in Z_q^*$ and computes its related public parameter $Pk_{RSU_k} = Sk_{RSU_k}.g$. Then, the TA generates RSU_k's long term digital certificate $Cert_{RSU_k} = \langle Pk_{RSU_k}, T_R, \sigma_{TA} \rangle$, where T_R is the expiry date and $\sigma_{TA} = \text{Sign}_{Sk_{TA}}(Pk_{RSU_k} \| T_R)$. At last, the TA stores $\langle PPs, Sk_{RSU_k}, Cert_{RSU_k}, SCID_{RSU_k} \rangle$ onto RSU_k.
- As for each vehicle V_i, the TA checks the V_i's real identity RID_{V_i}, chooses the V_i's private key $Sk_{V_i} \in Z_q^*$ and computes its related public parameter $Pk_{V_i} = Sk_{V_i}.g$. Then, the TA generates V_i's long term digital certificate $Cert_{V_i} = \langle Pk_{V_i}, T_R, \sigma_{TA} \rangle$, where $\sigma_{TA} = \text{Sign}_{Sk_{TA}}(Pk_{V_i} \| T_R)$. At last, the TA stores $\langle PPs, Sk_{V_i}, Cert_{V_i} \rangle$ onto V_i.

Group session key generation phase

As shown in Fig. 8.2, this phase comprises the following steps:

- *Step 1*: In each region, there is a RSU_k that periodically broadcasts an enrollment message in the form of $\langle T_1, SCID_{RSU_k}, Cert_{RSU_k}, \sigma_1 \rangle$, where T_1 is the timestamp and the signature $\sigma_1 = \text{Sign}_{Sk_{RSU_k}}(T_1 \| SCID_{RSU_k} \| Cert_{RSU_k})$.
- *Step 2*: For each vehicle V_i in the communication range of the RSU_k, V_i checks T_1's freshness by finding out if $T_r - T_1 \leq T_\Delta$ holds or not to avoid replay attacks, verifies the signature σ_1 as $\text{Verf}_{Pk_{RSU_k}}(\sigma_1)$ to avoid impersonation attacks, and checks if $Cert_{RSU_k} \in CRL$. Then, V_i replies with a message in the form of $\langle T_2, M_{V_i}, Cert_{V_i}, \sigma_2 \rangle$, where $M_{V_i} = \text{Enc}_{K_{V_i-RSU_k}}(PID_{V_i})$, $K_{V_i-RSU_k} = Sk_{V_i}.Pk_{RSU_k}$, PID_{V_i} is a random number $\{0,1\}^{N_2}$ of length $N_2 = 256$ bits chosen by V_i, and $\sigma_2 = \text{Sign}_{Sk_{V_i}}(T_2 \| M_{V_i} \| Cert_{V_i})$.
- *Step 3*: The RSU_k in turn checks T_2's freshness, verifies the signature σ_2 as $\text{Verf}_{Pk_{V_i}}(\sigma_2)$, checks if $Cert_{V_i} \in CRL$, then decrypts M_{V_i} to get PID_{V_i} as $\text{Dec}_{K_{V_i-RSU_k}}(M_{V_i})$, where $K_{V_i-RSU_k} = Sk_{RSU_k}.Pk_{V_i}$ (using Diffie-Hellman key exchanging protocol). At last, RSU_k stores $Cert_{V_i}$ and its associated PID_{V_i}.
- *Step 4*: The RSU_k encrypts the group session key K_{GS} to get $C_{GS} = \text{Enc}_{K_{V_i-RSU_k}}(K_{GS})$ and uses the **IssueGSK**(PID_{V_i}, C_{GS}) function to publish C_{GS} related to PID_{V_i} through a transaction Tx. At last, RSU_k maps the transaction address $TxID$ to PID_{V_i} using the **UpdateGSK**(PID_{V_i}, $TxID$) function.
- *Step 5*: Finally, V_i retrieves $TxID$ by calling the **ViewGSK**(PID_{V_i}) function using $SCID_{RSU_k}$. By using $TxID$, V_i can obtain the transaction Tx information, including C_{GS}. At last, V_i decrypts C_{GS} to get K_{GS} as $\text{Dec}_{K_{V_i-RSU_k}}(C_{GS})$.

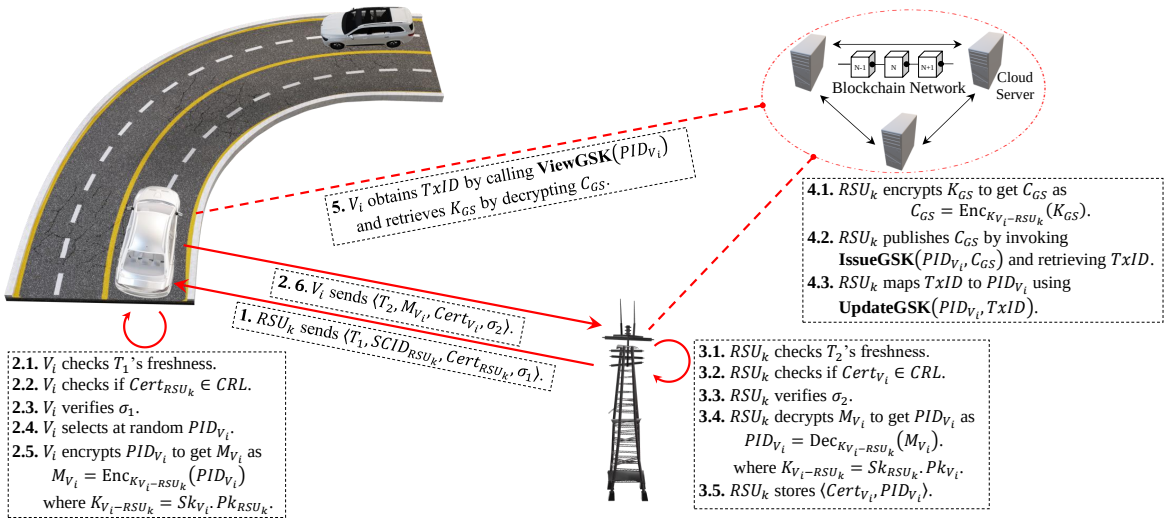


Figure 8.2: Group session key distribution process.

Note that the *SC*'s **IssueGSK** and **UpdateGSK** functions allow the RSU_k to dynamically update K_{GS} of group members without incurring an additional communication cost.

Signature generation and verification phase

In this phase, the signature is generated by V_i and verified by the group members V_j (i.e., surrounding vehicles) $\forall j \in [1, N-1]$, where N is the total number of vehicles in the communication range of RSU_k. This phase is presented in a two-step process.

- *Step 1*: V_i broadcasts a safety-related message m to surrounding vehicles in the form of $\langle m, T_3, PID_{V_i}, \sigma_3 \rangle$, where $\sigma_3 = \text{Enc}_{K_{GS}}(H_1(m \| T_3 \| PID_{V_i}))$.
- *Step 2*: $\forall j \in [1, N-1]$, V_j checks T_3's freshness and verifies σ_3 by testing if $H_1(m \| T_3 \| PID_{V_i}) \stackrel{?}{=} \text{Dec}_{K_{GS}}(\sigma_3)$ holds or not.

Note that, the signature is generated using symmetric key cryptography, a choice made for its lower computational cost compared to public key cryptography. This way optimises efficiency without compromising security.

8.2 Security analysis

This section shows that the proposed scheme complies with VANET security and privacy requirements.

8.2.1 Message authentication

The proposed scheme allows the group manager RSU_k to initially authenticate V_i using TA's signature $\sigma_{TA} \in Cert_{V_i}$, which proves V_i's ownership to Pk_{V_i}. Thus, it is hard to forge a valid signature signed by Sk_{V_i} under the difficulty of solving the ECDLP. While V_j verifies V_i's signature for subsequent transmissions by checking whether $H_1(m \| T_i \| PID_{V_i}) \stackrel{?}{=} \text{Dec}_{K_{GS}}(\sigma_i)$ holds, wherein σ_i's security level depends on the key length $|K_{GS}|$ used for generating σ_i using symmetric key cryptography.

8.2.2 Conditional privacy/identity anonymity

Conditional privacy is maintained since only TA retains the link between $Cert_{V_i}$ and RID_{V_i}, preventing the identification of RID_{V_i} by any other terminals inside the network.

1. *Unlinkability*: The proposed SC_{RSU_k} supports unlinkability as no terminal can link between $Cert_{V_i}$ and the dynamically updated PID_{V_i} since PID_{V_i} is sent encrypted to RSU_k and published decrypted in the blockchain, making it infeasible to track V_i's transmitted messages from different sessions.

2. *Resistance to active attacks*: This scheme proves to be resistant to the following attacks.

   (a) *Resistance to modification*: To modify the message contents, an attacker needs to forge a valid signature which is infeasible without having the group session key K_{GS}. Therefore, the recipient can easily detect modification attacks by verifying the attached signature $H_1(m\|T_i\|PID_{V_i}) \stackrel{?}{=} \text{Dec}_{K_{GS}}(\sigma_i)$.

   (b) *Resistance to impersonation*: To impersonate V_i, an attacker needs to generate a valid signature using Sk_{V_i} at the first transmission slot. In other words, the attacker needs to deduce Sk_{V_i} from $Pk_{V_i} = Sk_{V_i}.g$ under the difficulty of solving the ECDLP.

   (c) *Resistance to replaying*: The attached timestamp T_i allows the recipient to verify the received messages' freshness in the same session by checking if $T_r - T_i \leq T_\Delta$ holds. While the dynamically updated K_{GS} resists replaying attacks from different sessions, supporting forward and backward secrecy.

8.3 Performance analysis

This section evaluates the performance of the proposed scheme by implementing it on the Ethereum blockchain and measuring its computation and communication costs.

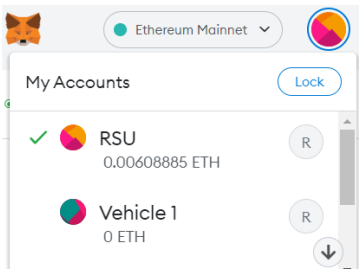
8.3.1 Implementation in the Ethereum blockchain

To discuss the feasibility of the proposed scheme, SC_{RSU_k} is implemented on the *Remix* 0.25.4 compiler, an open-source smart contracts-based blockchain system. *Remix*-based smart contracts are written in *Solidity*, a javascript-like language. Using the *MeteMask*, a *chrome* plug-in extension, the deployment and interaction with the functions of SC_{RSU_k} are performed on the *Ethereum MainNet*. Following are the details of the implementation.

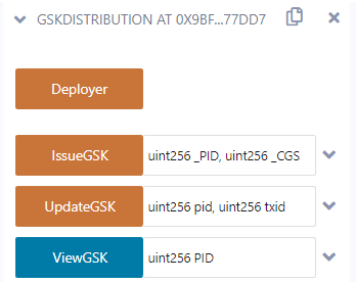
1. In *MeteMask*, two accounts with different addresses are set up for RSU_k and V_i, as depicted in Fig. 8.3(a). Then, the account of RSU_k is funded, and the smart contract SC_{RSU_k} is deployed in the blockchain network, obtaining its associated address $SCID_{RSU_k} = 0xCF180843dA8E6fe5Ae3F7baE982e62640d430C$. Fig. 8.3(b) shows SC_{RSU_k}'s functions. More details about the deployment are given in Fig. 8.3(c), including the gas cost of deploying SC_{RSU_k}.

