Kasdi-Merbah Ouargla University

Faculty of New Information and Communication Technologies

Department of Computer Science and Information Technology

Thesis

Presented by:

Messaoud Benguenane

Submitted to obtain the degree of

Doctorate 3rd Cycle LMD in Computer Science

Option: Computer systems and networks

Thesis subject

Etudes et modélisation de la sécurité dans les réseaux sans fil Peer to Peer

Defended on: 27/09/2023.

Board of Examiners:

Abdelhakim Cheriet	MCA	University of Ouargla	Chairman
Ahmed Korichi	Pr.	University of Ouargla	Supervisor
Chaker Abdelaziz Kerrache	MCA	University of Laghouat	Examiner
Younes Guellouma	MCA	University of Laghouat	Examiner
Oussama Aiadi	MCA	University of Ouargla	Examiner
Billal Khaldi	MCA	University of Ouargla	Examiner

Acknowledgments

First and foremost, I would like to express my deepest gratitude to my research supervisor, Pr. Ahmed Korichi, Professor at Ouargla University, for his invaluable guidance, unwavering support, and dedicated involvement throughout every stage of my Ph.D. study process. His expertise, encouragement, and patience have been instrumental in shaping the direction of my research and academic growth. I am truly grateful for his mentorship and the opportunities he has provided me.

I would also like to extend my heartfelt thanks to the esteemed members of my thesis committee:

A special appreciation goes to Dr. Abdelhakim Cheriet, who graciously accepted the honor of serving as the president of my jury. I am sincerely grateful for his valuable insights, critical feedback, and expert evaluation, which have significantly enriched the quality of my thesis.

I extend my sincere gratitude to Dr. Chaker Abdelaziz Kerrache for accepting to be part of my thesis jury. His expertise and valuable contributions have greatly enhanced the rigor and depth of my research.

I would also like to express my deep appreciation to Dr. Younes Guellouma for agreeing to be a member of my thesis jury. His insightful comments and constructive suggestions have played a vital role in shaping the final outcome of my research.

I extend my sincere gratitude to Dr. Oussama Aiadi for accepting to be part of my thesis jury. His expertise and valuable contributions have greatly enhanced the rigor and depth of my research.

I would also like to express my deep appreciation to Dr. Billal Khaldi for agreeing to be a member of my thesis jury. His insightful comments and constructive suggestions have played a vital role in shaping the final outcome of my research.

Furthermore, I would like to extend my heartfelt thanks to Dr. Brik Bouziane for his unwavering support and invaluable assistance whenever I needed guidance or clarification. His expertise and willingness to help have been a constant source of motivation and inspiration.

Finally, I am profoundly grateful to my family: my loving father and mother, my supportive brothers, and my caring wife. Their unwavering belief in my abilities, endless encouragement, and understanding have been the driving force behind my accomplishments. Their love and support have sustained me throughout the writing of this thesis and throughout my life in general.

In conclusion, I would like to express my deepest appreciation to all those who have contributed to the completion of this thesis. Their support, guidance, and encouragement have played an instrumental role in my academic journey, and I am truly grateful for their presence in my life.

Messaoud Benguenane

الملخص

ساهم تطور وسائل النقل في تحسين العديد من جوانب حياة الإنسان. من ضمن هذا التطور ظهور شبكات الاتصال بين المركبات (VANETs) والتي صممت لتعزيز سلامة حركة المرور وتزويد السائقين بمعلومات السلامة المرورية. حيت تسمح هذه الشبكات للسيارات بالتواصل وتبادل البيانات فيما بينها أو مع وحدات جانب الطريق (RSU). لكن وللأسف هذه الشبكات تتأثر بالثغرات الأمنية مما يجعلها عرضة للهجمات السبرانية والتي بدورها قد تؤدي إلى وقوع حوادث تعرض حياة الناس للخطر. ولهذا فان التعامل مع هذه الثغرات الأمنية هو أمر بالغ الأهمية، حيث أن أبسط اهمال قد يؤدي الى عواقب وخيمة. في هذه الأطروحة نهدف إلى تطوير حلٍ أمنيٍ قويًا يضمن نقل البيانات بشكل آمن ويعزز الثقة بين المركبات المشاركة في تبادل المعلومات. حيث يتم تقسيم بحثنا الى ثلاثة أقسام رئيسية. أولاً، نقوم بمراجعة شاملة للثغرات الحالية المتعلقة بشبكات المركبات VANETs، مع التركيز بشكل خاص على أنواع مختلفة من الهجمات. حيث توفر هذه المراجعة فهمًا أساسيًا للتحديات الأمنية التي تواجه هذا النوع من الشبكات. ثانيًا، قمنا بدراسة الحلول الأمنية الحالية المقترحة لشبكات المركبات. من خلال تقييم دقيق، نقيم نقاط القوة والضعف لهذه الحلول وملاءمتها لبيئات VANET. استخلصنا من خلال هذا التحليل على ضرورة وجود بروتوكول أمني مبتكر يمكنه التعامل بشكل فعال مع المتطلبات والتحديات الفريدة لـ VANETs. ولتلبية هذه الحاجة، نقوم بتقديم بروتوكول توجيه آمن جديد يسمى SecE-V2X (بروتوكول توجيه آمن وفعال لاتصال المركبات مع كل شيء). تم تصميم SecE-V2X بشكل خاص لتوفير نقل بيانات آمن وموثوق لـ VANETs. من خلال الاستفادة من تقنيات التشفير المتقدمة وآليات الثقة وبروتوكولات الاتصال الفعالة، يضمن SecE-V2X نقل البيانات بشكل آمن ومصادقة بين المركبات في VANETs. من خلال تطوير وتقييم بروتوكول SecE-V2X، نساهم في تقدم الاتصال الآمن في VANETs. ثم من خلال محاكاة معمقة وتقييم الأداء، أثبتنا فعالية وكفاءة البروتوكول المقترح في التصدي للهجمات وتعزيز أمن شبكات المركبات.

الكلمات المفتاحية:.

شبكات الاتصال بين المركبات، أمن الشبكات، الهجمات، المحاكاة.

Abstract

The evolution of transportation has revolutionized various aspects of human life, offering numerous benefits. In order to improve traffic safety and give drivers essential safety information, a specialized class of mobile ad-hoc networks known as vehicular ad-hoc networks (VANETs) has arisen. However, the reliance on VANET applications for road safety also introduces vulnerabilities, making them susceptible to malicious attacks that can lead to accidents and jeopardize lives. Addressing these security concerns is of paramount importance, as even the smallest oversight can have devastating consequences. This thesis aims to develop a robust security solution that ensures secure data transmission and fosters trust among participating vehicles in VANETs. Our research is organized into three key sections. Firstly, we conduct a comprehensive review of existing vulnerabilities inherent in VANETs, with a specific emphasis on various types of attacks. This analysis provides a foundational understanding of the security challenges faced by VANETs. Secondly, we extensively investigate existing security solutions proposed for vehicular networks. Through a meticulous evaluation, we assess their strengths, weaknesses, and suitability for VANET environments. This critical analysis highlights the need for an innovative and efficient security protocol that can effectively address the unique requirements and challenges of VANETs. To fulfill this need, we introduce a novel secured routing protocol known as SecE-V2X (Secure and Efficient routing protocol for Vehicle-to-Everything). SecE-V2X is specifically designed to offer trusted and reliable routing for VANETs. Leveraging advanced cryptographic techniques, trust mechanisms, and efficient communication protocols, SecE-V2X ensures secure and authenticated data transmission among vehicles in VANETs. By developing and evaluating the SecE-V2X protocol, we contribute to the advancement of secure communication in VANETs. Through extensive simulations and performance evaluations, we show how our proposed protocol efficiently mitigates attacks and enhances the security of vehicle networks.

Keywords: VANET; V2X ; Networks ; Security; Attacks; Simulation.

Résumé

L'évolution des transports a révolutionné divers aspects de la vie humaine, offrant de nombreux avantages. Les réseaux ad hoc véhiculaires (VANETs) se sont imposés comme une classe spécialisée de réseaux ad hoc mobiles conçus pour améliorer la sécurité routière et fournir aux conducteurs des informations cruciales en matière de sécurité. Cependant, la dépendance aux applications VANET pour la sécurité routière introduit également des vulnérabilités, les rendant susceptibles d'attaques malveillantes pouvant entraîner des accidents et mettre des vies en danger. Il est primordial de répondre à ces problèmes de sécurité, car même la plus petite négligence peut avoir des conséquences dévastatrices. Cette thèse vise à développer une solution de sécurité robuste qui garantit une transmission de données sécurisée et favorise la confiance entre les véhicules participants dans les VANETs. Notre recherche est organisée en trois sections clés. Tout d'abord, nous procédons à une revue complète des vulnérabilités existantes inhérentes aux VANETs, en mettant l'accent sur différents types d'attaques. Cette analyse fournit une compréhension fondamentale des défis de sécurité auxquels sont confrontés les VANETs. Ensuite, nous examinons de manière approfondie les solutions de sécurité existantes proposées pour les réseaux de véhicules. Par le biais d'une évaluation minutieuse, nous évaluons leurs forces, leurs faiblesses et leur pertinence pour les environnements VANET. Cette analyse critique met en évidence la nécessité d'un protocole de sécurité innovant et efficace capable de répondre efficacement aux exigences et aux défis uniques des VANETs. Pour répondre à ce besoin, nous introduisons un nouveau protocole de routage sécurisé appelé SecE-V2X (Secure and Efficient routing protocol for Vehicle-to-Everything). SecE-V2X est spécifiquement conçu pour offrir un routage fiable et de confiance pour les VANETs. En tirant parti de techniques cryptographiques avancées, de mécanismes de confiance et de protocoles de communication efficaces, SecE-V2X garantit une transmission sécurisée et authentifiée des données entre les véhicules dans les VANETs. En développant et en évaluant le protocole SecE-V2X, nous contribuons à l'avancement de la communication sécurisée dans les VANETs. À travers des simulations approfondies et des évaluations de performances, nous démontrons l'efficacité et l'efficience de notre protocole proposé dans la lutte contre les attaques et l'amélioration de la sécurité des réseaux de véhicules.

Mots-clés: VANET; V2X ; Réseaux ; Sécurité; Attaques; Simulation.

Contents

List of Figures

List of Tables

List of Code Segments

List of Publications

Published Journal Paper

Benguenane, M.; Korichi, A.; Brik, B.; Azzaoui, N. Towards Mitigating Jellyfish Attacks Based on Honesty Metrics in V2X Autonomous Networks. *Applied Sciences* 2023, *13*, 4591, doi:10.3390/app13074591.

Published Conference Papers

Benguenane, M.; Korichi, A.; Azzaoui, N. Geographical Routing Protocols in VANETs: Performance and Security Analysis. In Proceedings of the 2nd International Conference on Industry 4.0 and Artificial Intelligence (ICIAI 2021); Atlantis Press, 2022; pp. 158–163, doi: 10.2991/aisr.k.220201.028.

List of Abbreviations

Acronym	Description
ACA	Autonomous Component Architecture
AES-CCM	Advanced Encryption Standard Counter with CBC-MAC
ANN	Artificial Neural Network
AODV	Ad-hoc On-Demand Distance Vector
API	Application Programming Interface
ATA	Agent Trusted Authority
BSM	Basic Safety Message
CA	Certificate Authority
CALM	Continuous Air-interface, Long and Medium range
CAR	Connectivity Aware Routing Protocol
CCH	Control Channel
CityMob	City Mobility
CoE	Certainty of Event
COIN	Clustering for Open Inter Vehicular Communication Network
CONFIDENT	Cooperation Of Nodes: Fairness In Dynamic Ad hoc NeTworks
CRL	Certificate Revocation List
DDoS	Distributed Denial of Service
DoS	denial-of-service
DSDV	Destination Sequenced Distance Vector
DSR	Dynamic Source Control Routing
DSRC	Dedicated Short Range Communication
DYMO	Dynamic Manet on Demand
ECC	Elliptic Curve Cryptography
ECDSA	Elliptic Curve Digital Signature Algorithm
ECIES	Elliptic Curve Integrated Encryption Scheme
EDR	Event Data Recorder
EEBL	Electronic Emergency Brake Light
EED	End-to-End Delay
ERS	Dynamic Event-Based Reputation System
ETSI	European Telecommunications Standards Institute
FCC	Federal Communications Commission
GloMoSim	Global Mobile Information system Simulator
GNED	Graphical Network Editor
GNU	General Public License
GPCR	Greedy Perimeter Coordinator Routing
GPS	Global Positioning System
GPSR	Greedy perimeter stateless routing protocol
GrooveNet	Groove-based Vehicular Network Simulation Environment
GSM	Global System for Mobile Communications
GTNetS	Georgia Tech Network Simulator
GUI	Graphical User Interface
HTTP	Hypertext Transfer Protocol
IBS	ID-Based Signature
IBV	Identity-based Batch Verification