2. In the simulation, RSU_k generates C_{GS} and publishes it using the **IssueGSK** function, invoking the published transaction's address $TxID$. At last, the RSU_k maps $TxID$ to PID_{V_i} using the **UpdateGSK** function.

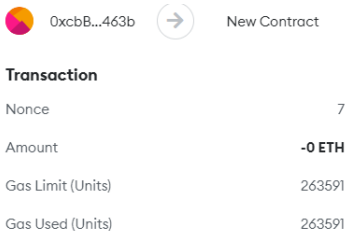
3. Switching to V_i's account and using $SCID_{RSU_k}$, V_i obtains $TxID$ by calling the **ViewGSK** function. At last, V_i retrieves Tx contents, C_{GS}, from the blockchain using $TxID$.



(a) Terminals' accounts.



(b) Smart contract's functions.



(c) The cost of deploying the smart contract.

Figure 8.3: Terminals' addresses and SC_{RSU_k}'s functions.

Table 8.2 shows the gas costs per Wei for SC_{RSU_k}'s functions, where Wei is the smallest unit in ETH, 1 ETH $= 10^{18}$ Wei. It is noteworthy to mention that the **Deployer** function is the most expensive in terms of gas costs. Since this process is only performed once, it is relatively inexpensive. As for the actual costs of **IssueGSK**, **UpdateGSK**, and **ViewGSK** functions, these are 0.0024, 0.0004, and 0.0003 ETH, respectively, which are acceptable for group key distribution.

8.3.2 Computation and communication comparisons

This subsection shows a detailed analysis of computation and communication costs.

Computation comparison

The exact estimates of time costs for different cryptographic operations in [147] are used, see Table 8.3. This evaluation is based on the MIRACL cryptographic library [126] using a quad-core i7 system with 16GB RAM. Based on that, the time required to verify a certain number of n received messages is evaluated for the schemes presented in [55], [56], [60], and the proposed,

Table 8.2: Gas costs associated with SC_{RSU_k}'s functions

Function	Gas used (Wei)	Actual cost (ETH)
Deployer	263591	0.002445
IssueGSK	69064	0.00049
UpdateGSK	46594	0.000308
ViewGSK	No fees	No fees

Table 8.3: The average execution time of different cryptographic operations in *msec* [147]

Operation	Definition	Time
T^{bp}	The bilinear pairing operation $e(.,.)$ in $\mathbb{G}_1$	13.44
T^{sm}_{bp}	The scalar multiplication operation in $\mathbb{G}_1$	2.521
T^{pa}_{bp}	The point addition operation in $\mathbb{G}_1$	0.018
T^{sm}_{ecc}	The scalar multiplication operation in $\mathbb{G}$	1.489
T^{pa}_{ecc}	The point addition operation in $\mathbb{G}$	0.008
T_h	The hashing operation (SHA-256)	0.003
T^{enc}_{AES}	Encryption operation using the AES algorithm	0.002
T^{dec}_{AES}	Decryption operation using the AES algorithm	0.001

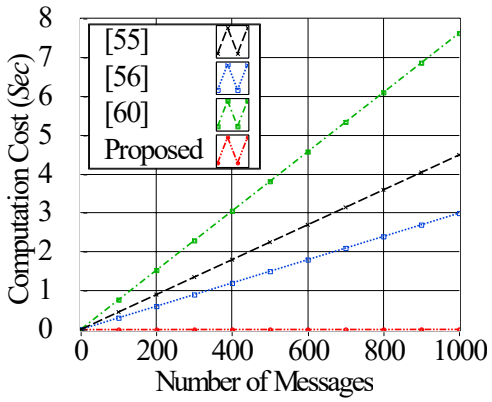
see Table 8.4. As can be seen, [55], [56], and [60] take $\approx (4.489n + 2.97)$, $(2.992n + 2.978)$, and $(7.62n + 5.042)$ *msec*, respectively to verify n messages. While the proposed scheme costs $\approx 0.004n$ *msec*. Fig. 8.4(a) shows the time taken to verify 1000 messages. It is evident from Fig. 8.4(a) that the proposed method exhibits the lowest computation cost slope in contrast to the traditional schemes outlined in [55], [56], and [60]. In comparison with [55], [56], and [60], the proposed scheme is more computationally efficient since [55], [56], and [60] are public-key cryptography-based, whereas the proposed scheme is a symmetric key cryptography-based.

Communication comparison

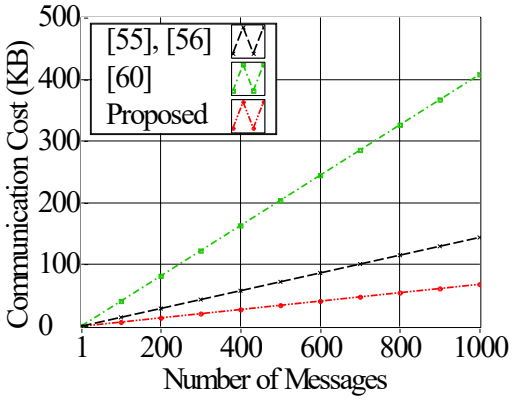
For the evaluation of communication costs, various parameter lengths are defined. For ECC's parameters of curve type $y^2 = x^3 + ax + b \; mod \; p$, the length of an element in $\mathbb{G}$ and Z^*_q are 40 and 20 bytes, respectively. For BP's parameters of curve type $y^2 = x^3 + x \; mod \; p$, the length of an element in $\mathbb{G}_1$ and Z^*_q are 128 and 20 bytes, respectively. While the length of the hashed value and timestamp are 32 and 4 bytes, respectively. According to [55], $\langle PID^1_i, PID^2_i, R_i, T_i, \sigma_m \rangle$ represents the signature, where $\{PID^1_i, PID^2_i, R_i\} \in \mathbb{G}$, $\sigma_m \in Z^*_q$, and T_i is the timestamp. Thus, the total signature size is $(3 \times 40 + 4 + 20) = 144$ bytes. Similarly, the transmission costs of [56] and [60] are calculated and presented in Table 8.4. According to the proposed scheme, $\langle T_i, PID_{V_i}, \sigma_i \rangle$ represents the signature, where PID_{V_i} and σ_i have the same length, 32 bytes each, and T_i is the timestamp. Thus, the total signature size is $(2 \times 32 + 4) = 68$ bytes. Fig. 8.4(b) shows the communication cost of transmitting 1000 messages. It is evident from Fig. 8.4(b) that the proposed method exhibits the lowest communication cost slope in contrast to the traditional schemes outlined in [55], [56], and [60]. Accordingly, it can be concluded that the proposed scheme saves high communication costs over those in [55], [56], and [60].

Table 8.4: Computation and communication comparisons

Scheme	Verification cost	Transmission cost
[55]	$(3n+2)T_{ecc}^{sm}+(2n-1)T_{ecc}^{pa}+$ $(2n)T_h \approx 4.489n+2.97$ *msec*	$144n$ *bytes*
[56]	$(2n+2)T_{ecc}^{sm}+(n)T_{ecc}^{pa}+(2n)T_h$ $\approx 2.992n+2.978$ *msec*	$144n$ *bytes*
[60]	$(3n+2)T_{bp}^{sm}+(3n)T_{bp}^{pa}+(n)T_h$ $\approx 7.62n+5.042$ *msec*	$408n$ *bytes*
Proposed	$(n)T_h+(n)T_{AES}^{dec}$ $\approx 0.004n$ *msec*	$68n$ *bytes*



(a) Computation comparison.



(b) Communication comparison.

Figure 8.4: Computation and communication costs of verifying and transmitting a number of n messages.

8.4 Summary

This chapter proposes a blockchain-based group key distribution technique that exploits the immutability of blockchain technology to distribute group session keys among group members via a smart contract. The smart contract's functions enable the group manager to distribute and update the group key in a secure manner without violating VANET security or privacy requirements. The scheme was tested for its resistance to active attacks. Additionally, the computation comparison demonstrated that the proposed scheme reduces the time needed to verify 1000 messages by 99% when compared to that of [55], [56], and [60]. While the transmission cost is reduced by 52.7% and 83.3% compared to that of [55, 56] and [60], respectively.

Chapter 9

Conclusions and Future Works

This chapter concludes the findings of this thesis and offers future insights aimed at enhancing the performance of authentication in VANETs.

9.1 Research questions and contributions

This thesis highlights the potential benefits of PHY-layer authentication in reducing the computational and communication overhead associated with cryptography-based authentication. The study presents a comprehensive classification of authentication techniques in VANETs, examining the strengths and weaknesses of each approach. It is evident that PHY-layer authentication cannot serve as a standalone method in VANETs but shows promise as a solution when combined with upper-layer methods. Furthermore, the research conducts a thorough evaluation and comparison of various authentication methods and algorithms, providing valuable insights into their effectiveness and performance. The classification facilitates a clear understanding of which PHY-layer authentication approaches are suitable for integration with upper-layer methods, taking into consideration factors such as computation availability, broadcasting rate, and channel conditions. It is crucial to note that not all PHY-layer authentication methods designed for indoor scenarios with low channel variations can effectively function in outdoor environments with high mobility and dense traffic. This analysis offers researchers in the field a valuable resource, inspiring innovation and the development of robust authentication schemes to meet the evolving demands of VANET applications in the future. In conclusion, this thesis addresses the research questions mentioned in Chapter 1 through the following exploration:

1. To address Q_1, Chapter 2 comprehensively reviews and classifies the current state-of-the-art of authentication in wireless communication. Specifically, existing crypto-based, PHY-layer-based, and cross-layer-based authentication methods are thoroughly examined and discussed, highlighting the limitations associated with each technique.

2. To address Q_2, the scheme presented in Chapter 3 employs the unique features of wireless

channels for re-authenticating the communicating terminal at the physical layer. In this scheme, a low-complexity cross-layer authentication scheme is introduced for VANET applications, employing the short-term channel reciprocity for re-authentication to address some of the performance limitations issues, particularly those related to the significant overheads of signature generation and verification. A lightweight pseudo-identity-based algorithm is proposed for verifying the legitimacy of the corresponding terminals at the first time slot, which increases the scheme's availability and mitigates the effect of flooding types of DoS attacks on the network. For re-authentication, a location-dependent-based PHY-layer re-authentication step is proposed for the identity re-verification process, which helps in detecting and preventing Sybil types of attacks.

3. To address Q_3: Chapter 4 introduces a critical consideration in existing PHY-layer authentication methods. These methods commonly assume a terminal distance of half a wavelength, ensuring strong decorrelation between authentic and wiretapped channel responses, a condition feasible in V2V communication but rendered impractical in V2I scenarios due to potential RSU compromises. To overcome this challenge, an innovative adaptive Chebyshev chaotic mapping-based secret key extraction technique is proposed in Chapter 4. This technique facilitates the generation of a PHY-layer signature, ensuring robust message authentication for both V2V and V2I communication.

4. To address Q_4, Chapter 5 presents an authentication scheme that utilises the RIS to improve the authentication performance in VANETs. The proposed scheme takes advantage of the unique properties of the RIS to enhance the channel characteristics and improve the reliability of authentication for NLoS scenarios. Chapter 6 introduces a key generation approach that utilises the RIS for secure message exchange in vehicular communication. The RIS improves the performance of the key extraction in scenarios with low SNR values and NLoS. Also, it enhances the network's security against DoS attacks. In this context, an efficient RIS configuration optimisation technique is proposed, which reinforces signals received from legitimate users while weakening signals from potential adversaries.

5. Additional contributions to address Q_3: Chapters 7, and 8 of this thesis present two effective authentication schemes that address some limitations of existing PHY-layer authentication and secret key extraction methods. Moreover, a novel technique for blockchain-based group key distribution is introduced. The following points provide a brief overview of the contributions made in each chapter pertaining to Q_3.

    - In PHY-layer secret key extraction, the imperfect channel reciprocity causes a number of mismatched bits in the extracted key, and existing reconciliation approaches suffer from high complexity and security flaws, posing a significant challenge. To address this issue, Chapter 7 proposes a smart contract-based blockchain technol-

ogy that enables a trusted third party to publish the correction sequence of the mismatched bits through a transaction. This transaction is used for reconciling the mismatched bits and enabling subsequent message authentication.

- Chapter 8 proposes a novel method for distributing group keys through the use of blockchain technology. This method enables the RSU to act as a group manager, distributing and updating group session keys among group members with minimal communication and computation costs via the use of smart contracts. Accordingly, a GS-based message authentication process is developed as a lightweight solution.

This research demonstrates the effectiveness of cross-layer authentication in VANET applications. The findings of this research serve as a foundation for future research endeavours aimed at enhancing the capabilities and performance of cross-layer authentication mechanisms. By addressing the identified challenges and exploring new techniques, researchers can strive to achieve superior performance in wireless applications and pave the way for more efficient and secure authentication protocols. Continued efforts in this area will contribute to the development of robust and high-performing cross-layer authentication solutions for diverse wireless communication scenarios.

9.2 Limitations and challenges

Identifying and acknowledging the limitations and challenges of this thesis is an important aspect of providing a comprehensive assessment. Here are some limitations and challenges that need to be considered in future works.

1. *Limited real-world validation*: Conducting real-world experimental analyses of PHY-layer authentication, particularly at diverse speeds of moving vehicles, introduces a complex and significant challenge. The varying vehicle speeds can lead to dynamic changes in channel conditions, making it challenging to precisely measure and assess the performance of authentication methods under different vehicle speed scenarios, especially when dealing with fixed SNR values. This challenge underscores the need for comprehensive investigations and innovative methodologies to effectively account for the dynamic nature of vehicular environments.

2. *Adaptable RIS configuration*: Optimising the RIS configuration to accommodate dynamic terminals remains an active area of investigation within the research community, particularly in the context of RIS-assisted authentication and key extraction (Chapters 5 and 6). The challenge lies in the need for the RIS to continually adapt and update its configuration for forthcoming time slots and changing vehicle locations, an aspect that needs further exploration.

3. *RIS synchronisation*: The synchronisation challenge constitutes another critical aspect in the context of RIS-assisted authentication and key extraction (Chapters 5 and 6). Achieving precise synchronisation between RIS configurations and the dynamic terminals, especially in rapidly changing vehicular environments, remains an active area of investigation and presents a significant research endeavour within the scientific community.

9.3 Future works

There is substantial potential for continued research and development in this domain. Several promising avenues for future investigation encompass the exploration of novel designs for adaptable cross-layer authentication schemes tailored to various applications of VANET. The following subsections provide a concise introduction to these avenues.

9.3.1 Machine learning-based adaptive cross-layer authentication

In the context of PHY-layer re-authentication, it is evident that certain methods, such as tag-based approaches, necessitate high SNR values to achieve satisfactory performance. Conversely, other methods like keyed-based PHY-layer authentication methods exhibit acceptable performance even in lower SNR conditions, whereas crypto-based methods prove advantageous for re-authentication under poor SNR conditions. Against this backdrop, employing a machine learning model to classify and optimise the most suitable re-authentication method during the initial time slot, based on the estimated SNR value, becomes a viable approach.

9.3.2 Decentralised smart contract-based blockchain for efficient secret key reconciliation

Smart contracts have demonstrated their effectiveness in reconciling discrepancies arising from the extraction of key bits at the physical layer. By leveraging a smart contract, a trusted third party can play the role of a referee in facilitating communication between two terminals and publishing the correction sequence. To further enhance this solution, the design of a decentralised smart contract incorporating public key cryptography can significantly reduce the overhead associated with relying on a trusted third party. In this scenario, the smart contract serves as the referee itself, eliminating the requirement for an external entity and introducing more efficient reconciliation solutions.

9.3.3 Federated learning for efficient PHY-layer re-authentication

Federated learning is a decentralised machine learning approach where multiple devices (e.g., IoT devices, smartphones) collaborate to train a global machine learning model without sharing

raw data with a central server [165, 166]. In the context of physical layer authentication performance, federated learning can be applied to improve the authentication process in the following ways:

1. *Collaborative model training*: Devices in a wireless network can collectively participate in training a shared physical layer authentication model. Each device can use its locally collected data to update the model parameters without transmitting the raw data to a central authority. The global model can then be improved by aggregating the locally updated models from different devices.

2. *Privacy preservation*: Traditional centralised authentication methods often require devices to transmit sensitive information to a central server, raising privacy concerns. Federated learning helps to address this issue by allowing devices to keep their data locally and only share model updates. This way, the privacy of the individual devices' data is preserved.

3. *Adaptation to diverse environments*: Different devices in a wireless network may experience varying communication conditions due to channel fading, interference, and mobility. By incorporating federated learning, devices can adaptively learn and update the physical layer authentication model based on their local experiences, leading to improved performance in diverse environments.

4. *Real-time learning*: Federated learning can facilitate continuous learning and adaptation of the authentication model in real-time. As devices collect new data and experience changing channel conditions, they can continuously update the global model without requiring centralised retraining.

5. *Robustness to attacks*: In the presence of adversarial attacks that attempt to spoof or compromise the authentication process, federated learning can provide robustness by aggregating the knowledge from multiple devices, making it harder for attackers to target a single central authority.

However, it is essential to consider challenges such as communication overhead, synchronisation, and model aggregation techniques to ensure efficient and effective federated learning for physical layer authentication. Additionally, security measures should be taken to protect the federated learning process itself, ensuring that adversaries cannot manipulate the model updates or extract sensitive information during the collaboration.

9.3.4 Efficient handover authentication methods

The dynamic nature of the wireless channel, considering both its spatial and temporal variations, offers an opportunity to develop a robust handover mechanism. Specifically, in scenarios where a vehicle V_i operates within the overlapping coverage areas of consecutive Roadside Units (RSUs) denoted as RSU_j and RSU_{j+1}, the channel characteristics $Ch_{V_i \rightarrow RSU_j}(t)$ exhibit a unique difference and high degree of decorrelation with $Ch_{V_i \rightarrow RSU_{j+1}}(t)$ at a given instance t. This decorrelation manifests as a potential means to establish a mapping between ($Ch_{V_i \rightarrow RSU_j}(t)$, $Ch_{V_i \rightarrow RSU_{j+1}}(t)$) across diverse positions. This mapping facilitates the establishment of delegation of trustworthiness among different network nodes, paving the way for a lightweight handover authentication method. Such a method capitalises on the inherent correlation in channel characteristics, enabling reliable and efficient handover procedures.

In conclusion, the horizon of possibilities within this domain stretches wide, offering a boundless realm of untapped potential. The roadmap to innovation is illuminated by the promise of novel designs in adaptable cross-layer authentication schemes, each meticulously tailored to the diverse applications of VANET. As we embark on this journey, let us not only seek to secure our networks but also to empower them, forging a path toward a safer, smarter, and more interconnected future. The keys to progress lie within our collective resolve and the limitless creativity that awaits our exploration. The future is ours to shape, and the time to embark on this transformative voyage is now.

Appendix A

Derivation of Equation (3.16)

Proof. Considering an adversary $\mathscr{A}$ who is trying to forge σ_{V_2} of the vehicle V_2 by the construction of an algorithm C to solve the defined problems with a probability of success $\varepsilon_{\text{sig.Gen.}}$. Algorithm C initially holds two empty tables $T_{H_1}[.]$ and $T_{HMAC}[.]$ to simulate random oracles $H_1(.)$ and $HMAC_{key}(.)$, then answers $\mathscr{A}$'s oracle queries as follows:

- *ID queries*: For a query $\left(TID_{V_2}, PID_2^1, A_2\right)$, C holds $\left\langle A_1, \left(a_2, \alpha_2 \in Z_q^*\right)\right\rangle$, randomly selects r_{V_2} and $\beta \in Z_q^*$, then computes $A_2 = a_2 \cdot P, PID_2^1 = \alpha_2 \cdot r_{V_2} \cdot P, PK_{V_2,TA} = r_{V_2} \cdot \beta \cdot P, \rho = \alpha_2 \cdot PK_{V_2,TA}$, and $TID_{V_2} = r_{V_2} \cdot A_1$. If $T_{H_1}[\rho]$ is defined, then C halts, returns $\perp$, and sets $false \leftarrow true$, otherwise, it sets $T_{H_1}(\rho) \leftarrow H : \{0,1\}^{N_1}$, and returns $\left(TID_{V_2}, PID_2^1, A_2\right)$ to $\mathscr{A}$ under (r_{V_2}, β).

- *Sign queries*: For a query $\left(PID_2^2, \sigma_{V_2}, T_2\right)$, C selects $RID_{V_2} \in \{0,1\}^{N_2}$ at timestamp T_2, obtains H from ID queries, then computes $SK_{V_{1-2}} = a_2 \cdot A_1$ and $PID_2^2 = RID_{V_2} \oplus H$. If $T_{HMAC}[TID_{V_2} \| PID_{V_2} \| T_2]$ is defined, C halts, returns $\perp$, and sets $false \leftarrow true$. Otherwise, it sets $HMAC_{SK_{V_{1-2}}}(TID_{V_2} \| PID_{V_2} \| T_2) \leftarrow \sigma_{V_2} : \{0,1\}^{N_2}$, and returns $\left(PID_2^2, \sigma_{V_2}, T_2\right)$ to $\mathscr{A}$ under RID_{V_2}.

Finally, it is assumed that $\mathscr{A}$ successfully generated a forged signature $\langle TID_{V_2}, PID_{V_2}, \sigma_{V_2}, A_2, T_2\rangle$ under $\langle r_{V_2}, \beta, RID_{V_2}\rangle$ based on q_{ID} and q_s queries for ID and Sign oracles with probability $\varepsilon_{\text{Sig.Gen}} = \Pr[E_1]\Pr[E_2 \mid E_1]$, in which E_1 and E_2 are defined as:

- *Event* E_1 : Algorithm C did not abort due to signature simulation.

- *Event* E_2 : Non-trivial forgery is successfully returned by adversary $\mathscr{A}$.

The probability $\Pr[\neg false]$ must be computed, in which false indicates that the algorithm C aborts as a result of ID and Sign queries. The probability is evaluated according to the following claims.