ID	Identity
IDE	Integrated Development Environment
IDS	Intrusion Detection System
IEEE	Institute of Electrical and Electronics Engineers
IoT	Internet of Things
IoV	Internet of Vehicles
IP	Internet Protocol
ISO	International Organization for Standardization
ITS	Intelligent Transport System
IVC	Inter-Vehicular Communication
J-SIM	Java SIMulator
LDM	Local Dynamic Map
LDW	Local Danger Warning
LOP	Location Obscurer Proxy
MA	Misbehavior Authority
MAC	Medium access control
MANET	Mobile Ad-Hoc Network
ML	Machine Learning
MOVE	MObility model generator for VEhicular networks
NCTUns	National Chiao Tung University network simulator
NED	Network Description
NS-2	Network Simulator 2
NS-3	Network Simulator 3
OBU	On-Board Unit
OLSR	Optimized Link State Routing
OMNeT++	OBJECTIVE MODULAR NETWORK TESTBED in C++
OPNET	OPtimized Network Engineering Tool
OTCL	Object-oriented Tool Command Language
PARSEC	Parallel Simulation Environment for Complex System
PBFT	Practical Byzantine Fault Tolerance
PDC	Packet Drop Count
PDR	Packet Delivery Ratio
PKI	Public Key Infrastructure
PoS	Proof of Stake
PoW	Proof of Work
PTD	Packet Transfer Delay
PTI	Packet Transfer Interval
QoE	Quality of Experience
QoS	Quality of service
QualNET	Quality Networking
RCP	Resource Command Processor
RERR	Route Error
RMA	Resource Manager Applications
RREP	Route Reply
RREQ	Route Request
RSS	Received Signal Strength
RSU	Road Side Unite
RTA	Regional Trusted Authority
RTO	Regional Transport Office

SCH	Service Channel
SCMS	Security Certificate Management System
SEAD	Secure Efficient Ad hoc Distance Vector
SecE-V2X	Secure and efficient routing protocol for vehicle-to-everything
STRAW	STreet RAndom Waypoint
SUMO	Simulation of Urban Mobility
SVM	Support Vector Machine
TCP	Transmission Control Protocol
TMHSC	Trust Management Hybrid Cryptography
TPD	Trusted Personal Device
TPM	Trusted Platform Module
TraCI	Traffic Control Interface
TraNS	Traffic and Network Simulation Environment
UDP	User Datagram Protocol
UMTS	Universal Mobile Telecommunications Service
V2I	Vehicle-to-Infrastructure
V2V	Vehicle-to-Vehicle
V2X	Vehicle-to-Everything
VANET	vehicular Ad-hoc Network
VANETMobiSim	Vehicular Ad hoc Network Mobility Simulator
VARS	Vehicle Ad Hoc Reputation System
VEINS	Vehicles in Network Simulation
VENTOS	VEhicular NeTwork Open Simulator
WAVE	Wireless Access in Vehicular Environments
WIFI	Wireless Fidelity
WLAN	Wireless Local Area Network
WSA	WAVE Service Advertisement
WSM	WAVE Short Message
WSMP	WAVE Short Message Protocol
WSN	Wireless Sensor Network
ZRP	Zone Routing Protocol

Chapter 1

General Introduction

Content

1. Introduction

The transfer of data between computers has undergone a significant evolution since the first computers were developed. Early data transfer was accomplished using physical media, such as punch cards and magnetic tapes. As computing technology advanced, wired connections such as Ethernet cables emerged, connecting computers to local networks.

With the proliferation of mobile devices, the need for seamless and reliable data transfer has become more important than ever before. One of the key developments in this regard has been the emergence of wireless networks, which have made it possible to establish communication links without the need for physical cables or fixed infrastructure. The first wireless networks emerged in the 1990s, using infrared and radio waves to transmit data between devices. However, these early wireless networks had limited range and speed, and were not suitable for large-scale deployment[1].

The development of Wireless Fidelity (WIFI) technology in the late 1990s revolutionized wireless networking, providing high-speed and reliable connectivity over longer distances. Due to the widespread use of mobile devices like smartphones and tablets, WIFI has become a ubiquitous technology in modern society, enabling people to stay connected to the internet from almost anywhere [2].

In addition to WIFI, ad hoc networking has also emerged as a useful technology for wireless communication. Ad hoc networks are formed between devices without the need for a centralized infrastructure, making them suitable for scenarios where traditional networking is not feasible. Military operations, emergencies, and disaster assistance have all employed ad hoc networks. The ability of ad hoc networks to operate without the need for a centralized infrastructure makes them particularly useful in situations where traditional communication channels are unavailable or unreliable [3].

On the other hand, vehicular Ad-hoc Networks (VANETs) are a specific type of ad hoc network that enables communication between vehicles and infrastructure. VANETs have the potential to improve road safety, reduce traffic congestion, and enable new services and applications for drivers and passengers. The key challenge for VANETs is ensuring reliable communication in a dynamic and mobile environment, where the relative positions of the moving vehicles are perpetually changing [4].

Furthermore, Vehicle-to-Everything (V2X) is a broader term that includes not only VANETs but also vehicle-to-infrastructure, vehicle-to-pedestrian, and vehicle-to-network communication.

V2X technologies enable communication between vehicles and other road users, including infrastructure like traffic lights, pedestrians, and other vehicles. The potential applications of V2X technologies are numerous, including collision avoidance systems, traffic signal coordination, and real-time traffic information [5].

In recent years, VANET and V2X technologies have gained increasing attention from researchers, industry, and governments for the purpose of enhancing road safety and efficiency. With the emergence of autonomous and networked automobiles, the importance of VANET and V2X is only expected to grow in the coming years. The ability to connect vehicles to each other and to infrastructure in real-time will enable new services and applications that could revolutionize the way we move around our cities and beyond [6].

Wireless networks, including VANET, have introduced new security challenges, especially given the sensitive nature of the data being transmitted. In VANET networks, vehicles communicate with each other and with roadside infrastructure to share their positions, rates of movement, and intended actions. Using this information can improve road safety, reduce congestion in traffic, and enable new services and applications for drivers and passengers. However, it also creates potential security risks. Malicious actors could use VANET networks to launch cyber attacks, such as jamming or spoofing, that could lead to accidents or other disruptions. Ensuring the security of VANET and V2X networks is therefore crucial for their successful deployment. Various security mechanisms, such as encryption and authentication, have been proposed to address these concerns. However, the unique characteristics of VANET and V2X networks, such as the highly dynamic and mobile nature of vehicles, make designing effective security mechanisms a challenging task [7].

In this context, this research project aims to review various existing vulnerabilities that VANETs may be susceptible to. Secondly, we identified and analyzed the existing security solutions within vehicular networks. Then, we introduce a new secured routing protocol called SecE-V2X (Secure and Efficient routing protocol for Vehicle-to-Everything) which offers secured and trusted routing for VANET.

2. Research inevitability

The advancement of wireless communication and sensor technologies has led to the emergence of vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication. This innovative approach holds great potential in enhancing road safety, alleviating traffic congestion, and enriching the driving experience[8].

However, the successful deployment of VANET technology faces several challenges, including the need for robust and reliable communication protocols, the requirement for secure and efficient data transmission, and the need for a scalable and effective network infrastructure. Moreover, the unique characteristics such as their dynamic nature and the transmission of sensitive information in VANETs also make them vulnerable to various security threats, such as denial-of-service (DoS) attacks. These attacks can significantly influence the availability and reliability of communication channels, leading to potentially disastrous consequences on the road [9,10].

The existing security solutions within vehicular networks are limited in their ability to defend against sophisticated attacks[11,12]. Hence, a more comprehensive understanding of the vulnerabilities that VANETs are susceptible to is required. Additionally, there is a need for the development of new security protocols that can mitigate these vulnerabilities and provide a more secure and reliable communication infrastructure for vehicular networks.

3. Thesis context

Today, the trend of governments is more than ever oriented towards the development of smart cities, to offer technological infrastructures for intelligent applications. These infrastructures enable communication between their users, allowing the exchange and sharing of relevant information for various applications, including those related to road safety and comfort. This new revolution aims to orient scientific research towards this new type of infrastructure [13].

Among these infrastructures, VANETs are gaining prominence as a highly promising wireless technology that underpins Intelligent Transport Systems (ITS) and plays a crucial role in achieving the objectives of ITS, including collision avoidance, efficient traffic management, and provision of infotainment services. However, due to the various specificities and dynamic nature that characterize vehicular networks, they are still in the experimental phase and several issues need to be addressed before such a network can be deployed. These issues include channel access control, data routing, data dissemination, data collection and exchange, service provisioning, and data security [14]. Moreover, these networks are susceptible to a wide array of security threats, including DoS attacks, identity theft, and data privacy breaches. These threats pose a significant challenge to the deployment of VANETs, particularly in safety-critical applications, such as autonomous driving and emergency response [15].

4. Vehicular Ad hoc NETworks

VANETs are a distinct subset of Mobile Ad-Hoc Networks (MANETs) that are specifically designed to enable communication among vehicles. The participating nodes are vehicles equipped with onboard computers, network cards, and sensors capable of collecting and processing information [16]. Within a VANET, vehicles or nodes have the ability to engage in communication through Vehicle-to-Vehicle (V2V) channels, enabling direct interaction and data exchange. Additionally, VANETs facilitate Vehicle-to-Infrastructure (V2I) communication, allowing vehicles to connect with infrastructure components like Road Side Units (RSUs) [17], as illustrated in Figure 1.1. The main objectives of this type of network are to offer those applications that enable the construction of an Intelligent Transport System, and those related to driver safety and comfort [18].

An Intelligent Transport System (ITS) is a system that uses advanced technologies to improve transportation safety, efficiency, and sustainability. An essential element of ITS is the Vehicle-to-Everything (V2X) communication technology, which facilitates seamless communication between vehicles as well as other elements of the transportation infrastructure. This technology empowers vehicles to exchange information with each other and establish connections with various entities such as traffic signals and road-side units [19].

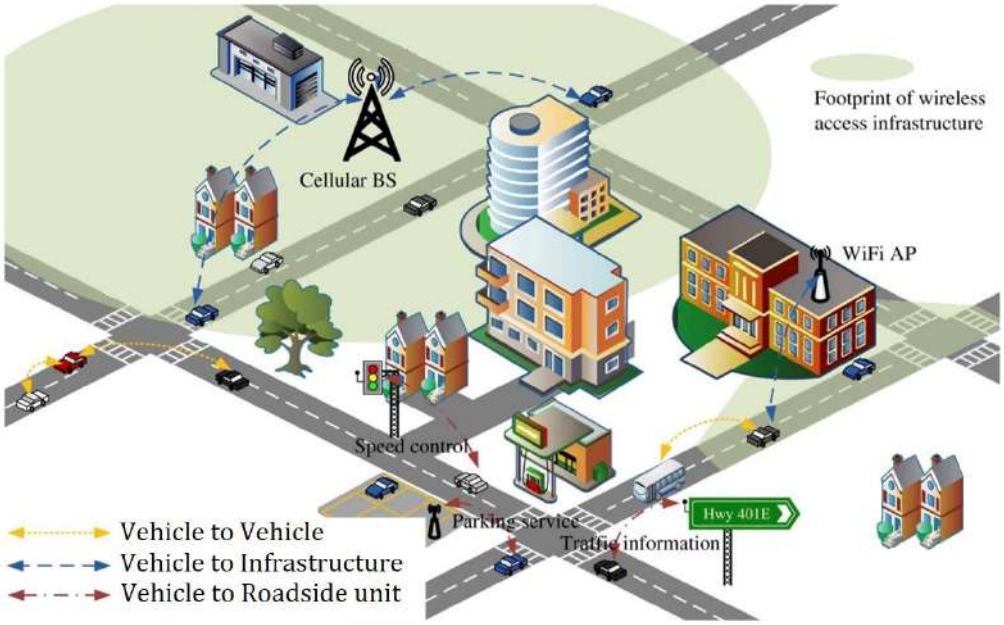

Figure 1.1. Communication in vehicular ad hoc networks.

In VANET, vehicles can share information in real-time with each other and the transportation infrastructure, allowing for faster and more accurate responses to incidents, better traffic management, improved navigation, and enhanced overall transportation system efficiency [20].

4. 1. Characteristics of VANET

Vehicular Ad Hoc Networks (VANETs) possess unique characteristics that distinguish them from other wireless network types and need to be carefully considered during the design of protocols for this type of network. In the following, we present the characteristics of VANETs, focusing on [21]:

- **High Mobility:** The primary characteristic of VANETs is high mobility, as the nodes in the network are vehicles that can move at high speeds. This high mobility results in frequent topology changes, making routing and communication challenging.
- **Dynamic Network Topology:** The topology of a VANET is highly dynamic due to the mobility of the nodes. This dynamic nature of the network requires the use of adaptive and efficient routing protocols to ensure that data can be transmitted effectively.
- **Self-Organizing:** VANETs are self-organizing, meaning that they do not require a fixed infrastructure to operate. Instead, the vehicles themselves form the network and act as nodes, creating a decentralized system.
- **Limited Communication Range:** The communication range in VANETs is limited due to the use of wireless communication technologies. The limited communication range requires the use of multi-hop routing protocols to extend the range of communication.
- **Multi-Hop Communication:** Due to the limited communication range, VANETs require multi-hop communication, where messages are relayed through intermediate nodes to reach their destination.
- **High Bandwidth Requirements:** VANETs require high bandwidth to support the transmission of multimedia data, such as video and audio, for applications such as entertainment and traffic monitoring.
- **Safety-Critical Applications:** VANETs are often used for safety-critical applications such as collision avoidance and traffic monitoring. Therefore, reliable and fast communication is essential to ensure the safety of passengers and other road users.