Claim 1. $\Pr[E_1] = \Pr[\neg false] \geq 1 - \frac{q_{ID}^2 q_S^2}{|N_1||N_2|}$

Proof. The probability $\Pr[false]$ can be evaluated by estimating the multiplication of the following probabilities.

- *Scenario 1.* $false \leftarrow true$ is obtained in the ID queries if H is occurred by chance in a previous query to the oracle $H_1(.)$ under (r_{V_2}, β). There are at most q_{ID} queries in table $T_{H_1}[.]$, the probability for a single ID query is at most $\frac{q_{ID}}{|N_1|}$, and the probability for q_{ID} queries is $\frac{q_{ID}^2}{|N_1|}$.
- *Scenario 2.* $false \leftarrow true$ is obtained in the Sign queries if σ_{V_2} is occurred by chance in a previous query to the oracle $HMAC_{SK_{V_{1-2}}}(.)$ under $SK_{V_{1-2}} \in \mathbb{G}$ and RID_{V_2}. There are at most q_s queries in table $T_{HMAC}[.]$, the probability for a single Sign query is at most $\frac{q_s}{|N_2|}$, and the probability for q_s queries is $\frac{q_s^2}{|N_2|}$.

Claim 2. $\Pr[E_2 \mid E_1] \geq \varepsilon$

Proof. $\Pr[E_2 \mid E_1]$ is the probability that $\mathscr{A}$ generates a valid forgery, and C does not halt due to $\mathscr{A}$'s ID and Sign queries which means that all responses to these queries are valid. Therefore $\mathscr{A}$ will produce a valid forgery with probability ε.

At last, the probability that $\mathscr{A}$ successfully impersonates V_2 by computing a non-trivial forgery under $\langle r_{V_2}, \beta, RID_{V_2} \rangle$ is at least

$$\varepsilon_{\text{Sig.Gen}} = \varepsilon \left(1 - \frac{q_{ID}^2 q_S^2}{|N_1|\,|N_2|}\right)$$

Appendix B

AVISPA Simulation Codes

Code 1: HLPSL code for the role of the vehicle V_1, played by V_1

```
role role_V1 (V1,V2,RSU,RTA:agent,KV1,KV2,KTA,KRSU,KRTA:public_key,
              KV1Rta:symmetric_key,SND,RCV:channel(dy))
played_by V1
def=
        local
                State:nat,TR,T1,T2,T4,T6,PPV1,PPV2,CS,M:text,
                K1,K2:symmetric_key
        init
                State:=0
        transition
                1. State=0 /\ RCV(start) = | > State':=1 /\ T1':=new() /\
                   SND(PPV1.T1'.KV1.TR.{KV1.TR}_inv(KTA).{PPV1.T1'
                   .KV1.TR.{KV1.TR}_inv(KTA)}_inv(KV1))
                   %% V1 hopes that PPV1 will be authenticated by V2
                   /\ witness(V1,V2,auth_1,PPV1)

                2. State=1 /\ RCV(PPV2.T2'.KV2.TR.{KV2.TR}_inv(KTA).
                   {PPV2.T2'.KV2.TR.{KV2.TR}_inv(KTA)}_inv(KV2))
                   = | > State':=2 /\ K1':=new() /\ T4':=new() /\
                   SND({K1'}_KV1Rta.KV2.TR.{KV2.TR}_inv(KTA).KV1.
                   TR.{KV1.TR}_inv(KTA).T4'.{{K1'}_KV1Rta.KV2.TR.
                   {KV2.TR}_inv(KTA).KV1.TR.{KV1.TR}_inv(KTA).T4'}
                   _inv(KV1))
                   %% V1 verifies the received PPV2 from V2
                   /\ request(V1,V2,auth_2,PPV2)
                   %% V1 believes in the secrecy of K1' transmitted to the RSU
                   /\ secret(K1',sec_1,{V1,RSU})
                   %% V1 hopes that {K1'}_KV1Rta will be authenticated
                      by the RSU
                   /\ witness(V1,RSU,auth_4,{K1'}_KV1Rta)

                3. State=2 /\ RCV(KV1.KV2.CS'.{KV1.KV2.CS'}
                   _inv(KRTA)) = | > State':=3 /\ T6':=new() /\ K2':=
                   xor(Cs',K1) /\ T6':=new() /\ SND(M.T6'.KV1.KV2.CS.
                   {KV1.KV2.CS}_inv(KRTA).{M.T6'.KV1.KV2.CS.{KV1.
                   KV2.CS}_inv(KRTA)}_K2')
                   %% V1 verifies the received CS' from the RTA
                   /\ request(V1,RTA,auth_6,CS')
                   %% V1 hopes that the message M will be authenticated
                      by V2
                   /\ witness(V1,V2,auth_7,M)
end role
```

Code 2: HLPSL code for the role of the vehicle V_2, played by V_2

```
role role_V2 (V2,V1,RSU,RTA:agent,KV1,KV2,KTA,KRSU,KRTA:public_key,
              KV2Rta:symmetric_key,SND,RCV:channel(dy))
played_by V2
def=
        local
                State:nat,TR,T1,T2,T3,T6,PPV1,PPV2,CS,M:text,
                K1,K2:symmetric_key
        init
                State:=0
        transition
                1. State=0 /\ RCV(PPV1.T1'.KV1.TR.{KV1.TR}_inv(KTA).
                   {PPV1.T1'.KV1.TR.{KV1.TR}_inv(KTA)}_inv(KV1))
                    = | > State':=1 /\ T2':=new() /\ SND(PPV2.T2'.KV2.
                   TR.{KV2.TR}_inv(KTA).{PPV2.T2'.KV2.TR.{KV2.TR}
                   _inv(KTA)}_inv(KV2)) /\ K2':=new() /\ SND({K2'}
                   _KV2Rta.KV1.TR.{KV1.TR}_inv(KTA).KV2.TR.{KV2.TR}
                   _inv(KTA).T3'.{{K2'}_KV2Rta.KV1.TR.{KV1.TR}
                   _inv(KTA).KV2.TR.{KV2.TR}_inv(KTA).T3'}_inv(KV2))
                   %% V2 verifies the received PPV1 from V1
                   /\ request(V2,V1,auth_1,PPV1)
                   %% V2 hopes that PPV2 will be authenticated by V1
                   /\ witness(V2,V1,auth_2,PPV2)
                   %% V2 believes in the secrecy of K2' transmitted to the RSU
                   /\ secret(K2',sec_2.{V2,RSU})
                   %% V2 hopes that {K2'}_KV2Rta will be authenticated
                        by the RSU
                   /\ witness(V2,RSU,auth_3,{K2'}_KV2Rta)

                2. State=1 /\ RCV(M.T6.KV1.KV2.CS'.{KV1.KV2.CS'}
                   _inv(KRTA).{M.T6.KV1.KV2.CS'.{KV1.KV2.CS'}
                   _inv(KRTA)}_K2) = | > State':=2
                   %% V2 verifies the received message M from V1
                   /\ request(V2,V1,auth_7,M)
end role
```

Code 3: HLPSL code for the role of the RSU, played by RSU

```
role role_RSU (RSU,V1,V2,RTA:agent,KV1,KV2,KTA,KRSU,KRTA:
               public_key,SND,RCV:channel(dy))
played_by RSU
def=
        local
                State:nat,TR,T3,T4,T5:text,
                K1,K2,KV1Rta,KV2Rta:symmetric_key
        init
                State:=0
        transition
                1. State=0 /\ RCV({K2'}_KV2Rta'.KV1.TR.{KV1.TR}
                   _inv(KTA).KV2.TR.{KV2.TR}_inv(KTA).T3'.{{K2'}
                   _KV2Rta'.KV1.TR.{KV1.TR}_inv(KTA).KV2.TR.{KV2
                   .TR}_inv(KTA).T3'}_inv(KV2)) = | > State':=1
                   %% RSU verifies the received {K2'}_KV2Rta' from V2
                   /\ request(RSU,V2,auth_3,{K2'}_KV2Rta)

                2. State=1 /\ RCV({K1'}_KV1Rta'.KV2.TR.{KV2.TR}
                   _inv(KTA).KV1.TR.{KV1.TR}_inv(KTA).T4'.{{K1'}
                   _KV1Rta'.KV2.TR.{KV2.TR}_inv(KTA).KV1.TR.
                   {KV1.TR}_inv(KTA).T4'}_inv(KV1)) = | > State':=2 /\
                    T5':=new() /\ SND({K1'}_KV1Rta'.{K2}_KV2Rta.KV1.
                   KV2.T5'.KRSU.TR.{KRSU.TR}_inv(KTA).{{K1'}
                   _KV1Rta'.{K2}_KV2Rta.KV1.KV2.T5'.KRSU.TR.{KRSU.
                   TR}_inv(KTA)}_inv(KRSU))
                   %% RSU verifies the received {K1'}_KV1Rta' from V1
                   /\ request(RSU,V1,auth_4,{K1'}_KV1Rta')
                   %% RSU beleives in the secrecy of K1' transmitted
                        to the RTA
                   /\ secret(K1',sec_3,{RSU,RTA})
                   %% RSU beleives in the secrecy of K2 transmitted to the RTA
                   /\ secret(K2,sec_4,{RSU,RTA})
                   %% RSU hopes that {K1'}_KV1Rta' and {K2}_KV2Rta
                        will be authenticated by RTA
                   /\ witness(RSU,RTA,auth_5,{K1'}_KV1Rta'.{K2}_KV2Rta)
end role
```

Code 4: HLPSL code for the role of the RTA, played by RTA

```
role role_RTA (RTA,V1,V2,RSU:agent,KV1,KV2,KTA,KRSU,KRTA:public_key,
               KV1Rta,KV2Rta:symmetric_key,SND,RCV:channel(dy))
played_by RTA
def=
        local
              State:nat,TR,T5,CS:text,
              K1,K2:symmetric_key
        init
              State:=0
        transition
              1. State=0 /\ RCV({K1'}_KV1Rta'.{K2}_KV2Rta.KV1.KV2.
                 T5'.KRSU.TR.{KRSU.TR}_inv(KTA).{{K1'}_KV1Rta.
                 {K2'}_KV2Rta.KV1.KV2.T5'.KRSU.TR.{KRSU.TR}
                 _inv(KTA)}_inv(KRSU)) = | > State':=1 /\ CS':=xor(K1',
                 K2') /\ SND(KV1.KV2.CS'.{KV1.KV2.CS'}_inv(KRTA))
                 %% RTA verifies the received {K1'}_KV1Rta' and {K2'}
                       _KV2Rta from the RSU
                 /\ request(RTA,RSU,auth_5,{K1'}_KV1Rta'.{K2'}_KV2Rta)
                 %% RTA hopes that CS' will be authenticated by V1
                 /\ witness(RTA,V1,auth_6,CS')
end role
```

Code 5: HLPSL code for the roles of session and environment and protocol goals

```
role session (V1,V2,RSU,RTA:agent,KV1,KV2,KTA,KRSU,KRTA:public_key,
              KV1Rta,KV2Rta:symmetric_key,SND,RCV:channel(dy))
def=
        local
              SND1,RCV1,SND2,RCV2,SND3,RCV3,SND4,RCV4:channel(dy)
        composition
              role_V1 (V1,V2,RSU,RTA,KV1,KV2,KTA,KRSU,KRTA,
                       KV1Rta,SND1,RCV1) /\
              role_V2 (V2,V1,RSU,RTA,KV1,KV2,KTA,KRSU,KRTA,
                       KV2Rta,SND2,RCV2) /\
              role_RSU (RSU,V1,V2,,RTA,KV1,KV2,KTA,KRSU,KRTA,
                       SND3,RCV3) /\
              role_RTA (RTA,V1,V2,RSU,KV1,KV2,KTA,KRSU,KRTA,
                       KV1Rta,KV2Rta,SND4,RCV4)
end role
role environment ()
def=
        const
              kv1,kv2,kta,krsu,krta:public_key,
              k1,k2,kv1rta,kv2rta:symmetric_key,
              v1,v2,rsu,rta:agent,
              auth_1,auth_2,auth_3,auth_4,auth_5,auth_6,auth_7,sec_1,sec_2,
              sec_3,sec_4:protocol_id
              intruder_knowledge={v1,v2,rsu,rta,kv1,kv2,kta,krsu,krta}
        composition
              session(v1,v2,rsu,rta,kv1,kv2,kta,krsu,krta,kv1rta,kv2rta)
end role
goal
              secrecy_of sec_1,sec_2,sec_3,sec_4
              authentication_on auth_1,auth_2,auth_3,auth_4,auth_5,auth_6,
              auth_7
end goal
environment()
```

Bibliography

[1] World Health Organization. (2020). *2nd global safety report on road safety 2011-2020*. Available at: https://www.who.int/groups/united-nations-road-safety-collaboration/decade-of-action-for-road-safety-2011-2020.