- **Security and Privacy Concerns:** VANETs are vulnerable to security and privacy attacks due to the open nature of the network and the broadcast nature of wireless communication. Therefore, security and privacy measures must be put in place to protect the data and the users' privacy.

4. 2. VANET Applications

The embedded resources and communication capabilities of intelligent vehicles have made it possible to envision several applications for vehicular networks. Three main categories can be used to group these applications [22].

- **Security:** Security-related applications are specifically developed to enhance the safety of both drivers and passengers. These applications encompass a range of functionalities such as collision warning systems, lane departure warning systems, intersection collision avoidance systems, and various other safety-oriented features. These applications rely on the ability of vehicles to communicate with each other in real-time and provide early warnings to drivers about potential hazards on the road.
- **Traffic Management:** Traffic management applications are designed to improve the overall efficiency of traffic flow on the road. These applications include traffic congestion detection and mitigation, dynamic routing, and intelligent traffic signal control, among others. By providing real-time information about traffic conditions and routing suggestions to drivers, these applications can help reduce traffic congestion and improve overall travel times.
- **Comfort and Entertainment:** Comfort and entertainment applications are designed to improve the overall driving experience for drivers and passengers. These applications include internet access, infotainment systems, and social networking, among others. By providing drivers and passengers with access to a range of entertainment and communication options, these applications can help make long drives more enjoyable and reduce driver fatigue.

4. 3. VANET Standardization

Standardization and normalization are crucial for ensuring the successful deployment and operation of VANETs, as they help to ensure interoperability, compatibility, and reliability across different systems and devices. In order to achieve this, various standardization organizations have been working to develop and promote standards and guidelines for VANETs. Although many

organizations have launched standardization and normalization activities for VANETs, they are often strongly influenced by the concerns and particular interests of the regions in which they are most active. To our knowledge, several standardization projects have been abandoned, such as the ISO project, which aimed to provide vehicles with a software platform called CALM (Continuous Air-interface, Long and Medium range), and the ETSI (European Telecommunications Standards Institute) projects, except for the IEEE project which has been completed and is the most widely used by the VANET community. These organizations have developed a range of standards and guidelines related to VANETs, including protocols for communication, security, and network management. By adhering to these standards and guidelines, developers and manufacturers can ensure that their products and systems are compatible with other VANET systems and devices and that they meet the requirements for safety and reliability [23].

The IEEE has played a major role in the standardization of VANETs. In 1999, the Federal Communications Commission (FCC) in the United States allocated a dedicated frequency band of 75 MHz between 5.850 and 5.925 GHz for inter-vehicle communications, known as Dedicated Short Range Communication (DSRC) [24]. DSRC divides this band into seven 10 MHz communication channels, along with a 5 MHz guard band as shown in Figure 1.2. Among these channels, one is designated as the Control Channel (CCH) for transmitting control packets and alerts, while the remaining six channels serve as Service Channels (SCH) for other types of data messages.

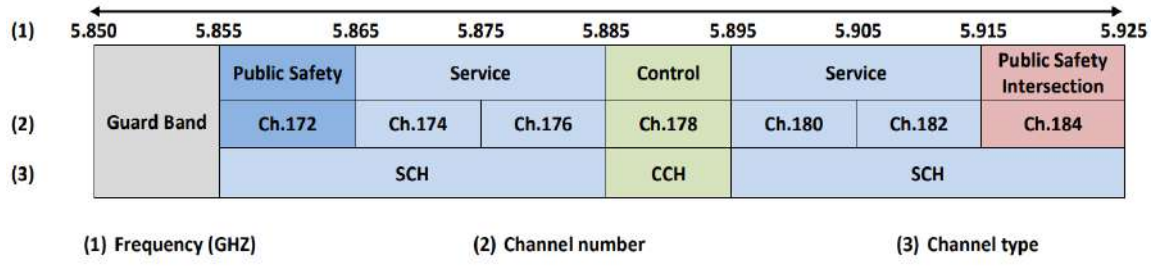

Figure 1.2. DSRC frequency bands

In order to define a standard for inter-vehicle communications by including the 802.11p norm[25], the IEEE organization gained control of the DSRC band in 2003. In particular, the IEEE 802.11p standard, also known as Wireless Access in Vehicular Environments (WAVE), is a set of protocols specifically designed for vehicular communication. This standard is based on the IEEE 802.11 standard, which is widely used for Wireless Local Area Networks (WLANs). In addition to the IEEE 802.11p standard, the IEEE has also developed other standards related to VANETs, such as the IEEE 1609 series of standards for WAVE. These standards cover topics

such as security, network management, and application support and they are structured into four components (see Figure 1.3) [23,26]:

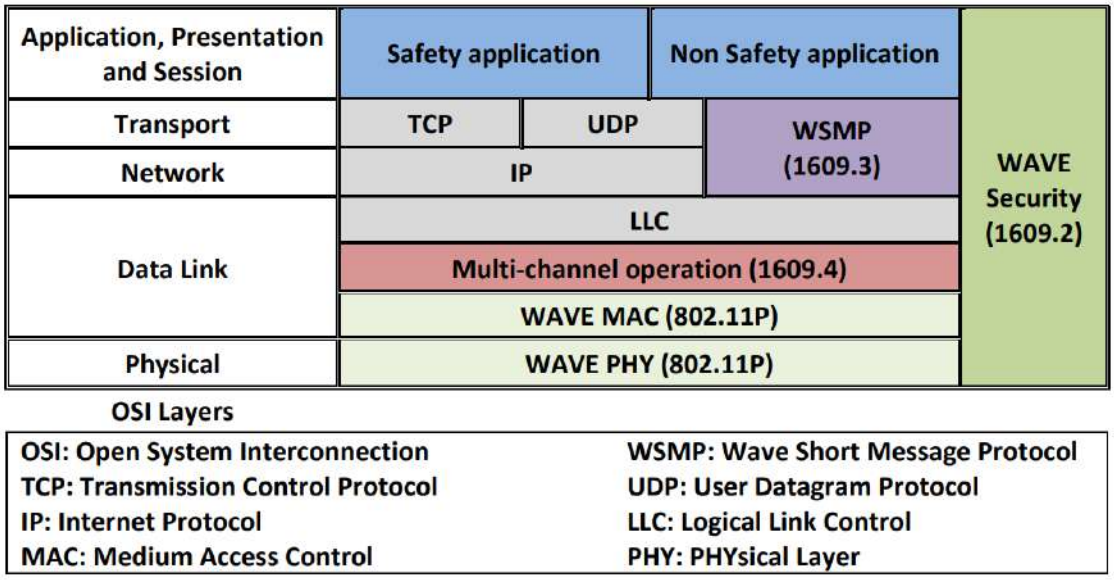

Figure 1.3. The WAVE 1609 model

- **IEEE 1609.1:** defines how messages are structured and stored in the application layer. It comprises three components: the Resource Manager Applications (RMA), which is a remote entity that communicates with the Resource Manager (RM), and the Resource Command Processor (RCP), which receives and executes commands from the RMA.

- **IEEE 1609.2:** outlines a set of methods that govern the security of management and application messages within the DSRC/WAVE system. It provides guidelines on how vehicles can effectively establish and maintain the authenticity, confidentiality, integrity, and non-repudiation of messages exchanged in the system. Depending on the specific security service utilized, the format of messages may vary. For instance, transaction messages undergo both signing and encryption processes, whereas alert messages are solely signed.

- **IEEE 1609.3:** introduces the WAVE Short Message (WSM) and the corresponding WAVE Short Message Protocol (WSMP), which facilitate network and transport layer functionalities for road safety applications. The WSMP is specifically designed to accommodate low-latency applications like Local Danger Warnings (LDW). Moreover, this protocol defines the WAVE Service Advertisement (WSA) message, which serves to announce the existence of DSRC services at a particular location. For instance, it can indicate the availability of a traffic information service provided by a RSU.

- **IEEE 1609.4:** in conjunction with IEEE 802.11p, focuses on the physical layer of the DSRC system used in VANETs. Specifically, it defines the operational characteristics of the DSRC system within the frequency band of 5.850 GHz to 5.925 GHz. This frequency band is divided into seven 10 MHz channels, comprising one control channel (CCH) and six service channels (SCH). These channels support data rates ranging from 6 to 27 Mbit/s. IEEE 1609.4 outlines the organization, scheduling, and utilization of these channels to enable multiple devices to communicate simultaneously on the same channel.

4. 4. Routing in VANET

Routing in VANET is the process of finding a communication path between two or more vehicles or between a vehicle and an infrastructure node (such as a roadside unit). This enables the exchange of data between the communicating nodes. However, VANET environments are more complex than traditional ad-hoc networks because of the fast-changing topology, which can cause delays and data packet loss when the routing process changes the chosen paths. As a result, numerous routing protocols have emerged in response to the specific challenges of VANETs. These protocols can be classified into three primary categories: topology-based protocols, geo-based protocols, and cluster-based protocols [27].

4. 4. 1. Topology-based protocols

Initially developed for MANETs, topology-based protocols were eventually modified for use in vehicle networks due to their similar properties such as decentralized control and mobility. However, VANETs have unique characteristics such as high node mobility and the need for mobility patterns. Topology-based protocols work by selecting topological links between network nodes to establish end-to-end paths between the source and destination nodes. Topology-based routing protocols in VANETs can be further divided into three subcategories: proactive, reactive, and hybrid. Proactive protocols ensure that routing information is continuously updated and shared with all nodes in the network at regular intervals. Reactive protocols, on the other hand, establish routes on-demand and do not maintain routing information for the entire network. Hybrid protocols employ a combination of proactive and reactive strategies, allowing for the benefits of both approaches to be utilized effectively [28].

- **Reactive routing protocols**

Reactive routing protocols in VANETs are a specific type of routing protocol that establish a route to the destination node solely when a request is initiated. These protocols do not engage in route maintenance when there are no active requests within the network. In order to determine the path to the destination node, a route discovery process is started prior to transmitting the request. Some popular protocols in this category are Dynamic Source Control Routing (DSR) [29], Dynamic Manet on Demand (DYMO) [30], and Ad-hoc On-Demand Distance Vector (AODV) [31], which is widely used in VANET after being introduced in MANET networks.

- **Proactive routing protocol**

Proactive routing protocols in VANETs are specifically designed to constantly maintain current information about the entire network. These protocols establish routes between all nodes in the network, regardless of whether they are actively being used or not. Routes are regularly updated by transmitting packets throughout the network, without taking into account factors such as network load, data transfer rate, or network size. Optimized Link State Routing (OLSR) [32] and Destination Sequenced Distance Vector (DSDV) [33] are some of the more well-known protocols in this routing category.

- **Hybrid routing protocols**

Hybrid routing protocols in VANETs represent a fusion of reactive and proactive routing protocols, intending to improve routing efficiency and scalability by leveraging the strengths of both types. These protocols aim to mitigate the limitations associated with purely proactive or reactive routing approaches. They achieve this by reducing the overhead generated during proactive or reactive routing and minimizing the transmission delay typically encountered in reactive protocols when transmitting data. Hybrid protocols focus on making the route discovery process more efficient. They are particularly well-suited for vehicular networks characterized by a relatively

small number of nodes and low mobility. Among the hybrid routing protocols, the Zone Routing Protocol (ZRP) stands out as one of the most widely adopted solutions [34].

4. 4. 2. Geographical routing protocols

Geographical routing protocols specifically cater to vehicular networks and operate based on the geographical location of the nodes. These protocols utilize geolocation technologies such as the Global Positioning System (GPS) to determine the positions of the nodes and calculate the optimal path to the destination by traversing through intermediate nodes. Unlike other routing protocols, nodes in geographical routing protocols do not maintain routing tables for remote nodes beyond a single hop, nor do they exchange information regarding link states with neighboring nodes. Instead, when sending a request, a node must possess knowledge of the destination's position and include it in the packet header, enabling intermediate nodes to determine the appropriate forwarding direction [35]. Geographical routing protocols encompass four main types: geo-unicast, geo-multicast (geo-anycast), geocast (geo-broadcast), and temporal geocast [23].

- **Geo-unicast:** When information reaches a destination area, it is transmitted to a specific node there.
- **Geo-anycast:** When the message arrives, any node in the destination area will receive the information.
- **Geocast:** All nodes present in the destination area at the time the message arrives are the intended recipients of the information.
- **Time Geocast:** When a message arrives for a specific amount of time, information is sent to every node in the destination area.

There is a lot of ongoing research in the field of geographic routing, and many protocols have been created to facilitate this type of routing. Some of these protocols include Greedy Perimeter Stateless Routing Protocol (GPSR) [36], Greedy Perimeter Coordinator Routing (GPCR) [37], and Connectivity Aware Routing Protocol (CAR) [38]. GPSR is a popular unicast geographic routing protocol that works well in highway environments with uniformly distributed nodes. This protocol relies on the location and address of each node in the network for the routing process. The contributions in this thesis are based on it.