[2] European Commission. (2020, June). European road accident database. Available at: https://transport.ec.europa.eu/index_en.

[3] UN Economic Commission for Europe. (2020, July). A foundational safety system concept to make roads safer in the decade 2021-2030. Available at: https://unece.org/transport/publications/safety-system-concept-make-roads-safer.

[4] Zeadally, S., Hunt, R., Chen, Y. S., Irwin, A., & Hassan, A. (2012). Vehicular ad-hoc networks (VANETS): status, results, and challenges. *telecommunication systems*, 50, 217–241. doi: 10.1007/s11235-010-9400-5.

[5] Al-Shareeda, M. A., Anbar, M., Hasbullah, I., & Manickam, S. (2021). Survey of authentication and privacy schemes in vehicular ad hoc networks. *IEEE Sensors Journal*, 21(2), 2422-2433. doi: 10.1109/JSEN.2020.3021731.

[6] Grafling, S., Mahonen, P., & Riihijarvi, J. (2010). Performance evaluation of IEEE 1609 wave and IEEE 802.11p for vehicular communications. In ICUFN Conference Proceedings (pp. 344-348). doi: 10.1109/ICUFN.2010.5547184.

[7] Abu Talib, M., Abbas, S., Nasir, Q., & Mowakeh, M. F. (2018). Systematic literature review on internet-of-vehicles communication security. *International Journal of Distributed Sensor Networks*, 14. doi: 10.1177/1550147718815054.

[8] Raya, M., & Hubaux, J. P. (2005). The security of vehicular ad hoc networks. In Proceedings of the 3rd ACM Workshop Security Ad Hoc Sensor Networks (pp. 11-21). doi: 10.1145/1102219.1102223.

[9] Shamir, A. (1985). Identity-based cryptosystems and signature schemes. In G.R. Blakley & D. Chaum (Eds.), Advances in Cryptology. CRYPTO 1984 (pp. 47-53). Lecture Notes in Computer Science. Springer, Berlin, Heidelberg. doi: 10.1007/3-540-39568-7_5.

[10] Chaum, D., & Heyst, E. V. (1991). Group signatures. In Proceedings of the Workshop on the Theory and Application of Crypto. Tech. (pp. 257-265). Berlin, Heidelberg: Springer. doi: 10.1007/3-540-46416-6_22.

[11] Xiao, L., Greenstein, L. J., Mandayam, N. B., & Trappe, W. (2008). Using the physical layer for wireless authentication in time-variant channels. *IEEE Transactions on Wireless Communications*, 7(7), 2728-2739. doi: 10.1109/TWC.2008.070194.

[12] Liu, J., & Wang, X. (2016). Physical layer authentication enhancement using two-dimensional channel quantization. *IEEE Transactions on Wireless Communications*, 15(6), doi: 10.1109/TWC.2016.2535442.

[13] Mukherjee, A., Fakoorian, S. A. A., Huang, J., & Swindlehurst, A. L. (2014). Principles of physical layer security in multiuser wireless networks: A survey. *IEEE Communications Surveys & Tutorials*, 16(3), 1550-1573. doi: 10.1109/SURV.2014.012314.00178.

[14] Fu, B., Xiao, Y., Deng, H., & Zeng, H. (2014). A survey of cross-layer designs in wireless networks. *IEEE Communications Surveys & Tutorials*, 16(1), 110-126. doi: 10.1109/SURV.2013.081313.00231.

[15] Haenel, A., Haddad, Y., Laurent, M., & Zhang, Z. (2021). Practical cross-layer radio frequency-based authentication scheme for internet of things. *Sensors*, 21(12). doi: 10.3390/s21124034.

[16] Wen, H., Zhang, J., Liao, R., Tang, J., & Pan, F. (2019). Cross-layer authentication method based on radio frequency fingerprint [Patent number US 10251058 B2, United States Patent].

[17] Shawky, M. A., Shah, S. T., Usman, M., Abbasi, Q. H., Imran, Ansari, S., & Taha, A. How Secure Are Our Roads? An In-Depth Review of Authentication in Vehicular Communications. *Under Review*.

[18] Shawky, M. A., Bottarelli, M., Epiphaniou, G., & Karadimas, P. (2023). An efficient cross-layer authentication scheme for secure communication in vehicular ad-hoc networks. *IEEE Transactions on Vehicular Technology*, 72. doi: 10.1109/TVT.2023.3244077.

[19] Shawky, M. A., Abbasi, Q. H., Imran, M. A., Ansari, S., & Taha, A. (2022). Cross-layer authentication based on physical-layer signatures for secure vehicular communication. In Proceedings of the IEEE Intelligent Vehicles Symposium (IV), Aachen, Germany. doi: 10.1109/IV51971.2022.9827444.

[20] Shawky, M. A., Usman, M., Imran, M. A., Abbasi, Q. H., Ansari, S., & Taha, A. (2023). Adaptive chaotic map-based key extraction for efficient cross-layer authentication in VANETs. *Vehicular Communications*, 39. doi: 10.1016/j.vehcom.2022.100547.

[21] Shawky, M. A., Usman, M., Imran, M. A., Abbasi, Q. H., Ansari, S., & Taha, A. (2022). Adaptive and efficient key extraction for fast and slow fading channels in V2V communications. IEEE 96th Vehicular Technology Conference (VTC2022-Fall), London, United Kingdom. doi: 10.1109/VTC2022-Fall57202.2022.10012884.

[22] Shawky, M. A., Shah, S. T., Mollel, M. S., Kazim, J. U., Abbasi, Q. H., Imran, M. A., Ansari, S., & Taha, A. (Under Review). Reconfigurable intelligent surface-assisted cross-layer authentication for secure and efficient vehicular communications. *IEEE Transactions on Wireless Communications*.

[23] Shawky, M. A., Shah, S. T., Abbasi, Q. H., Imran, M. A., Hasan, S. F., Ansari, S., & Taha, A. (2023). RIS enabled secret key generation for secured vehicular communication in the presence of denial-of-service attacks. *MDPI Sensors*, 23(8). doi. 10.3390/s23084104.

[24] Shawky, M. A., Usman, M., Flynn, D., Imran, M. A., Abbasi, Q. H., Ansari, S., & Taha, A. (2023, May). Blockchain-based secret key extraction for efficient and secure authentication in VANETs. *Journal of Information Security and Applications*, 74, 2214-2126. doi: 10.1016/j.jisa.2023.103476.

[25] Shawky, M. A., Jabbar, A., Usman, M., Imran, M. A., Abbasi, Q. H., Ansari, S., & Taha, A. (2023). Efficient blockchain-based group key distribution for secure authentication in VANET. *IEEE Networking Letters*, 5(1), 64-68. doi. 10.1109/LNET.2022.3056545.

[26] Yang, C., Qin, B., Zhou, X., Sun, Y., He, S., & Wu, Q. (2015). Privacy-preserving traffic monitoring in vehicular ad hoc networks. IEEE 29th International Conference on Advanced Information Networking and Applications Workshops (pp. 22-24). doi: 10.1109/WAINA.2015.31.

[27] Ben Jaballah, W., Conti, M., & Lal, C. (2020). Security and design requirements for software-defined VANETs. *Computer Networks*, 169. doi: 10.1016/j.comnet.2020.107099.

[28] Bariah, L., Shehada, D., Salahat, E., & Yeun, C. Y. (2015). Recent advances in VANET security: A survey. IEEE 82nd Vehicular Technology Conference (VTC2015-Fall) (pp. 1-7). Boston, MA, USA. doi: 10.1109/VTCFall.2015.7391111.

[29] Liu, J., Li, J., Zhang, L., Dai, F., Zhang, Y., Meng, X., & Shen, J. (2018). Secure intelligent traffic light control using fog computing. *Future Generation Computer Systems*, 78, 817-824. doi: 10.1016/j.future.2017.02.017.

[30] Poornachander, V. (2016). Security issues on cryptography and network security. *International Journal of Computer Science and Information Technologies*, 7, 1648-1654.

[31] Bai, L., Zhu, L., Liu, J., Choi, J., & Zhang, W. (2020). Physical layer authentication in wireless communication networks: A survey. *Journal of Communications and Information Networks*, 5, 237-264. doi: 10.23919/JCIN.2020.9200889.

[32] Manvi, S. S., & Tangade, S. (2017). A survey on authentication schemes in VANETs for secured communication. *Vehicular Communications*, *9*. doi: 10.1016/j.vehcom.2017.02.001.

[33] Koblitz, N. (1987). Elliptic curve cryptosystems. *Mathematics of Computation*, 48 (177), 203–209. doi:10.2307/2007884.

[34] Koblitz, N., & Menezes, A. (2005). Pairing-based cryptography at high security levels. Cryptography and Coding. Lecture Notes in Computer Science, 3796, 13–36. doi:10.1007/11586821_2.

[35] Alshaikhli, I. F., & AlAhmad, M. A. (2015), Cryptographic Hash Function, Handbook of Research on Threat Detection and Countermeasures in Network Security, *IGI Global*, 80–94. doi:10.4018/978-1-4666-6583-5.ch006.

[36] Raya, M., & Hubaux, J. P. (2005). The security of vehicular ad hoc networks. The 3rd ACM Workshop Security Ad Hoc Sensor Networks (pp. 11-21). doi: 10.1145/1102219.1102223.

[37] Oulhaci, T., Omar, M., Harzine, F., & Harfi, I. (2016). Secure and distributed certification system architecture for safety message authentication in VANET. *Telecommunication Systems*, 64. doi: 10.1007/s11235-016-0200-4.

[38] Wang, S., & Yao, N. (2017). LIAP: A local identity based anonymous message authentication protocol in VANETs. *Computer Communications*, 112, 154-164. doi: 10.1016/j.comcom.2017.09.005.