4. 4. 3. Routing protocols based on clustering

Clustering routing protocols are specifically designed for networks characterized by the presence of clusters of nodes. In the context of vehicular networks, nodes often form clusters based on their spatial proximity. Each node within a cluster can assume the role of a cluster head, gateway, or member. The cluster head is responsible for maintaining information pertaining to gateways and members, facilitating direct communication among the nodes within the cluster. The packet transmission process in clustering routing protocols is similar to that of the Ad hoc On-Demand Distance Vector (AODV) protocol, with the distinction that relaying of data or control packets is limited to the cluster head and gateways. However, configuring clusters and selecting suitable cluster heads present challenges in this type of protocol. The establishment and maintenance of clusters incur significant delays and overhead within the network, and certain protocols necessitate the presence of Road Side Units (RSUs) to support these operations [39]. One notable clustering protocol employed in vehicular networks is the Clustering for Open Inter Vehicular Communication Network (COIN) [40], which leverages Global Positioning System (GPS) data to divide the network into clusters based on factors such as node mobility, driver behavior, and inter-vehicle distances.

4. 5. Challenges and issues in VANET

Vehicle Ad-hoc Networks (VANETs) have significant promise for enhancing traffic flow, lowering congestion, and enabling a wide range of new services and applications. However, there are still several challenges and issues that need to be addressed to fully realize the benefits of VANETs[23].

4. 5. 1. Infrastructure deployment

VANETs require the deployment of infrastructure such as RSUs to support communication and data dissemination. The cost and maintenance of this infrastructure can be a significant challenge, particularly in rural areas with low population density.

4. 5. 2. Scalability

Scalability is a crucial factor in VANET since it determines the network's ability to manage a significant number of nodes effectively. However, the density of mobile nodes in VANET can vary significantly depending on the vehicular environment and traffic conditions. For instance, during peak traffic hours or accidents, hundreds of vehicles can be present on the road, leading to congestion in message dissemination and increasing packet loss and transmission latency in the network. Therefore, it is essential to address the challenges posed by scalability to ensure efficient

communication among nodes in VANET networks. For that, more researches are needed in this field to find effective solutions to deal with large numbers of nodes in VANET networks.

4. 5. 3. Mobility and Robustness of routing

The rapid mobility and high node density in vehicular networks make data packet routing more challenging, requiring more robust and self-configuring routing protocols. Routing resilience allows networks to sustain satisfactory performance even when confronted with challenges such as link or node failures, node removal, or malicious attacks. Self-configuring routing involves resolving network issues solely through the collaborative efforts of individual nodes, without relying on centralized control. At present, VANET data routing techniques are adaptations of those employed in MANETs, despite the contrasting mobile node contexts between the two. Consequently, there is a demand for fresh insights to inspire the development of efficient, decentralized routing protocols that are robust and capable of self-organization within VANETs.

4. 5. 4. Security and privacy

The security of vehicular networks is a critical issue that involves controlling access to the network and preventing unauthorized access, misuse, or denial of network resources. One of the major challenges in VANET security is ensuring authenticity, which involves verifying the validity of user identities and protecting against attacks that use falsified identities. Confidentiality is another challenge, where access to certain types of information is restricted to protect the network's sensitive data. As a result, developing effective security policies for VANETs is essential to ensure the network's integrity and prevent unauthorized access, eavesdropping, and malicious attacks that could compromise the confidentiality of the information exchanged.

5. Problematic of thesis

The problematic of this thesis is securing VANET from attacks. The primary concern revolves around the vulnerability of VANETs to various forms of attacks, stemming from their unique characteristics such as the high mobility of nodes, the absence of centralized control, and the utilization of an open wireless communication medium.

One of the main challenges is that attackers can launch DoS attacks by flooding the network with a high volume of traffic, which can lead to network congestion and a high packet loss rate, making the network unusable [41,42]. Another challenge is that most of the proposed solutions such as encryption and decryption focus on detecting and preventing attacks from outside the network while neglecting the possibility of attacks by authentic members of the network [7].

Moreover, many of the proposed solutions require extensive computing resources, which can be challenging to implement in the resource-constrained VANET environment [43]. Additionally, there is a lack of studies that investigate the effectiveness of combining different security mechanisms to enhance the overall security of VANETs.

Therefore, this study aims to address these challenges by proposing a novel solution for securing VANETs from different attacks that is efficient, effective, and compatible with the resource-constrained VANET environment. This proposed solution aims to mitigate not only external attacks on the network but also the potential risks posed by internal threats. To achieve this, we advocate for the integration of two robust security mechanisms: trust-based systems and blockchain technology. By combining these mechanisms, we can significantly enhance the overall security of VANETs.

6. Objectives

The aim of this thesis is to improve the security and reliability of communication in vehicle ad-hoc networks by proposing a new secured routing protocol that can mitigate vulnerabilities and attacks in VANETs. The objectives of our research are:

- Review and analyze the existing vulnerabilities that VANETs are susceptible to, with a particular focus on attacks.
- Evaluate the effectiveness of the existing security solutions within vehicular networks in mitigating attacks and identify their limitations.
- Propose a new secured routing protocol that can provide a more secure and reliable communication infrastructure for vehicular networks.
- Implement and test the proposed secured routing protocol using simulation tools and compare its performance with other existing secured protocols.
- Assess the practicality and feasibility of the proposed protocol in real-world scenarios and identify any potential issues or challenges in its implementation.

7. Contributions

Our research endeavors have led to three significant contributions, each addressing pivotal aspects of VANET security:

- **Contribution 1 –** Our study commences with a thorough survey that delves into the multifaceted perspectives of routing security within VANETs. This comprehensive analysis not only reviews existing research but also meticulously identifies and examines critical research gaps and challenges that plague these networks. We evaluate a spectrum of existing solutions in light of these gaps, providing a comprehensive view of the state-of-the-art solutions and their limitations. This survey serves as a foundational resource, shedding light on the intricate challenges that we subsequently tackle in our research.
- **Contribution 2 –** Recognizing the importance of empirical evaluation, we introduce a novel simulation model designed to scrutinize the performance of various security solutions in the unique environment of VANETs. This model offers an invaluable tool for comprehensively assessing the effectiveness of security protocols, allowing us to make informed decisions about their suitability in VANET scenarios.
- **Contribution 3 –** The heart of our contributions lies in the development of the Secure and Efficient routing protocol for Vehicle-to-Everything, abbreviated as SecE-V2X. This innovative protocol represents a groundbreaking achievement in the realm of VANET security. By seamlessly integrating blockchain technology and trust mechanisms, SecE-V2X ensures secure, authenticated, and trustworthy routing in VANETs. Our extensive evaluations and simulations demonstrate how this protocol significantly mitigates security threats and enhances the overall safety of vehicular networks.

8. Thesis structure

This thesis consists of five chapters, organized as follows:

- **The second chapter:** In this chapter, we delve into the topic of security in VANETs. We provide an overview of general security services concepts and classify various attacks known to occur within this network category. Additionally, we present various proposed solutions that have been developed to mitigate these attacks.
- **The third chapter:** This chapter introduces our new routing protocol SecE-V2X which aims to enhance security in VANETs. We present a detailed description of our proposed protocol and outline its main features and advantages. Then, we discuss the security analysis of the proposed protocol.
- **The fourth chapter:** This chapter of our thesis provides an implementation of the proposed protocol, test, and in-depth analysis of the data collected from our

simulations and experiments. We present and discuss the outcomes of our experiments, including the performance of our proposed secured routing protocol in different VANET scenarios. Moreover, we compare our results with those of other secured routing protocols to evaluate the effectiveness of our protocol.

- **In the last chapter:** we summarize the main contributions of our research and highlight the key findings of our study. We restate our research objectives and discuss how our work has addressed the research questions that were posed and emphasize the significance of our findings for the VANET research community. The chapter also includes a comprehensive discussion of the challenges and limitations encountered during the research process. Finally, we conclude the chapter by summarizing the main contributions of our research and providing recommendations for further investigation.

Chapter 2

Literature Review on VANET Security

Content

1. Introduction

The emergence of VANETs aligns with the progress of the automotive industry, where vehicles are now equipped with radars, sensors, and cameras to enhance passenger safety and driving comfort. These advanced vehicles can gather data from their local sensors or other vehicles on the road, which can be utilized to develop a variety of applications such as traffic prediction, the mean velocity on a road segment, detection of dangerous areas, and parking space availability. The role of VANET technology will be vital in the domain of road services in the near future, decreasing risks on the road through the transmission of warning messages, providing access to various applications that enhance passenger comfort, and optimizing traffic flow [22].

By employing safety applications for traffic management, VANET has the potential to significantly reduce road accidents. However, the high speed of vehicles and the dynamic nature of the network topology result in short-lived connections between nodes (vehicles and RSUs). This implies that application implementation needs to consider different time constraints and security configurations. Security plays a vital role in network systems, and with the emergence of new attacks in various networks, including VANET, a comprehensive security system is essential to mitigate these threats. VANET is susceptible to various attacks such as black holes, warm holes, denial of service, and more [15].

To address these challenges, the proposed security algorithms should be robust enough to meet the requirements and effectively counter different types of attacks. For instance, in the event of an accident, the affected vehicle or neighboring nodes (vehicles or RSUs) to notify rescue services like civil protection and ambulance services send alert messages. The timely transmission of these alert messages is crucial to ensure prompt assistance. However, achieving this within a short timeframe becomes challenging in the presence of malicious nodes.

In this chapter, we will examine the concepts and problems related to the issue addressed in this thesis. We will begin by defining security services in VANET, categorizing some known attacks, and discussing various proposed approaches in the security field of VANET. Finally, the chapter will be concluded.

2. Overview of VANETs Security

In their work, the authors in [44] raise a crucial question about the security of VANETs. They ask whether hackers can exploit the wireless network of intelligent vehicles to cause accidents, thereby emphasizing the need for automakers to prioritize VANET's security. As safety is paramount in

VANETs, ensuring the integrity and non-alteration of critical information by malicious individuals is of utmost importance. Therefore, securing VANET systems should be capable of determining driver responsibility while preserving their privacy. In order to guarantee the effective operation of intelligent transportation systems, it is essential to ensure the security and protection of communication, information flow, and vehicle/driver data within the vehicular network.

2. 1. Security requirements

Security requirements are the essential attributes that must be included in any security solution to guarantee that the system is safeguarded and shielded against potential threats and attacks. After examining different security aspects of VANET to ensure a secure network for intelligent transportation systems, it is necessary to fulfill the following requirements:

- **Confidentiality:** It is a security principle that ensures that sensitive information or resources are only accessible to authorized individuals while keeping them protected from malicious activities, scams, and piracy. This means that the data is kept secret from unauthorized access and only those with the necessary permissions can access it [45].
- **Availability:** It is a critical aspect of security that ensures that resources and services are always accessible to authorized users whenever they require them. In vehicular ad hoc networks, this is particularly important, especially in safety applications where fast response is required to avoid accidents or other dangerous situations. In a denial-of-service (DoS) attack, vehicles may refuse to provide services, which is unacceptable in safety applications. Even a few seconds of delay in providing the necessary information can be detrimental to the overall security of the system. Therefore, it is essential to have alternative scenarios in place to ensure that the requested services and resources are always available to authorized users, even in the event of a DoS attack or other similar threats. By ensuring the availability of services and resources, we can maintain the overall integrity and functionality of the vehicular ad hoc network, as well as minimize the risks of accidents and other dangerous situations [46].
- **Integrity:** Integrity is an essential security requirement in VANETs, which refers to maintaining the accuracy and consistency of data transmitted between vehicles or between a vehicle and a roadside unit. Data integrity ensures that the data packets sent by the sender are not modified or tampered with by an attacker during transmission. This is crucial to maintain trust between vehicles and the overall functioning of VANETs. Data accuracy is especially important in safety applications, where even a slight modification of data can lead

to disastrous consequences. To ensure integrity, various security measures can be employed such as digital signatures, public key infrastructure, and cryptography revocation mechanisms. These measures can protect data packets from being tampered with and provide an end-to-end secure communication channel between the sender and receiver [45].

- **Authenticity and Privacy:** Authentication ensures that messages sent over the network are from legitimate sources and not from an intruder or an attacker. To maintain privacy, VANETs use pseudonyms to protect the identity of users. Pseudonyms are temporary identities used by vehicles to communicate with other vehicles and infrastructure without revealing their true identity. This ensures that users can communicate and interact with each other anonymously, thereby protecting their privacy. However, it is important to ensure that pseudonyms are not misused by attackers to launch attacks or compromise the security of the network. Therefore, authentication plays a critical role in ensuring the security and privacy of VANETs [47].

- **Non-repudiation:** This security feature ensures that the originator of a message cannot deny having sent that message. It is an important security requirement in VANETs, particularly in applications where the information being transmitted is critical and must be accurately identified. Non-repudiation ensures that messages are authenticated and that their contents are tied to the sender. In VANETs, digital signatures are often used as a mechanism to provide non-repudiation services. By providing non-repudiation, the risk of malicious nodes injecting additional information into transmitted messages is eliminated, thereby increasing the overall security of the system [48].

- **ID traceability:** In VANET, this feature refers to the ability to trace the real identities of vehicles that send messages within the network. This feature is important in cases where accountability is necessary, such as identifying the source of malicious messages or accidents. It allows for the tracking of vehicles that violate traffic laws or are involved in accidents and helps in resolving disputes that may arise from such incidents. However, Identity (ID) traceability must be balanced with privacy concerns, and the use of pseudonyms or other anonymization techniques can be employed to protect the privacy of drivers while still allowing for traceability in certain situations [45].

- **Scalability:** It is an important aspect of VANETs as the number of vehicles communicating within the network can grow quickly. The goal of scalability is to ensure that the network can handle an increasing number of vehicles without causing any

disruption or loss in data transfer and network performance. The administrative complexity of managing a large number of vehicles also increases with the network's size, which can cause issues if not addressed properly [47].

2. 2. VANET constraints and security impact

Several security challenges are presented by the characteristics and features of VANETs, which can have an impact on the implementation of security measures for establishing secure communication in V2V and V2I. Below are some of the security challenges that we have identified in VANETs:

- **Wireless Link Use:** Wireless links in VANETs are also subject to interference and noise, which can lead to transmission errors and packet loss. These issues can affect the performance of security mechanisms and the overall reliability of the network. Moreover, the mobility of vehicles in VANETs causes frequent changes in the network topology, which can result in unstable wireless links and challenges in establishing secure communications. Therefore, VANET security mechanisms should be designed to be adaptable to dynamic network conditions and to provide seamless communication despite wireless link instability [47].
- **Network Scale:** This challenge is a significant issue in VANETs due to the potentially large number of vehicles involved. Cryptographic key distribution can pose a challenge for security measures due to the difficulty and time-consuming nature of distributing keys to a large number of vehicles. As the number of vehicles in the network increases, the complexity of managing the keys increases as well. Therefore, a robust and scalable security system is essential for VANETs to maintain their functionality and ensure the confidentiality of communications. Any changes in the number of vehicles communicating in the network should be taken into account in the design of the security system to ensure scalability. A carefully studied and planned approach to VANET deployment is necessary to overcome this challenge [48].
- **Network Volatility:** Network volatility refers to the unpredictable and short-lived nature of communications between vehicles in VANETs. The connection between two vehicles can be established for a brief period of time, and then it may be terminated due to the vehicles' movement. This presents a challenge for security measures that rely on long-lived contexts, such as verifying identities, as it is difficult to maintain such contexts in VANETs. Moreover, in vehicular networks, link disruptions between vehicles are a prevalent incident, particularly when vehicles are traveling in opposite directions, resulting in frequent network

disconnections. Therefore, VANETs require security solutions that are designed to handle the transient nature of network connections and can quickly adapt to the network topology changes [49].

- **Heterogeneity:** Heterogeneity is a challenge in VANET security as the vehicles that make up the network can support different types of applications and have varying equipment capabilities. This diversity poses a challenge for implementing security measures that can function across a range of equipment and applications. Moreover, secure mechanisms must be applied to these various applications without compromising network efficiency and scalability. This challenge can be addressed through the development of security measures that are flexible and adaptable to different equipment and applications, while also maintaining the necessary level of security [47].
- **Delay constraints:** Sensitive delay is a significant challenge for VANETs, especially for safety-critical applications that require real-time responses. Meeting these time constraints is crucial to avoid catastrophic consequences, such as accidents or delayed rescue operations. However, ensuring real-time communication is not always easy, as the network may suffer from delays or packet losses. Moreover, the real-time requirements make the applications more vulnerable to Denial of Service (DoS) attacks, which can lead to significant delays in the network. To address these issues, VANETs need to be designed with robust mechanisms that can provide real-time responses while ensuring secure and reliable communication. One approach is to focus on preventing attacks, rather than detection and recovery, since this can reduce the impact of the attack on the network's real-time response [49].
- **Multi-hop connection:** It is the communication mode where a vehicle sends messages to a set of neighboring vehicles, which then pass on the messages to their own set of neighbors, and so on, until the messages reach their intended destination. This mode of communication is used when the intended receiver is out of range or not in the line of sight of the sender. However, multi-hop communication also introduces security challenges as the messages pass through multiple vehicles, making it challenging to maintain the confidentiality, integrity, and authenticity of the messages. Additionally, misbehaving or malicious vehicles can disrupt communication by intentionally dropping or modifying the messages, leading to a denial of service or unauthorized access to the network. Therefore, the behavior of vehicles in the network must be monitored, and security mechanisms

should be in place to isolate and punish any misbehaving or malicious vehicle to maintain the security and reliability of the multi-hop communication [47].

- **Cooperativeness:** Cooperativeness is a key aspect of VANETs, where the vehicles are expected to cooperate with each other to share information and disseminate it to other vehicles in the network. However, this aspect also poses significant security challenges as it can be vulnerable to attacks such as bogus information attacks. In such attacks, an attacker can inject false information into the network, which can be spread and further propagated by cooperative vehicles, leading to incorrect decisions and unsafe situations [49].

3. Attacks and menaces

Most vehicles and drivers are typically assumed to be trustworthy in VANETs. However, there are malicious vehicles or attackers that intentionally misbehave for various selfish reasons. These attackers can launch different types of attacks to disrupt the normal functioning of the network, compromise the privacy of the users, or cause harm to the users themselves. It is essential for researchers to understand the motives behind such attacks and develop appropriate security solutions to protect the network from these malicious actors.

3. 1. Adversaries and attackers

The security of a vehicular network is heavily dependent on understanding the nature of adversaries. An adversary is a compromised entity that utilizes various techniques to successfully breach the security of honest vehicles in order to achieve its goal. The adversaries can be broadly categorized into two classes [50]:

- **Selfish drivers:** They are motivated by their own self-interest and may misbehave to maximize their driving profit. For example, they may send false information about congestion ahead to divert other vehicles from their route and gain an advantage.
- **Malicious adversaries:** They pose a greater threat to the system and can cause serious damage to other drivers. They may intentionally tamper with messages and provide false information or cheat the system to gain more resources. In the worst-case scenario, they may attempt to sabotage the network by compromising the Roadside Units (RSUs).

It is crucial to understand these types of adversaries to develop effective solutions to secure the vehicular network.

3. 2. Attacks in VANETs

The emergence and proliferation of VANETs have led to the appearance of various attacks. This section explains the most common attacks that can be carried out on routing protocols. While certain attacks in vehicular ad hoc networks draw from existing attacks targeting mobile ad hoc networks, there are also distinct attacks that are specific to vehicular ad hoc networks. Attacks that are unique to certain routing protocols are not included in this study, as they are not discussed in the literature.

3. 2. 1. Denial-of-service attack (DoS)

DoS attacks aim to disrupt the availability of network resources by overwhelming a system with a flood of high-frequency signals, more requests than the system can handle, or other methods that render the system unable to perform its valid activities. In vehicular ad hoc networks (VANETs), DoS attacks are typically launched near roadside units (RSUs) to block communication between vehicles and RSUs, preventing access to network services such as sending and receiving safety or non-safety messages. Another type of DoS attack, known as Distributed DoS (DDoS), is more severe, where the attacker launches attacks from different locations simultaneously. There are various types of DoS attacks in VANET, with the malicious node launching the attack in different ways [46].

- **JellyFish attack:** The JellyFish attack is a type of DoS attack that targets both control and data protocols. It is difficult to detect and protect against as it conforms to protocol norms, making it seem like normal network traffic. The attacker aims to degrade network performance by dropping, delaying, or reordering packets, which causes congestion and reduces throughput for all flows. There are three sub-categories of JellyFish attacks, including the JellyFish Reorder Attack, which reorders packets to cause delays and congestion. The JellyFish Periodic Drop Attack drops packets in a periodic manner, leading to intermittent connectivity and degraded network performance. The JellyFish Delay Variance Attack creates variance in packet delay, leading to reduced throughput and increased latency [15].
- **Flooding attack:** A flooding Attack is a type of network attack that aims to disrupt the normal network operation by overwhelming the network with a large volume of traffic. The main goal of this attack is to consume network resources and degrade network performance, causing a denial of service to legitimate users. In the context of VANETs, a flooding attack is initiated with the intention of depleting network resources, such as

bandwidth, and draining the resources of individual nodes, including battery power. One of the main targets of flooding attacks in VANETs is the routing protocol. By broadcasting multiple route request packets for a destination that does not exist in the network, the attacker can create congestion and consume network resources, leading to a denial of service for legitimate vehicles. This type of attack has a significant impact on on-demand routing protocols, which rely on the exchange of control messages to discover routes [51].

- **Jamming attack:** A jamming attack is a form of DoS attack that is executed at the physical level. It involves transmitting a signal to disrupt the communication channel between the sender and the receiver intentionally. This results in the reduction of the signal-to-noise ratio for the receiver, which eventually leads to the denial of service. To perform a successful jamming attack, the attacker needs to jam the communication channel at the same time that the useful signal is being transmitted, and they need to choose the most effective signal transmission model that can merge with the receiver's signal the best. If the jamming attack is successful in a Vehicular Ad Hoc Network, it can have serious consequences such as causing accidents due to the inability of vehicles to communicate with each other, leading to a lack of cooperation between vehicles, or even leading to the failure of safety-critical applications[47].

- **Blackhole attack:** This attack is a type of DoS attack that targets the routing protocol in a network. The attacker exploits vulnerabilities in the routing protocol by advertising false routes to the destination. The attacker broadcasts false routing information, making other nodes believe that the attacker has the best path to the destination node or network resource, and as a result, victim nodes send their traffic to the attacker instead of the actual destination. Once the attacker receives the traffic, it simply discards it, resulting in a DoS [52].

3. 2. 2. Bogus information attack

A Bogus Information Attack is a kind of attack in which an attacker node generates and disseminates false information in a vehicular ad hoc network. This attack aims to deceive other nodes in the network and change their behavior according to the attacker's intentions. The malicious node generates fake messages about its environment and broadcasts them to the network, causing confusion and possible disruption of communication. The impact of this attack can be significant, as drivers may change their route based on false information, leading to traffic congestion, delays, or even accidents. To mitigate this type of attack, secure message authentication

and verification mechanisms can be employed to ensure the authenticity and integrity of messages [49]. We can identify different types of bogus information attacks, which include:

- **Replay attack:** In this attack, an attacker stores a message and then resends it at a later time to mislead other network entities. In VANETs, replay attacks can be particularly dangerous as they can lead to the dissemination of false or outdated information, which can have serious consequences for road safety. The main goal of a replay attack is to exploit the circumstances when the original message was transmitted. The attacker captures and stores information circulating within the network, subsequently replaying it at a later time, even if it is no longer accurate or valid. For instance, the attacker may retain a received message concerning a past accident or traffic incident and later resend it. As long as the message remains valid, the attacker can exploit it to deceive others [23].
- **Sybil attack:** This attack involves a malicious node acquiring multiple identities within the network, by either stealing or creating them. The attacker then utilizes these identities to masquerade as a group of nodes, tricking other nodes into believing that they are receiving messages from several legitimate vehicles. Geographical routing is particularly vulnerable to this type of attack since the attacker can disseminate false location information, leading to misinformation about events across different network positions. The attacker's objective is to manipulate the network's behavior, such as redirecting a group of vehicles onto an alternative route for their own gain. Another instance of a Sybil attack is the Node Impersonation Attack, wherein a malicious vehicle involved in a traffic accident alters its identity and pretends to be a moving vehicle, transmitting inaccurate information about road conditions to the network [53].
- **False position information:** This is a type of attack where the attacker manipulates its location information to deceive other vehicles in the network. In this attack, the attacker broadcasts false information about its location to other vehicles in the network, which can result in misleading information being disseminated. The attacker can deceive the other vehicles by broadcasting its location as being at a different position in the network than its actual location. As a result, the other vehicles may make decisions based on this false information, such as changing their route or speed. False position information attacks can be executed through various techniques, such as altering GPS signals, replaying previous messages with false position information, or hacking the position sensor of the vehicle. This attack can have severe consequences in VANETs, as it can cause accidents or congestion, leading to a decrease in the overall efficiency of the network [54].

- **Sensor tampering attack:** An attacker physically alters the sensor data collected by an On-Board Unit (OBU) or RSU, leading to the generation and dissemination of incorrect information in the network. For example, an attacker may tamper with a speed sensor, causing it to report a higher speed than the actual speed of the vehicle. This can cause other vehicles in the vicinity to make incorrect decisions about their own speed and maneuvering, leading to accidents or other safety issues. Similarly, an attacker could tamper with other sensors such as the GPS or environmental sensors like temperature or humidity, which can lead to false information being disseminated about the vehicle's location or the environment, respectively [49].

3. 2. 3. Eavesdropping attack

An Eavesdropping attack is a type of passive attack that is carried out by an attacker who secretly intercepts and monitors the communication between two entities in the network. This attack is also known as a sniffing or wiretapping attack. In the context of VANETs, this can be achieved by intercepting wireless signals transmitted between two vehicles or between a vehicle and an infrastructure node. The attacker gains unauthorized access to confidential information, which may include the location and actions of a vehicle, as well as the content of messages exchanged between the vehicles. Eavesdropping attacks can be very difficult to detect because the attacker does not actively modify the communication or inject any malicious code into the network. The attacker simply listens to the communication and captures the data [46].