[39] Wang, S., Mao, K., Zhan, F., & Liu, D. (2020). Hybrid conditional privacy preserving authentication scheme for VANETs. *Peer-to-Peer Networking and Applications*. doi: 10.1007/s12083-020-00916-3.

[40] Lu, Z., Liu, W., Wang, Q., Qu, G., & Liu, Z. (2018, August). A privacy-preserving trust model based on blockchain for VANETs. *IEEE Access*, 6, 45655-45664. doi: 10.1109/ACCESS.2018.2864189.

[41] Lu, Z., Wang, Q., Qu, G., Zhang, H., & Liu, Z. (2019, December). A blockchain-based privacy-preserving authentication scheme for VANETs. *IEEE Transactions on Very Large Scale Integration Systems*, 27(12), 2792-2801. doi: 10.1109/TVLSI.2019.2929420.

[42] Lin, C., He, D., Huang, X., Kumar, N., & Choo, K. R. (2021, December). BCPPA: A blockchain-based conditional privacy-preserving authentication protocol for vehicular

ad hoc networks. *IEEE Transactions on Intelligent Transportation Systems*, 22(12). doi: 10.1109/TITS.2020.3002096.

[43] Liu, Y., Wang, L., & Chen, H. (2014). Message authentication using proxy vehicles in vehicular ad hoc networks. *IEEE Transactions on Vehicular Technology*, 64(8), 3697-3710. doi: 10.1109/TVT.2014.2358633.

[44] Asaar, M. R., Salmasizadeh, M., Susilo, W., & Majidi, A. (2018). A secure and efficient authentication technique for vehicular ad-hoc networks. *IEEE Transactions on Vehicular Technology*, 67(6), 5409-5423. doi: 10.1109/TVT.2018.2822768.

[45] Bayat, M., Pournaghi, M., Rahimi, M., & Barmshoory, M. (2019). NERA: A new and efficient RSU based authentication scheme for VANETs. *Wireless Networks*, 26, 3083-3098. doi: 10.1007/s11276-019-02039-x.

[46] Al-shareeda, M. A., Anbar, M., Manickam, S., & Hasbullah, I. H. (2020). An efficient identity-based conditional privacy-preserving authentication scheme for secure communication in a vehicular ad hoc network. *Symmetry*, 12(10), 1687-1712. doi: 10.3390/sym12101687.

[47] Lo, N. W., & Tsai, J. L. (2016). An efficient conditional privacy-preserving authentication scheme for vehicular sensor networks without pairings. *IEEE Transactions on Intelligent Transportation Systems*, 17(5), 1319-1328. doi: 10.1109/TITS.2015.2502322.

[48] Wei, Z., Li, J., Wang, X., & Gao, C. (2019). A lightweight privacy-preserving protocol for VANETs based on secure outsourcing computing. *IEEE Access*, 7, 62785-62793. doi: 10.1109/ACCESS.2019.2915794.

[49] Zhang, G., Liao, Y., Fan, Y., & Liang, Y. (2020). Security analysis of an identity-based signature from factorization problem. *IEEE Access*, 8, 23277-23283. doi: 10.1109/ACCESS.2020.2964040.

[50] Cui, J., Wei, L., Zhang, J., Xu, Y., & Zhong, H. (2019). An efficient message-authentication scheme based on edge computing for vehicular ad hoc networks. *IEEE Transactions on Intelligent Transportation Systems*, 20(5), 1621-1632. doi: 10.1109/TITS.2018.2827460.

[51] Limbasiya, T., & Das, D. (2020). Lightweight secure message broadcasting protocol for vehicle-to-vehicle communication. *IEEE Systems*, 14(1), 520-529. doi: 10.1109/JSYST.2019.2932807.

[52] Lyu, C., Gu, D., Zeng, Y., & Mohapatra, P. (2016). PBA: Prediction-based authentication for vehicle-to-vehicle communications. *IEEE Transactions on Dependable and Secure Computing*, 13(1), 71-83. doi: 10.1109/TDSC.2015.2399297.

[53] Zhong, H., Han, S., Cui, J., Zhang, J., & Xu, Y. (2019). Privacy-preserving authentication scheme with full aggregation in VANET. *Information Sciences*, 476, 211–221. doi: 10.1016/j.ins.2018.10.021.

[54] Cui, J., Chen, J., Zhong, H., et al. (2022). Reliable and efficient content sharing for 5G-enabled vehicular networks. *IEEE Transactions on Intelligent Transportation Systems*, 23(2), 1247-1259. doi: 10.1109/TITS.2020.3023797.

[55] He, D., Zeadally, S., Xu, B., & Huang, X. (2015). An efficient identity-based conditional privacy-preserving authentication scheme for vehicular ad hoc networks. *IEEE Transactions on Information Forensics and Security*, 10(12), 2681-2691. doi: 10.1109/TIFS.2015.2473820.

[56] Li, J., Choo, K. R., Zhang, W., Kumarid, S., Rodrigues, J. J. P. C., Khan, M. K., & Hogrefe, D. (2018). EPA-CPPA: An efficient, provably-secure and anonymous conditional privacy-preserving authentication scheme for vehicular ad hoc networks. *Vehicular Communications*, 13, 40-50. doi: 10.1016/j.vehcom.2018.07.001.

[57] Sutrala, A. K., Bagga, P., Das, A. K., Kumar, N., Rodrigues, J. J. P. C., & Lorenz, P. (2020). On the design of conditional privacy preserving batch verification-based authentication scheme for internet of vehicles deployment. *IEEE Transactions on Vehicular Technology*, 69(5), 5535-5548. doi: 10.1109/TVT.2020.2981934.

[58] Ming, Y., & Cheng, H. (2019). Efficient certificateless conditional privacy-preserving authentication scheme in VANETs. *Mobile Information Systems*, 2019, 7593138. doi: 10.1155/2019/7593138.

[59] Tan, H., & Chung, I. (2020). Secure authentication and key management with blockchain in VANETs. *IEEE Access*, 8, 2482-2498. doi: 10.1109/ACCESS.2019.2962387.

[60] Li, J., Ji, Y., Choo, K. -K. R., & Hogrefe, D. (2019). CL-CPPA: Certificate-less conditional privacy-preserving authentication protocol for the internet of vehicles. *IEEE Internet of Things Journal*, 6(6), 10332-10343. doi: 10.1109/JIOT.2019.2938008.

[61] Wang, Y., Liu, Y., & Tian, Y. (2022). ISC-CPPA: Improved-security certificateless conditional privacy-preserving authentication scheme with revocation. *IEEE Transactions on Vehicular Technology*, 71(11), 12304-12314. doi: 10.1109/TVT.2022.3194060.

[62] Ali, I., Hassan, A., & Li, F. (2019). Authentication and privacy schemes for vehicular ad hoc networks (VANETs): A survey. *Vehicular Communications*, 16, 45-61. doi: 10.1016/j.vehcom.2019.02.002.

[63] Zhang, L., Wu, Q., Solanas, A., & D.-Ferrer, J. (2010). A scalable robust authentication protocol for secure vehicular communications. *IEEE Transactions on Vehicular Technology*, 59(4), 1606-1617. doi: 10.1109/TVT.2009.2038222.

[64] Zhang, C., Xue, X., Feng, L., Zeng, X., & Ma, J. (2019). Group-signature and group session key combined safety message authentication protocol for VANETs. *IEEE Access*, 7. doi: 10.1109/ACCESS.2019.2958356.

[65] Lim, K., Liu, W., Wang, X., & Joung, J. (2019). SSKM: Scalable and secure key management scheme for group signature based authentication and CRL in VANET. *Electronics*, 8. doi: 10.3390/electronics8111330.

[66] Jiang, Y., Ge, S., & Shen, X. (2020). AAAS: An anonymous authentication scheme based on group signature in VANETs. *IEEE Access*, 8, 98986-98998. doi: 10.1109/ACCESS.2020.2997840.

[67] Bottarelli, M., Epiphaniou, G., Kbaier, D., Karadimas, P., & Al-Khateeb, H. (2018, August). Physical characteristics of wireless communication channels for secret key establishment: A survey of the research. *computers and security*, 78, 454-476. doi: 10.1016/j.cose.2018.08.001.

[68] Xiao, L., Greenstein, L. J., Mandayam, N. B., & Trappe, W. (2008). Using the physical layer for wireless authentication in time-variant channels. *IEEE Transactions on Wireless Communications*, 7(7). doi: 10.1109/TWC.2008.070194.

[69] Tugnait, J. K. (2013). Wireless user authentication via comparison of power spectral densities. *IEEE Journal on Selected Areas in Communications*, 31(9). doi: 10.1109/JSAC.2013.130912.

[70] Chin, W.-L., Le, T. N., & Tseng, C.-L. (2015). Authentication scheme for mobile OFDM based on security information technology of physical layer over time-variant and multipath fading channels. *Information Sciences*, 321, 238-249. doi: 10.1016/j.ins.2015.01.040.

[71] Liu, J., & Wang, X. (2016). Physical layer authentication enhancement using two-dimensional channel quantization. *IEEE Transactions on Wireless Communications*, 15(6). doi: 10.1109/TWC.2016.2535442.

[72] Liu, J., Wang, X., & Tang, H. (2017). Physical layer authentication enhancement using maximum SNR ratio based cooperative AF relaying. *Wireless Communications and Mobile Computing*, 4, 1-16. doi: 10.1155/2017/7206187.

[73] Hamamreh, J., & Arslan, H. (2018). Joint PHY/MAC layer security design using ARQ with MRC and null-space independent, PAPR-aware artificial noise in SISO systems. *IEEE Transactions on Wireless Communications*, 17(9). doi: 10.1109/TWC.2018.2855163.

[74] Liao, R.-F., Wen, H., Wu, J., Pan, F., Xu, A., Jiang, Y., Xie, F., & Cao, M. (2019). Deep-learning-based physical layer authentication for industrial wireless sensor networks. *Sensors*, 19(11). doi: 10.3390/s19112440.

[75] Gao, N., Ni, Q., Feng, D., Jing, X., & Cao, Y. (2019). Physical layer authentication under intelligent spoofing in wireless sensor networks. *Signal Processing*. doi: 10.1016/j.sigpro.2019.107272.

[76] Li, N., Xia, S., Tao, X., Zhang, Z., & Wang, X. (2020). An area-based physical layer authentication framework to detect spoofing attacks. *Science China Information Sciences*, 63. doi: 10.1007/s11432-019-2802-x.

[77] Jadoon, A. K., Li, J., & Wang, L. (2021). Physical layer authentication for automotive cyber physical systems based on modified HB protocol. *Frontiers Computer Science*, 15. doi: 10.1007/s11704-020-0010-4.

[78] Zhang, J., Rajendran, S., Sun, Z., Woods, R., & Hanzo, L. (2019). Physical layer security for the internet of things: authentication and key generation. *IEEE Wireless Communications Magazine*, 26, 92-98. doi: 10.1109/MWC.2019.1800455.

[79] Hou, W., Wang, X., Chouinard, J-Y., & Refaey, A. (2014). Physical layer authentication for mobile systems with time-varying carrier frequency offsets. *IEEE Transactions on Communications*, 62(5). doi: 10.1109/TCOMM.2014.032914.120921.