3. 2. 4. Wormhole attack

It is a type of attack in VANETs where two or more attackers collaborate to create a shortcut between two distant points in the network by tunneling packets between them. The attackers then replay the packets at the other end of the network. The aim of this attack is to manipulate the logical topology of the network, collect and modify a large amount of traffic. The attackers use the shortest path or best route between the two points as the tunnel, and then replay the packets to deceive nodes into believing that the malicious nodes have the best paths. This can cause the deceived nodes to choose the malicious nodes as the best path for their communications. The availability, confidentiality, and integrity of the communication channel between two legitimate nodes might be compromised by wormhole attacks. In addition, it can also disrupt the routing mechanism of the network, leading to packet loss, network congestion, and network performance degradation [51].

3. 2. 5. Repudiation attack

A repudiation attack is a type of security attack where an individual or entity denies having performed an action, transaction, or communication. In other words, a repudiation attack allows an attacker to perform some action while making it appear as if it was not carried out by them. This type of attack can be particularly damaging in situations where the authenticity of actions is crucial, such as in financial transactions or legal contracts. Repudiation attacks can be challenging to detect and prevent, as they often involve manipulating or altering logs or audit trails [47].

3. 2. 6. The man in the middle attack

An attacker in this attack intercepts a communication between two parties that they believe are directly communicating with each other. In a VANET, this type of attack involves an attacker vehicle that inserts itself between two communicating vehicles and controls the communication. The attacker can modify, intercept, and forge messages between the two victims, without them realizing that their communication has been compromised. The goal of this attack is usually to bypass authentication and integrity mechanisms or to perform eavesdropping. For example, the attacker can pretend to be a legitimate RSU and intercept communication between a vehicle and the actual RSU, thus gaining access to sensitive information such as private messages [48].

3. 3. Classification of Attacks in VANETs

Numerous studies have been conducted by researchers to examine the attacks on VANETs. Categorizing these attacks can be beneficial since the unique features of VANETs make them susceptible to vulnerabilities and limitations that demand effective solutions. By dividing the attacks, we can enhance our ability to manage and prevent them.

In VANETs, the attacks can be categorized based on different criteria. In [47], the authors categorized the attacks based on attack network layers. The authors in [45] indicate four criteria: Security Services; Attacker Type; VANET Layers; VANET Components. According to attacker objectives, the authors in [55] classified the attacks into three categories: communication-controlling attacks, communication-preventing attacks, and selfish attacks. Based on the target location of the attackers, in [49], the authors classified the attacks into two groups: Inter-vehicle attacks and Intra-vehicle attacks.

The location of attacks can vary across different layers of networks, and they can have different targets such as controlling communication, preventing vehicles from connecting or acquiring

resources for the attackers. Therefore, in this thesis, we categorize attack models in vehicular networks based on the attacker's genre, objectives, targeted service, and network layers.

3. 3. 1. Classification Based on attacker genre

Depending on the attacker genre, VANET attackers may be classified based on several factors such as the level of participation and harm caused. The first factor is the level of participation, which distinguishes attackers as active or passive. Active attackers are directly involved in carrying out the attack, whereas passive attackers monitor the information without taking an active role[56]. The second factor is whether the attacker is an insider or outsider of the network, also known as an internal or external attacker. Internal attackers have complete information about the network, while external attackers lack information about the network structure [57]. The third factor is the motivation behind the attack, which distinguishes attackers as rational or malicious. Rational attackers carry out an attack for financial gain or personal reasons, while malicious attackers carry out attacks without any personal benefits [58].

3. 3. 2. Classification Based on Attacker Objectives

In this classification, there are two main categories: communication controlling attacks and communication preventing attacks. Communication-controlling attacks are designed to collect and modify the information exchanged in the network to gain control over communication. This can be done in various ways, such as convincing other vehicles to take alternative routes, suppressing packets of warning traffic jams or injecting fake warnings to mislead other vehicles. The ultimate goal of communication-controlling attacks is to gain an advantage over other vehicles and to gain control over communication [59].

On the other hand, communication-preventing attacks target the availability of network services, such as Denial of Service attacks. DoS attackers try to block the principal communication medium, making it unavailable to other vehicles. This type of attack can have serious consequences, as it can completely disrupt the communication in the network, making it impossible for vehicles to exchange vital information. In summary, the attacker's objectives play a critical role in determining the type of attack they will launch and the potential impact it can have on the network [59]. Table 2.1 below, shows a comprehensive list of the attacks mentioned earlier along with their respective objectives.

Table 2.1. Classification of attacks based on attacker objectives

Attacks	Attacker objectives	
	Communication preventing	Communication controlling
DoS	√	
Jellyfish	√	
Flooding	√	
Jamming	√	
Black hole	√	
Replay		√
Sybil		√
False position information		√
Sensor tampering	√	√
Eavesdropping		√
Wormhole	√	√
Repudiation		√
The man in the middle		√
Message tampering		√
Traffic analysis		√
Impersonation		√

3. 3. 3. Classification Based on security services

In network security, various types of attacks can occur that aim to compromise the security services. The attacks can be categorized based on the specific security service that is being targeted as follow [45]:

- **Integrity Attacks:** These attacks aim to compromise the integrity of the data being transmitted in the network. The attacker can modify, delete, or inject false data to disrupt the communication or to mislead the receiving end.
- **Confidentiality Attacks:** These attacks aim to compromise the confidentiality of the data being transmitted in the network. The attacker can eavesdrop on the communication and collect sensitive information that can be used for malicious purposes. An example of such an attack is the eavesdropping attack.
- **Availability Attacks:** These attacks aim to compromise the availability of network services. The attacker can launch a DoS attack to overload the network with traffic, making it unavailable for legitimate users.

- **Authentication Attacks:** These attacks aim to compromise the authentication mechanism of the network. The attacker can impersonate a legitimate node to gain unauthorized access to the network or to perform malicious actions. One such attack is the Sybil attack mentioned earlier.
- **Non-repudiation Attacks:** These attacks aim to compromise the non-repudiation service of the network. The attacker can deny the authenticity of the data being transmitted or refuse to accept responsibility for their actions. An example of such an attack is the repudiation attack.
- **ID Traceability Attacks:** These attacks aim to compromise the traceability of the nodes in the network. The attacker can create fake identities or modify the legitimate ones to hide their real identity and perform malicious actions. One such attack is the impersonation attack.

Table 2.2 shows the classification of attacks based on the compromised security service.

Table 2.2. Classification of attacks based on the compromised security service

Attacks	Compromised services					
	Integrity	Confidentiality	Availability	Authentication	Non-repudiation	ID-Traceability
DoS			√			
Jellyfish			√			
Flooding			√			
Jamming			√			
Black hole			√			
Replay	√			√		
Sybil				√		√
False position information				√		
Eavesdropping		√				
Wormhole	√	√	√			
Repudiation					√	
The man in the middle	√	√				
Message tampering	√					
Traffic analysis		√				
Impersonation						√

3. 3. 4. Classification Based on Network Layers

In vehicular networks, attacks can take advantage of vulnerabilities in different layers of the network, allowing attackers to gather sensitive information or control the network. For instance, attackers can extract users' sensitive information from their exchanged messages with service providers or inject false traffic information into the network at the application layer. Similarly, attackers can modify the content of their broadcast messages to impersonate a priority vehicle. At the network layer, attackers can manipulate information about the positions and locations of vehicles to hide their true identity or divert traffic to their desired destination. They can also broadcast fake information to appear as a reliable relay candidate in multi-hop communication. At the Medium access control (MAC) and physical layers, attackers can flood the channel with useless traffic to disrupt communication and prevent legitimate communication [47]. In Table 2.3, we provide a list of attacks according to different network layers.

Table 2.3. Classification of attacks based on network layer

Attacks	Compromised layer				
	Physical	MAC	Network	Transport	Application
DoS	√	√	√	√	√
Jellyfish			√		
Flooding			√		
Jamming	√	√			
Black hole			√		
Replay		√	√	√	√
Sybil		√	√	√	√
False position information	√				√
Sensor tampering	√				
Eavesdropping	√				
Wormhole			√	√	
Repudiation					√
The man in the middle				√	
Message tampering	√	√	√	√	√
Traffic analysis					
Impersonation		√			√

4. Security solutions in VANETs

To ensure the security of vehicular networks and tackle the security issues mentioned earlier, various solutions have been proposed using different methods. In this section, we will discuss the security architecture, followed by a review of the existing works related to security solutions in the field of VANETs.

4. 1. Security architecture

In order to achieve comprehensive security in a system, the authors in [60] proposed a security architecture consisting of five levels: material, authentication, trust, message, and cryptographic levels. In this thesis, the security architecture can be categorized into three parts (See Figure 2.1): the first part is focused on prevention, which includes security material and authentication levels; the second part is concerned with detection and correction, which involves trust level and message/data level, and the third part is about privacy, which is related to the cryptographic level.

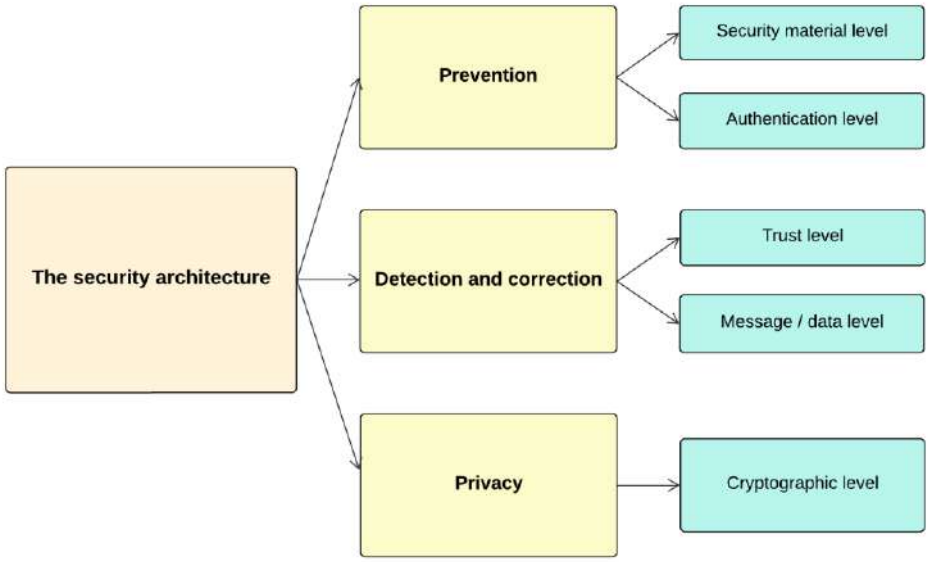

Figure 2.1. The proposed security architecture

- **The prevention:** In this part, the material level of the security architecture is the foundation for ensuring security in vehicular networks. At this level, the security concerns are focused on the physical resources, such as the on-board units (OBUs), GPS receivers, radars, Event Data Recorders (EDRs), antennas, and other hardware devices that are part of the vehicular network infrastructure. To address the security concerns at the material level, the Trusted Platform Module (TPM) specifications can be used. A TPM is a hardware component that is designed to protect data and store it in shielded locations. On the other

hand, the authentication level encompasses various forms of authentication. Firstly, the system considers users' authentication to prevent unauthorized access by individuals who are not authorized to communicate within the network. Secondly, messages are authenticated to ensure that receivers can verify whether the received message was sent by the appropriate entities and that it was not altered while in transit. At this level, equipment authentication is also verified to avoid false nodes or fake equipment. Finally, location authentication is necessary to verify the sender's position. Since these authentications are carried out in stages throughout transmission, packets may be refused at any stage of the communication process. This makes it easier to maintain the system's access control requirements.

- **The detection:** This phase encompasses two levels. The first level, known as the trust level, focuses on establishing a system that ensures the reliability of nodes within the network. This involves implementing mechanisms such as trust systems, plausibility checks, or reputation systems. The reputation system evaluates a node's trustworthiness based on factors like speed, position, acceleration, and feedback from other users. If the node receives positive ratings, it is permitted to communicate, while negative ratings result in the node being expelled from the network. The plausibility check verifies information based on the event, while the trust system utilizes techniques like the Trusted Platform Module (TPM) to ascertain the trustworthiness of nodes. Furthermore, this level addresses the non-repudiation requirement by collecting sufficient information about the sender's node to validate its identity as the author of a message. At the second level, the focus shifts to the message or data itself to ensure its security. The literature suggests the use of digital signatures, and the concept of a vehicular Public Key Infrastructure (PKI) is proposed. In this PKI, each vehicle possesses its own set of public/private key pairs for signing safety broadcast messages. Even if a malicious node manages to bypass the authentication and trust levels, the transmitted message is still subject to verification to guarantee the integrity of the information. This level plays a critical role in ensuring the authenticity and integrity of the received messages, protecting against tampering or anomalies.

- **Privacy:** This part is about the cryptographic level, which is concerned with the privacy of users in the network. It involves the use of privacy solutions such as private/public keys or anonymous identity protocols to ensure that the users' privacy is protected. This level ensures that information is encrypted before being transmitted. Confidentiality of the information must be maintained at this step. The main objective of this level is to prevent

eavesdropping, which is a security threat to the network. Cryptography is used to provide confidentiality, integrity, and authenticity of the data.

4. 2. VANET security solutions taxonomy

After conducting a literature review, several researchers have proposed solutions to address the aforementioned attacks. Based on our analysis, there are four primary aspects of VANET security that serve as the main categories in our classification. These categories are described in the following subsections.

4. 2. 1. Reputation and trust solutions

Reputation and trust-based systems have emerged as significant approach to ensuring security in VANETs. Reputation, in simple terms, refers to the gathered information from other nodes concerning an entity. This information assists in making a decision about the trustworthiness of that entity. Since network nodes are often limited by their sensing range, the reputation feature is essential in determining the trustworthiness of entities outside their range. There are several reasons why reputation and trust-based systems are preferred as security solution for wireless networks. One reason is the individuality of nodes in VANETs. Additionally, there is a lower cost associated with producing nodes in Wireless Sensor Networks (WSNs) compared to other types of networks. Reputation-based systems also provide a viable security solution, especially with the limitations of cryptography in dealing with internal attacks. The reputation and trust-based systems consist of four steps, which are as follows: collection of information, dissemination of information, modeling of information, and making decisions based on the information collected [61]. These components have been illustrated in Figure 2.2.

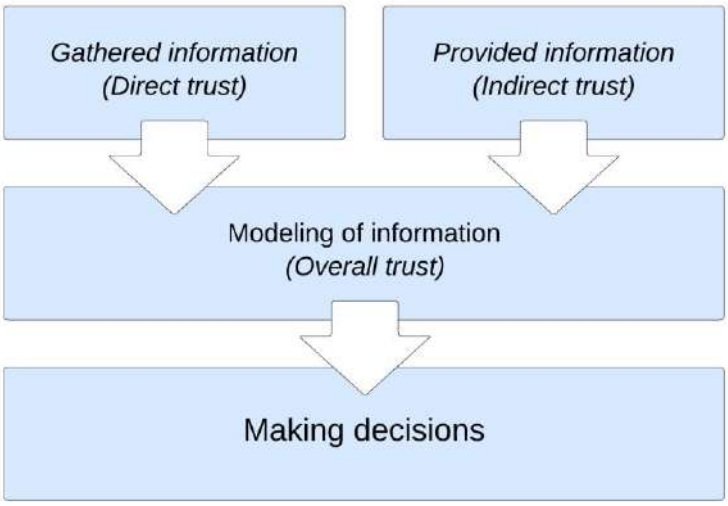

Figure 2.2. Components of a reputation-based system

Numerous studies have been conducted using reputation systems, and some of these studies will be examined in accordance with the suggested system.

Buchegger and Boudec [62] proposed a system called CONFIDENT, which stands for Cooperation Of Nodes: Fairness In Dynamic Ad hoc NeTworks. The primary goal of CONFIDENT is to detect and exclude malicious nodes from participation while encouraging cooperative behavior. The system achieves this through a distributed and symmetric reputation system that utilizes both first-hand and second-hand information to update the reputation metric. CONFIDENT also employs a watchdog mechanism, where any node that detects a deviation from monitored behavior sends an alarm message to inform other nodes. Misbehaving nodes are excluded and blacklisted, which prevents other nodes from serving them. CONFIDENT introduces four novel components: the monitor, trust manager, reputation system, and path manager. The monitor's role is to ensure the accurate forwarding of packets, while the reputation system maintains a table with entries for node identifiers and their respective ratings. The trust manager handles the exchange of alarm messages, while the path manager is responsible for assessing and ranking paths, eliminating paths that include malicious nodes, and disregarding route requests from such nodes. While CONFIDENT offers an effective solution for networks characterized by a small number of nodes and low mobility, it may encounter scalability challenges when deployed in larger networks. Additionally, scenarios with high mobility may experience a substantial increase in overhead.

Vehicle Ad Hoc Reputation System (VARS) is a modular reputation system architecture designed for VANETs proposed by Dotzer et al. [63] and it is based on the opinion piggybacking method, where every forwarding node appends its own opinion to a message to enable confident decisions on event messages. The VARS architecture adopts an entity-centric approach, focusing on the assessment of distributed content rather than the behavior of individual nodes. Receivers have the ability to evaluate the opinions of other nodes, using them as a basis for determining the trustworthiness of a message. VARS enables each forwarding node to generate its own opinion about the data, taking into account aggregated opinions attached to the message. Peer opinion is influenced by various metrics such as direct trust, indirect trust, sender-based reputation level, and geo-situation-oriented reputation levels. However, the model lacks sufficient details regarding how sender-based information is updated, and the authors did not provide a comprehensive explanation of the indirect trust method. One drawback of the VARS system is that every node contributes an opinion on distributed content, creating a bias in favor of earlier node opinions. This can be problematic in dynamic environments where the continuous addition of overhead to packets can

lead to inefficiencies. Moreover, the scheme primarily relies on event-based messages, while beacon messages could offer valuable insights for trustworthiness determination. The authors acknowledge limitations of the VARS scheme, including the increased package overhead resulting from appending the opinions of all intermediate nodes.

The use of trust algorithms in VANETs can improve the stability of routing protocols and enhance the security of the network. In [64], a trust algorithm was proposed to handle frequent link breakages and communication overhead in reactive routing protocols. The algorithm calculates the trust metric of each node based on its link quality, distance to destination, direction, and speed. One limitation of this proposal is that it did not take into account any security requirements to choose the forwarder node, which can leave the network vulnerable to attacks. In contrast, [65] proposes a trust-based protocol that evaluates and calculates the trust value of each node based on its role, direct interactions, and recommendations of other nodes. The protocol uses an asymmetric cryptography system to ensure that each vehicle has a public key certificate containing its role, ID, and public key. The use of broadcast hello messages periodically, including the trust value of neighboring nodes, their velocity, and their position, helps to improve the trust calculation.

Kim et al. [66] propose a method to detect false information in VANETs using a message-filtering model that incorporates a threshold curve and a Certainty of Event (CoE) curve. The CoE evaluates the confidence level of a received message by considering data from multiple sources, such as local sensors, RSUs, and reputation mechanisms. The prioritization of these sources can be adjusted based on the specific event to optimize computational resources. As a vehicle approaches a genuine event, it receives an increasing number of messages reporting the event, leading to a higher CoE value. This approach ensures that a malicious actor controlling only a small portion of the network cannot deceive the vehicle, as the solution relies on the honesty of the majority. The threshold curve demonstrates the driver's indifference to the distance to the event, where the CoE value rises as the threshold value decreases. When the CoE value surpasses the predefined threshold value specific to the application, an alert message is transmitted to the driver, and the reputations of the vehicles reporting the event are elevated. Conversely, if the alert does not meet the threshold, it is disregarded, and the reputations of the corresponding vehicles are diminished. However, it should be noted that this model exclusively addresses false information attacks in the Electronic Emergency Brake Light (EEBL) application and does not encompass various other applications.

The trust-based security framework proposed by Raya et al. [67] offers an approach to assess the reliability of messages or data shared by entities in vehicular ad hoc networks, with a specific

application focus on traffic safety. This framework stands out for its utilization of multiple sources of evidence to establish trust. It adopts a data-centric approach to establish trust, where a set of multiple reports associated with a particular event, along with their respective weights, are fed into a decision logic module. The decision logic module employs various techniques such as Bayesian inference and Dempster-Shafer theory to determine the level of trust attributed to the given data. This trust level serves as an indication of whether the reported event has indeed occurred. However, one limitation of this approach is its potential unsuitability for sparse environments. In such scenarios, where there is a scarcity of reports regarding an event, it may not be feasible to repeatedly establish trust between entities. Moreover, this technique relies on the accuracy of data sensed by sensors or received from other entities. If an entity's sensors fail to accurately detect an event, the evaluation result based on that received information may not be entirely precise.

The VSRP algorithm proposed in [68] is aimed at detecting and eliminating malicious nodes from a VANET network by assigning trust values to nodes based on their behavior. The algorithm focuses on the detection of malicious nodes and aims to improve driving efficiency in congested environments. However, the algorithm has some limitations that need to be addressed. Firstly, the algorithm relies on the detection range of sensors, which is only 50m. This means that if the distance between a vehicle and a congestion location is more than 50m, the vehicle may not detect the congestion and will not choose an alternative route. This can lead to decreased driving efficiency in congested environments. Secondly, the algorithm is heavily dependent on the detection sensors, which can lead to issues if a vehicle's sensor stops working. If a vehicle's sensor fails, the vehicle may reject all messages from other vehicles, leading to further communication issues in the network.

Nai-Wei et al. [69] introduced a reputation system called the Dynamic Event-Based Reputation System (ERS) with the objective of ensuring secure communication in VANETs. The ERS determines the reliability of incoming traffic by calculating event-based reputation and confidence values. The authors introduced four functions to compute thresholds for confidence and trust. The event reputation value indicates the severity level of a specific traffic event and initially starts at zero. As a vehicle detects an event through its onboard sensors, the reputation value increments by one. The event confidence value represents the count of unique vehicles that have transmitted messages concerning the same event. When a vehicle identifies an event, the Enhanced Reputation System (ERS) utilizes a straightforward algorithm to add the received message's reputation value to the existing reputation value. The message's event confidence list is also appended. If both the event reputation value and confidence value surpass the predefined thresholds for event reputation

and confidence, the traffic event is considered genuine, prompting the vehicle to transmit a traffic warning message to neighboring vehicles. This mechanism effectively prevents the dissemination of false traffic warning messages in vehicular ad hoc networks (VANETs). Nonetheless, this system faces some challenges. Firstly, in VANETs, vehicle IDs can change over time, but the authors did not specify how to ensure that vehicle IDs remain constant during an event. Secondly, in VANETs, high vehicle speeds lead to a brief time frame for detecting a specific event. If this occurs frequently, it may reduce the system's accuracy.

Guo et al. [70] suggested a technique for assessing the behavior of a vehicle, based on encounter tickets (ER). When two vehicles successfully exchange data, they mutually send electronically signed Encounter Reports (ERs) to each other, encompassing details of the encounter like the timestamp, vehicle IDs, and unique sequence numbers. Through the exchange of ERs during encounters, a vehicle can provide reliable information about its behavior to other vehicles. Additionally, each vehicle possesses a Trust Reputation (TR) value that diminishes if it selectively discards packets. The vehicle also maintains a blacklist and a meeting list that store information about previously encountered vehicles. If a vehicle's TR value falls below a certain threshold or if it attempts to falsify or replay ERs, other nodes can identify and isolate the vehicle by adding it to their blacklists. The authors also introduced an adaptive threshold mechanism and a flexible approach for dense and sparse networks using cluster analysis. However, it should be noted that an attacker can still discard packets after attaining a positive reputation by closely following behind other vehicles.

A trust-based system is proposed in [71] as a solution for security issues in VANETs. This system takes into account the attack history and attack profiles to address DoS/DDoS attacks. Several parameters, such as delay, average latency, packet delivery ratio, and energy consumption, are considered for evaluation. Another research work by Nandy et al. [72] introduces a collaborative intrusion detection system based on trust, which also aims to provide security against DoS/DDoS attacks for availability services. The experimentation is based on Packet Drop Count (PDC), Packet Transfer Delay (PTD), and Packet Transfer Interval (PTI) parameters.

To safeguard ad hoc on-demand distance vector (AODV) routing protocols in VANETs from Jellyfish and Blackhole attacks, researchers in [73] employed an intrusion detection system (IDS) approach. They examined the AODV routing protocol with and without the IDS algorithm, comparing scenarios without attacks to those with Blackhole and Jellyfish attacks. Through simulations, they observed that the inclusion of the IDS algorithm enhanced network performance

in terms of quality of service (QoS) metrics like packet delivery ratio (PDR), throughput, and end-to-end delay when compared to network performance under attack conditions.

There have been several proposals that focus on using transmitted data to detect malicious nodes in VANETs. These proposals include the works by Harit et al. [74], Ruj et al. [75], Wasef et al. [76], Calandriello et al. [77], and Vulimiri et al. [78]. However, these approaches have their limitations. One of the main issues with these proposals is their dependence on third parties, such as central authorities, which can be a bottleneck in terms of scalability. Additionally, these proposals often require the use of authentication or other methods to identify the message provider, which can add complexity to the system. In contrast, trust models designed for VANETs must take into account the unique characteristics of this environment, including the need for scalability and the potential for nodes to be mobile and unreliable. Therefore, it is important to design trust models that are specifically tailored to the VANET environment and that can operate without the need for centralized authorities or other third parties.

4. 2. 2. Public Key Infrastructure solutions

Public Key Infrastructure (PKI) is a security method used to ensure safe and private communication in vehicular networks. This method involves the use of both symmetric and asymmetric keys, which are collectively referred to as PKI. Along with these keys, private keys and hash functions are also employed. Each vehicle in the network is assigned a private and public key pair by a Certificate Authority (CA), with the private key stored on the vehicle and the public key published in a digital certificate issued by the CA. When a vehicle sends a message, it signs it with its own digital signature, encrypted by its private key, and then encrypts the message with the recipient's public key. The recipient can then decrypt the message and verify the signature using its own private key [55]. Figure 2.3 below describes the PKI architecture. To maintain privacy and prevent exposure of private keys, PKI uses a Certificate Revocation List (CRL) to revoke the certificates of misbehaving vehicles. However, in large vehicular networks, updating the CRL on individual vehicles can be difficult, and the list can become very large. To address this issue, an RSU-aided certificate revocation (RCR) mechanism has been proposed [79], whereby RSUs generate a warning whenever a vehicle with a revoked certificate passes by, which is then disseminated to all vehicles in the network using intervehicle communication.