[80] Fang, H., Wang, X., & Hanzo, L. (2018). Learning-aided physical layer authentication as an intelligent process. *IEEE Transactions on Communications*, 67. doi: 10.1109/TCOMM.2018.2881117.

[81] Wang, X., Hao, P., & Hanzo, L. (2016). Physical-layer authentication for wireless security enhancement: current challenges and future developments. *IEEE Communications Magazine*, 54. doi: 10.1109/MCOM.2016.7498103.

[82] Li, X., Liu, J., Ding, B., Li, Z., Wu, H., & Wang, T. (2019). A SDR-based verification platform for 802.11 PHY layer security authentication. *World Wide Web*. doi: 10.1007/s11280-018-0654-2.

[83] Ramabadran, P., Afanasyev, P., Malone, D., Leeser, M., McCarthy, D., O'Brien, B., Farrell, R., & Dooley, J. (2020). A novel physical layer authentication with PAPR reduction based on channel and hardware frequency responses. *IEEE Transactions on Circuits and Systems*, 67(2). doi: 10.1109/TCSI.2019.2952936.

[84] Widrow, B., & Stearns, S. D. (1985). Adaptive signal processing (Chapter 11, pp. 430-435). Prentice-Hall. doi: 10.1109/9780132447946.ch11.

[85] Shan, D., Zeng, K., Xiang, W., Richardson, P., & Dong, Y. (2013). PHY-CRAM: Physical layer challenge-response authentication mechanism for wireless networks. *IEEE Journal on Selected Areas in Communications*, 31(9), 1949-1959. doi: 10.1109/JSAC.2013.130914.

[86] Wu, X., & Yang, Z. (2015). Physical-layer authentication for multi-carrier transmission. *IEEE Communications Letters*, 19(1), 74-77. doi: 10.1109/LCOMM.2014.2375191.

[87] Cheng, L., Zhou, L., Seet, B.-C., Li, W., Ma, D., & Wei, J. (2017). Efficient physical-layer secret key generation and authentication schemes based on wireless channel-phase. *Mobile Information Systems (Hindawi)*, 2017. doi: 10.1155/2017/7393526.

[88] Ran, Y., Al-Shwaily, H., Tang, C., Tian, G. Y., & Johnston, M. (2019). Physical layer authentication scheme with channel based Tag Padding Sequence. *IET Communications*, 13, 1776-1780. doi: 10.1049/iet-com.2018.5749.

[89] Zhang, P., Liu, J., Shen, Y., Li, H., & Jiang, X. (2020). Lightweight tag-based PHY-layer authentication for IoT devices in smart cities. *IEEE Internet of Things Journal*, 7(5), 3977-3990. doi: 10.1109/JIOT.2019.2958079.

[90] Zhang, N., Fang, X., Wang, Y., Wu, S., Wu, H., Kar, D., & Zhang, H. (2020). Physical layer authentication for internet of things via WFRFT-based Gaussian tag embedding. *IEEE Internet of Things Journal*, 7(9), 9001-9010. doi: 10.1109/JIOT.2020.3001597.

[91] Althunibat, S., Sucasas, V., Mantas, G., & Rodriguez, J. (2018). Physical-layer entity authentication scheme for mobile MIMO systems. *IET Communications*, 12(6), 712-718. doi: 10.1049/iet-com.2017.0518.

[92] Wang, J., Shao, Y., Ge, Y., & Yu, R. (2020). Physical-layer authentication based on adaptive Kalman filter for V2X communication. *Vehicular Communications*, 26. doi: 10.1016/j.vehcom.2020.100281.

[93] Yang, J., Ji, X., Huang, K., Yi, M., & Chen, Y. (2017). AKA-PLA: Enhanced AKA based on physical layer authentication. *KSII Transactions on Internet and Information Systems*, 11(7), 3598-3617. doi: 10.3837/tiis.2017.07.024.

[94] Gope, P., Das, A. K., Kumar, N., & Cheng, Y. (2019). Lightweight and physically secure anonymous mutual authentication protocol for real-time data access in industrial wireless sensor networks. *IEEE Transactions on Industrial Informatics*, 15(9), 4999-5007. doi: 10.1109/TII.2019.2895030.

[95] Zenger, C. T., Pietersz, M., Zimmer, J., Posielek, J.-F., Lenze, T., & Paar, C. (2016). Authenticated key establishment for low-resource devices exploiting correlated random channels. *Computer Networks*, 109, 105-123. doi: 10.1016/j.comnet.2016.06.013.

[96] Chen, Y., Wen, H., Wu, J., Song, H., Xu, A., Jiang, Y., Zhang, T., & Wang, Z. (2019). Clustering based physical-layer authentication in edge computing systems with asymmetric resources. *Sensors*, 19(8), 1926. doi: 10.3390/s19081926.

[97] Yao, M., Wang, X., Gan, Q., Lin, Y., & Huang, C. (2021). An improved and privacy-preserving mutual authentication scheme with forward secrecy in VANETs. *Security and Communication Networks*, 1-13. doi: 10.1155/2021/6698099.

[98] Diffie, W., & Hellman, M. E. (1976). New directions in cryptography. *IEEE Transactions on Information Theory*, 22(6), 644-654. doi: 10.1109/TIT.1976.1055638.

[99] Wang, Q., Su, H., Ren, K., & Kim, K. (2011). Fast and scalable secret key generation exploiting channel phase randomness in wireless networks. In 2011 Proceedings IEEE INFOCOM (pp. 1422-1430). doi: 10.1109/INFCOM.2011.5934929.

[100] Zeng, K. (2015). Physical layer key generation in wireless networks: challenges and opportunities. *IEEE Communications Magazine*, 53(6), 33-39. doi: 10.1109/MCOM.2015.7120014.

[101] Tope, M., & McEachen, J. (2001). Unconditionally secure communications over fading channels. In 2001 MILCOM Proceedings Communications for Network-Centric Operations: Creating the Information Force (vol. 1, pp. 54-58). IEEE. doi: 10.1109/MILCOM.2001.985763.

[102] Mathur, S., Trappe, W., Mandayam, N., Ye, C., & Reznik, A. (2008). Radio-telepathy: extracting a secret key from an unauthenticated wireless channel. In Proceedings of the 14th ACM international conference on Mobile computing and networking - MobiCom '08 (pp. 128). ACM Press. doi: 10.1145/1409944.1409960.

[103] Kitaura, A., Iwai, H., & Sasaoka, H. (2007). A scheme of secret key agreement based on received signal strength variation by antenna switching in land mobile radio. In 2007 9th International Conference on Advanced Communication Technology (pp. 1763-1767). Gangwon, Korea (South). doi: 10.1109/ICACT.2007.358712.

[104] Guillaume, R., Ludwig, S., Müller, A., & Czylwik, A. (2015). Secret key generation from static channels with untrusted relays. In 2015 IEEE 11th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob) (pp. 635-642). Abu Dhabi, United Arab Emirates. doi: 10.1109/WiMOB.2015.7348022.

[105] Koorapaty, H., Hassan, A., & Chennakeshu, S. (2000). Secure information transmission for mobile radio. *IEEE Communications Letters*, 4, 52–55. doi: 10.1109/4234.824754.

[106] Mathur, S., Miller, R., Varshavsky, A., Trappe, W., & Mandayam, N. (2011). ProxiMate: proximity-based secure pairing using ambient wireless signals. Proceedings of the MobiSys conference, 211-224, June 2011. doi: 10.1145/1999995.2000016.

[107] Shehadeh, Y., & Hogrefe, D. (2011). An optimal guard-intervals based mechanism for key generation from multipath wireless channels. Proceedings of 4th IFIP International Conference on New Technologies, Mobility and Security, 1–5, February 2011. doi: 10.1109/NTMS.2011.5720584.

[108] Ye, C., Reznik, A., & Shah, Y. (2006). Extracting secrecy from jointly Gaussian random variables. In 2006 IEEE International Symposium on Information Theory (pp. 2593-2597). doi: 10.1109/ISIT.2006.262101.

[109] Goldsmith, A. (2012). Wireless communications. Cambridge University Press. doi: 10.1017/CBO9780511841224.

[110] Zhang, J., Marshall, A., Woods, R., & Duong, T. Q. (2014). Secure key generation from OFDM subcarriers' channel responses. In 2014 IEEE Globecom Workshops (GC Wkshps) (pp. 1302-1307). IEEE. doi: 10.1109/GLOCOMW.2014.7063613.

[111] Zhang, J., Woods, R., Marshall, A., & Duong, T. Q. (2015). An effective key generation system using improved channel reciprocity. In 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 1727-1731). IEEE. doi: 10.1109/ICASSP.2015.7178266.

[112] Zhang, J., Marshall, A., Woods, R., & Duong, T. Q. (2016). Efficient key generation by exploiting randomness from channel responses of individual OFDM subcarriers. *IEEE Transactions on Communications*, 64(6), 2578-2588. doi: 10.1109/TCOMM.2016.2552165.

[113] Baksi, S., Snoap, J., & Popescu, D. C. (2017, May). Secret key generation using one-bit quantized channel state information. Paper presented at IEEE Wireless Communications and Networking Conference (WCNC), 1-6. doi: 10.1109/WCNC.2017.7925527.

[114] Epiphaniou, G., Karadimas, P., Kbaier Ben Ismail, D., Al-Khateeb, H., Dehghantanha, A., & Choo, K. R. (2018, August). Nonreciprocity compensation combined with turbo codes for secret key generation in vehicular ad hoc social IoT networks. *IEEE Internet of Things Journal*, 5(4), 2496-2505. doi: 10.1109/JIOT.2017.2764384.

[115] Bottarelli, M., Karadimas, P., Epiphaniou, G., Ismail, D. K. B., & Maple, C. (2021, March). Adaptive and optimum secret key establishment for secure vehicular communications. *IEEE Transactions on Vehicular Technology*, 70(3), 2310-2321. doi: 10.1109/TVT.2021.3056638.

[116] Levy, B. C. (2008). Binary and Mary hypothesis testing. In Principles of Signal Detection and Parameter Estimation (pp. 27-32). Springer. doi: 10.1007/978-0-387-76544-0_2.

[117] Bellare, M., & Rogaway, P. (1993). Random oracles are practical: A paradigm for designing efficient protocols. In Proceedings of the 1st ACM Conference on Computer and Communications Security (pp. 62-73). doi: 10.1145/168588.168596.

[118] Karadimas, P., & Matolak, D. (2014, August). Generic stochastic modeling of vehicle-to-vehicle wireless channels. *Vehicular Communications*, 1(4), 153-167. doi: 10.1016/j.vehcom.2014.08.001.

[119] Karadimas, P., Vagenas, E. D., & Kotsopoulos, S. A. (2010, July). On the scatterers' mobility and second order statistics of narrowband fixed outdoor wireless channels. *IEEE Transactions on Wireless Communications*, 9(7), 2119-2124. doi: 10.1109/TWC.2010.07.080874.

[120] Berens, P. (2009, September). CircStat: A MATLAB toolbox for circular statistics. Journal of Statistical Software, 31(10), 1-21.

[121] Pfitzmann, A., & Köhntopp, M. (2001). Anonymity, unobservability, and pseudonymity - A proposal for terminology. In H. Federrath (Ed.), Designing Privacy Enhancing Technologies (pp. 1-9). Springer. Lecture Notes in Computer Science, vol. 2009. doi: 10.1007/3-540-44702-4_1.

[122] York, D. (2010). Control channel attacks: Fuzzing, DoS, SPIT, and toll fraud. In Seven Deadliest Unified Communications Attacks (pp. 71-92). Syngress. doi: 10.1016/B978-1-59749-547-9.00004-1.

[123] Faghihniya, M. J., Hosseini, S. M., & Tahmasebi, M. (2017, April). Security upgrade against RREQ flooding attack by using balance index on vehicular ad hoc network. *Wireless Networks*, 23, 1863-1874. doi: 10.1007/s11276-016-1259-2.

[124] Feistel, H. (1973, May). Cryptography and computer privacy. Scientific American, 228(5), 15-23. doi: 10.1038/SCIENTIFICAMERICAN0573-15.

[125] Dinov, I., Christou, N., & Sanchez, J. (2008). Central Limit Theorem: New SOCR applet and demonstration activity. *Journal of Statistics Education*, 16(2), 1-15. doi: 10.1080/10691898.2008.11889560.

[126] Scott, M. (2019). MIRACL Cryptographic Library: Multiprecision Integer and Rational Arithmetic C/C++ Library. Retrieved from https://github.com/miracl/MIRACL.

[127] Cui, J., Wang, Y., Zhang, J., Xu, Y., & Zhong, H. (2020, August). Full session key agreement scheme based on chaotic map in vehicular ad hoc networks. *IEEE Transactions on Vehicular Technology*, 69(8), 8914-8924. doi: 10.1109/TVT.2020.2997694.

[128] Mansour, M. B., Salama, C., Mohamed, H. K., & Hammad, S. A. (2018, March). VANET security and privacy – An overview. *International Journal of Network Security & Its Applications*, 10(2). doi: 10.5121/ijnsa.2018.10202.

[129] Yao, M., Wang, X., Gan, Q., Lin, Y., & Huang, C. (2021, April). An improved and privacy-preserving mutual authentication scheme with forward secrecy in VANETs. *Security and Communication Networks*. doi: 10.1155/2021/6698099.

[130] Arif, M., Wang, G., Bhuiyan, M. Z., Wang, T., & Chen, J. (2019, October). A survey on security attacks in VANETs: Communication, applications and challenges. *Vehicular Communications*. doi: 10.1016/j.vehcom.2019.100179.

[131] Zhang, L. (2008, August). Cryptanalysis of the public key encryption based on multiple chaotic systems. *Chaos, Solitons & Fractals*, 37(3), 669-674. doi: 10.1016/j.chaos.2006.09.047.

[132] Jungnickel, D. (1992). On the uniqueness of the cyclic group of order n. *American Mathematical Monthly*, 99(6), 545-547. doi: 10.1080/00029890.1992.11995889.

[133] Rife, D., & Boorstyn, R. (1974, September). Single-tone parameter estimation from discrete-time observations. *IEEE Transactions on Information Theory*, 20(5), 591-598. doi: 10.1109/TIT.1974.1055282.

[134] Liu, H., Wang, Y., Yang, J., & Chen, Y. (2013, April). Fast and practical secret key extraction by exploiting channel response. In Proceedings of IEEE INFOCOM (pp. 3048-3056). doi: 10.1109/INFCOM.2013.6567117.

[135] Li, Y., Cimini, L. J., & Sollenberger, N. R. (1998, July). Robust channel estimation for OFDM systems with rapid dispersive fading channels. *IEEE Transactions on Communications*, 46(7), 902-915. doi: 10.1109/26.701317.

[136] Wald, A. (1945, June). Sequential tests of statistical hypotheses. *Annals of Mathematical Statistics*, 16(2), 117-186.

[137] Kenney, J. B. (2011, July). Dedicated short-range communications (DSRC) standards in the united states. Proceedings of the IEEE, 99, 1162-1182. doi: 10.1109/JPROC.2011.2132790.

[138] Barker, E. B. (2000, December). A statistical test suite for random and pseudorandom number generators for cryptographic applications (800th ed.). National Institute of Standards and Technology.

[139] Ghosh, B. C., Bhartia, T., Addya, S. K., & Chakraborty, S. (2021, May). Leveraging public-private blockchain interoperability for closed consortium interfacing. In IEEE INFOCOM 2021 - IEEE Conference on Computer Communications (pp. 1-10). doi: 10.1109/INFOCOM42981.2021.9488683.

[140] Ogundoyin, S. O., & Kamil, I. A. (2021, October). An efficient authentication scheme with strong privacy preservation for fog-assisted vehicular ad hoc networks based on blockchain and neuro-fuzzy. *Vehicular Communications*, 31. doi: 10.1016/j.vehcom.2021.100384.

[141] Certicom Research. (2000, September). Standards for efficient cryptography, SEC 2: Recommended elliptic curve domain parameters 1.0, (pp. 9-10).

[142] Burrows, M., Abadi, M., & Needham, R. (1990). A logic of authentication. *ACM Transactions on Computer Systems*, 8(1), 18-36. doi: 10.1145/77648.77649.

[143] Armando, A., Basin, D., Boichut, Y., Chevalier, Y., Compagna, L., Cuéllar, J., & Mödersheim, S. (2005). The AVISPA tool for the automated validation of internet security protocols and applications. In International Conference on Computer Aided Verification (pp. 5-281). Springer. doi: 10.1007/11513988_27.

[144] Raczyński, M. (2021, January). What is the fastest blockchain and why? Analysis of 43 blockchains. Aleph Zero Website. Retrieved from https://alephzero.org/blog/what-is-the-fastest-blockchain-and-why-analysis-of-43-blockchains.

[145] Butler, C. (2021, March). Top 10 blockchains with transaction fees under 1$. DappRadar Website. Retrieved from https://dappradar.com/blog/top-10-blockchains-with-transaction-fees-under-1.

[146] Ahmad, K. (2021, December). 10 cryptocurrencies with almost zero transaction fees. MUO Website. Retrieved from https://www.makeuseof.com/cryptocurrencies-with-almost-zero-transaction-fees/.

[147] Son, S., Lee, J., Park, Y., Park, Y., & Das, A. K. (2022, June). Design of blockchain-based lightweight V2I handover authentication protocol for VANET. *IEEE Transactions on Network Science and Engineering*, 9(3), 1346-1358. doi: 10.1109/TNSE.2022.3142287.

[148] Cui, H., Wan, Z., Deng, R. H., Wang, G., & Li, Y. (2018, June). Efficient and expressive keyword search over encrypted data in cloud. *IEEE Transactions on Dependable and Secure Computing*, 15(3), 409-422. doi: 10.1109/TDSC.2016.2599883.

[149] Barker, E. (2020, May). Recommendation for key management. NIST Special Publication 800-57 Part 1 Revision 5, 53-54.

[150] Omnet++ discrete event simulator. (2019, November). Retrieved from https://omnetpp.org/.

[151] Simulation of urban mobility. (2022, June). Retrieved from https://sumo.dlr.de/docs/index.html.

[152] INET framework. (2022, June). Retrieved from https://inet.omnetpp.org/2021-05-18-INET-4.3.2-released.html.

[153] Sommer, C. et al. (2019). Veins: The open source vehicular network simulation framework. In: Virdis, A., Kirsche, M. (eds) Recent Advances in Network Simulation. EAI/Springer Innovations in Communication and Computing. Springer, Cham. doi: 10.1007/978-3-030-12842-5_6.

[154] Otoum, S., Ridhawi, I. A., & Mouftah, H. (2022). A federated learning and blockchain-enabled sustainable energy-trade at the edge: A framework for industry 4.0. *IEEE Internet of Things Journal*, 9(1), 111-120. doi: 10.1109/JIOT.2022.3140430.

[155] Rains, J., Tan, J., Zhang, Y., Ren, L., & Chen, Z. (2023). High-resolution programmable scattering for wireless coverage enhancement: An indoor field trial campaign. *IEEE Transactions on Antennas and Propagation*, 71(1), 518-530. doi: 10.1109/TAP.2022.3216555.

[156] Cui, J., Zhang, X., Zhong, H., Ying, Z., & Liu, L. (2019). RSMA: Reputation system-based lightweight message authentication framework and protocol for 5G-enabled vehicular networks. *IEEE Internet of Things Journal*, 6(4), 6417-6428. doi: 10.1109/JIOT.2019.2895136.

[157] Björnson, E., Wymeersch, H., Matthiesen, B., Popovski, P., Sanguinetti, L., & Carvalho, E. (2022). Reconfigurable intelligent surfaces: A signal processing perspective with wireless applications. *IEEE Signal Processing Magazine*, 39, 135-158. doi: 10.1109/MSP.2021.3130549.

[158] Lin, Z., Li, Y., Zhang, X., Li, Z., & Chen, X. (2022). Refracting RIS-aided hybrid satellite-terrestrial relay networks: joint beamforming design and optimization. *IEEE Transactions on Aerospace and Electronic Systems*, 58, 3717-3724. doi: 10.1109/TAES.2022.3155711.

[159] Lin, Z., Lin, M., Champagne, B., Zhu, W.-P., & Al-Dhahir, N. (2021). Secrecy-energy efficient hybrid beamforming for satellite-terrestrial integrated networks. *IEEE Transactions on Communications*, 69, 6345-6360. doi: 10.1109/TCOMM.2021.3088898.

[160] Niu, H., Li, J., Zhang, Q., Zhang, X., & Shen, X.S. (2023). Joint beamforming design for secure RIS-assisted IoT Networks. *IEEE Internet of Things Journal*, 10, 1628-1641. doi: 10.1109/JIOT.2022.3210115.

[161] Lu, X., Lei, J., Shi, Y., & Li, W. (2021). Intelligent reflecting surface assisted secret key generation. *IEEE Signal Processing Letters*, 28, 1036-1040. doi: 10.1109/LSP.2021.3061301.

[162] Krishna, N.J., & Prasanth, N. (2022). An insight view on denial of service attacks in vehicular ad hoc networks. In Advances in Computational Intelligence and Communication Technology (pp. 273-285). Springer, Singapore. doi: 10.1007/978-981-16-9756-2_27.

[163] You, C., Zheng, B., & Zhang, R. (2020). Intelligent reflecting surface with discrete phase shifts: channel estimation and passive beamforming. In Proceedings of IEEE International Conference on Communications (ICC) (pp. 1-6). Dublin, Ireland. doi: 10.1109/ICC40277.2020.9149292.

[164] NIST. (2001). A statistical test suite for random and pseudorandom number generators for cryptographic applications (800th ed.). National Institute of Standards and Technology.

[165] Wang, S., Li, N., Xia, S., Tao, X., & Lu, H. (2021). Collaborative physical layer authentication in internet of things based on federated learning. 2021 IEEE 32nd Annual International Symposium on Personal. Indoor and Mobile Radio Communications (pp. 714-719). Helsinki, Finland. doi: 10.1109/PIMRC50174.2021.9569531.

[166] Elbir, A. M., Papazafeiropoulos, A. K., & Chatzinotas, S. (2021). Federated learning for physical layer design. *IEEE Communications Magazine*. 59(11), 81-87. doi: 10.1109/MCOM.101.2100138.