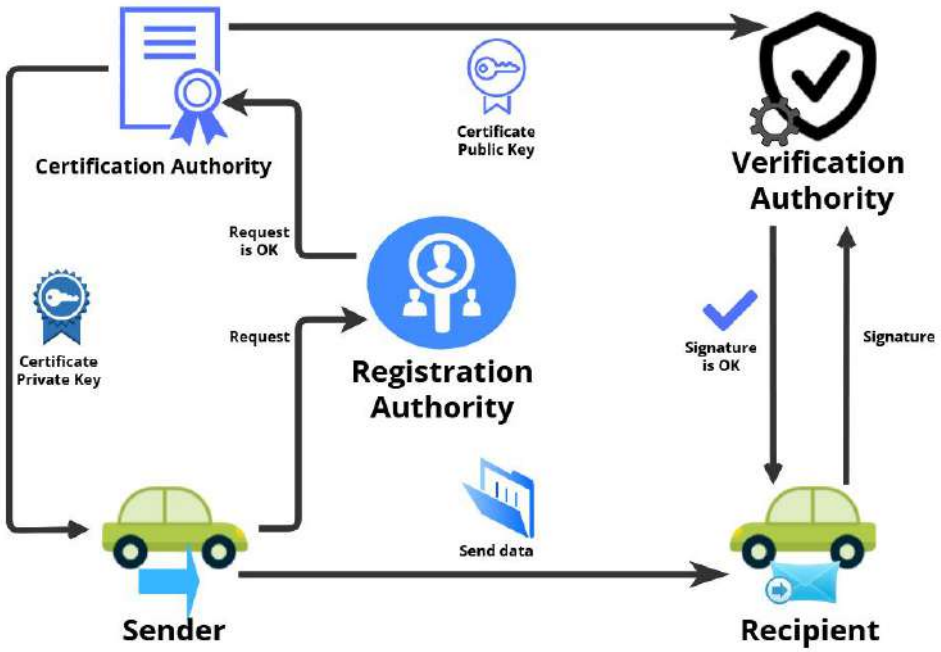

Figure 2.3. The architecture of Public Key Infrastructure solution

Despite the efficiency of PKI in securing VANETs, it has certain limitations that may lead to insecure VANET scenarios. While PKI uses anonymous certificates for identity privacy, it does not always provide location privacy. Attackers can monitor a specific vehicle between two points assigned by them and attack it by relating the anonymous certificate to that vehicle if it is moving at the same speed and in the same lane. Therefore, it is necessary to implement location privacy mechanisms for PKI to prevent such attacks [80]. Many researchers worked on proposing solutions using the Public Key Infrastructure approach, some of these proposals will be discussed as follows.

B. Dahill et al. [81] introduced a secure routing protocol called ARAN for ad hoc networks, which builds upon the AODV protocol. ARAN incorporates public key cryptography to mitigate various attacks like spoofing and tampering with routing messages. To achieve this, ARAN employs a certificate server with a publicly known key, along with timestamps to ensure route freshness. When a source node initiates a route discovery packet, each receiving node forwards the message to its neighbors, appending its own signature and certificate. The destination node responds to the first node it received the message from, and all reply messages are signed by the sender and verified by the subsequent hop. For ensuring the shortest path, the source commences with an encrypted confirmation message for the shortest path, and the destination node replies with the recorded shortest path to the source via its predecessor. Each neighbor signs the encrypted part of the message and attaches its certificate. ARAN also requires every node to maintain a routing table for each network node, and inactive routes are removed from the table. Although

ARAN demonstrates effective performance in route discovery and maintenance, it may encounter scalability issues and increased latency during route discovery due to higher packet overhead.

The authors in [82], proposed a routing protocol called Secure Efficient Ad hoc Distance Vector (SEAD) that aims to prevent multiple uncoordinated attackers from creating incorrect routing states in other nodes. SEAD is specifically designed for low-power environments and focuses on protecting against DoS attacks that can deplete bandwidth or processing time. To achieve this, SEAD utilizes efficient one-way hash functions instead of asymmetric operations. While the protocol is based on DSDV, it can be applied to other distance vector protocols as well. In SEAD, a node uses a single element from its hash chain to send a route update regarding itself, providing authentication for the minimum metric in other routing updates for that destination. Hash tree chains and packet leashes are also employed in SEAD to defend against attacks where an attacker attempts to advertise the same distance and sequence number they received. Additionally, SEAD incorporates source authentication to prevent routing loops. Although SEAD performs well compared to other distance vector routing protocols like DSDV, it has higher packet overhead, which may lead to network congestion. However, the increased number of routing advertisements enables nodes to maintain more up-to-date routing tables, thereby improving performance in highly mobile environments. SEAD is a suitable choice for networks with limited computational resources since it avoids the use of asymmetric cryptography, which typically requires additional computational power.

Y. Chun Hu et al.[83] proposed the ARIADNE routing protocol, which is an on-demand protocol based on DSR, with the objective of preventing attacks on routes and DoS attacks. ARIADNE achieves this goal by using highly efficient symmetric cryptography to authenticate and ensure the integrity of DSR signaling messages during routing discovery and route maintenance. The protocol verifies the authenticity and integrity of Route Requests (RREQs) to prevent nodes from being removed from the list and to ensure sender authenticity. Each hop verifies new information in the RREQ and the destination buffers the RREQ until intermediate nodes release the corresponding keys. To prevent intermediate nodes from removing nodes from the RREQ list, each intermediate node appends a one-way hash function to the packet header. ARIADNE also protects DSR route maintenance by requiring the authentication of every Route Error (RERR) message. The protocol prevents various DSR attacks, including routing loops, black/grey holes, and replay attacks. However, the protocol has some performance issues, such as larger signaling packets for long routes due to the increase in signaling message length with each intermediate node, and the time-delayed key disclosure which can increase the end-to-end delay of a route discovery

process, leading to a negative impact on the packet delivery ratio, particularly in highly mobile scenarios.

The security architecture presented in [84] is based on Public Key Infrastructure (PKI) and consists of functional entities that utilize long-term enrolment certificates for On-Board Equipment (OBEs) known as Bootstrap functions, as well as short-term digital certificates for pseudonym functions. The primary focus of this proposal is addressing the issue of trust. Within this architecture, V2V communication involves two types of messages: Basic Safety Messages (BSMs) and security information messages. Digital signatures and certificates are used for verification purposes in BSMs. Asymmetric encryption is implemented using the Elliptic Curve Integrated Encryption Scheme (ECIES), while the Elliptic Curve Digital Signature Algorithm (ECDSA) is employed for digital signatures to validate devices in communications between vehicles and the Security Certificate Management System (SCMS). Symmetric encryption using AES-CCM (Advanced Encryption Standard Counter with CBC-MAC) ensures confidentiality, and Message Authentication Code (MAC) guarantees integrity in communications within the SCMS, providing authenticity. This security architecture ensures privacy against both insiders and outsiders. It ensures that a single SCMS component cannot link any two certificates to the same device, and no stored information within SCMS can link certificates to a specific vehicle or owner. The Misbehavior Authority (MA) plays a crucial role in allowing only trusted nodes to continue by generating and publishing the Certificate Revocation List (CRL) and misbehavior reports in the Vehicular Ad hoc Network (VANET). The Location Obscurer Proxy (LOP) acts as an anonymizer proxy, shuffling misbehavior reports sent by OBEs to the MA. Overall, this architecture provides efficient privacy-preserving revocation mechanisms.

Rahbari et al. [85] have introduced a cryptographic technique that utilizes a Public Key Infrastructure (PKI) to detect Sybil attacks in Vehicular Ad Hoc Networks (VANETs). This approach consists of four phases and relies on the support of Road Side Units (RSUs). In the initial phase, each vehicle undergoes registration with an RSU to obtain a group authentication key, which is used for message authentication within the group. In the second phase, the RSU forwards the received message to the Local Certificate Authority (CA) because decrypting the message requires the private key of the Local CA. The third phase involves the Local CA utilizing the private key of the vehicle to verify the sender's identity after decrypting the message. To achieve this, the Local CA sends a request to the Home CA. In the fourth phase, the Home CA responds by providing the private key of the vehicle. The Local CA can then detect a Sybil attack by comparing the received reply message from the RSU. Simulation results indicate that the proposed method

exhibits low delay and is effective in detecting such attacks. However, the authors have noted that mobility issues may arise when vehicles move to regions belonging to other CAs. The authors mentioned that detecting a Sybil attack can be challenging if vehicles move to areas that are governed by other Certificate Authorities (CAs).

4. 2. 3. Identity cryptography solutions

Identity-based cryptography is a security solution that employs information that represents the user's identity to verify their digital signature. This information can include various identities like email address, network address, or user name. The technique was initially introduced by Adi Shamir[86] in 1984 but implemented practically later in 2001 by Boneh and Franklin [87]. The main advantage of using this technique is that it uses pseudonym generation, which can be modified as required for security purposes. This technique also allows users to have several pseudonyms for enhanced privacy. In VANET environments, ID-based cryptography can replace the PKI technique since it does not require the storage, fetching, and verification of public key certificates by a trusted third party in some road safety scenarios. This cryptography outperforms PKI in terms of time, communication bandwidth, and storage and also reduces the cost of CRL. Nonetheless, the main concern with ID-based cryptography is to ensure the privacy of the entity. To address this issue, irreversible algorithms must be employed to generate the pseudonym using the entity ID, while making sure that the pseudonym is only accessible to the same entity. As a result, many research studies have been carried out to tackle the privacy problem associated with ID-based cryptography [45,80]. Several important identity-based solutions are discussed here.

The authors of [88] propose a privacy protection mechanism to handle misbehaviors during VANET access. The solution uses an identity-based cryptosystem that doesn't require certificates for authentication. To ensure user privacy and traceability, a pseudonym-based scheme is employed. The privacy mechanism is based on threshold authentication, where extra authentication beyond the limit results in the revocation of the misbehaving user, except for certain types of misbehaviors such as hardware malfunctioning. The pseudonym-based privacy mechanism includes pseudonym generation and authentication, with the latter safeguarding privacy. Vehicles need to update their credentials regularly to maintain privacy. Pseudonyms hide the actual identity of vehicles to prevent neighboring vehicles and RSUs from deciphering the sender of a message. The authors show that their proposed scheme fulfills the predetermined security objectives, including privacy, traceability, non-frameability, efficient storage, and communication. However, they do not provide a mathematical proof or graph to illustrate the network performance.

Furthermore, there is no comparison made to demonstrate the efficiency and effectiveness of the scheme in relation to other existing schemes.

Hung Lu et al. [89] proposed an authentication system for VANETs that utilizes identity-based encryption and self-generated pseudonyms as identifiers to ensure privacy protection. The authentication system consists of three types: vehicle to roadside authentication, vehicle to vehicle authentication, and roadside to vehicle authentication. Prior to entering a road, a vehicle undergoes registration with a Regional Trusted Authority (RTA), which disseminates certified domain parameters for authentication and maintains hash values of registered vehicles alongside their real-world IDs. The Road Side Unit (RSU) periodically broadcasts beacon messages, to which vehicles respond by using their self-generated pseudonyms. The RSU verifies the pseudonym and uses an ID-Based Signature (IBS) scheme to generate an offline signature for the vehicle, which is used for authentication. An allocation set message is then broadcast for vehicle-to-vehicle communication. However, the scheme has some limitations, including the use of Elliptic Curve Cryptography (ECC) signature schemes that take more time for verification than RSA. Additionally, if a vehicle does not have the pseudonym and POI set of another vehicle in its storage, it sends a query to the nearest RSU for authentication, which can cause delays. As a result, the scheme may not be appropriate for safety-critical applications.

In [90], Kamat et al. has suggested a way to decentralize the issuance of pseudonyms in VANETs. Their proposed scheme involves the utilization of a central Trusted Authority (TA) or Certification Authority (CA) to generate the master secret for an Identity-Based Cryptography (IBC) scheme. The TA publishes system parameters and assigns unique Vehicle IDs (VIDs) to each vehicle. Roadside Units (RSUs) function as decentralized TAs and are responsible for issuing short-lived pseudonyms to vehicles. To generate a pseudonym identifier (PIDi), an RSU receives the system parameters, master secret, and a unique symmetric key (SKi). It authenticates the vehicle using an asymmetric public key certificate and encrypts the VID and timestamp with SKi. The resulting identifier is concatenated with the RSU identifier (IDRSU), a timestamp (TS), and a string denoting the pseudonym holder as a vehicle. The private key (PSKi) is extracted from the pseudonym identifier and transmitted to the vehicle along with the pseudonym key pair (PIDi, PSKi). To resolve pseudonyms, the central TA acquires the SKi of the RSU specified in the PIDi and decrypts the unique VID of the pseudonym holder. This scheme reduces the risk of abuse in case pseudonym-identity mappings are leaked. However, the potential for impersonation is heightened as each RSU possesses the master secret for key extraction, and reliance on RSUs for pseudonym issuance can result in increased overhead.

The authors in [91] propose a method to safeguard safety messages in VANETs by combining an OTIBAAGKA (one-time identity-based authenticated asymmetric group key agreement) protocol and a CMIX (cryptographic mix-zone) protocol. The OTIBAAGKA protocol encrypts safety messages using a private group key, while the CMIX protocol ensures unauthorized access to the network is prevented. Any vehicle within the CMIX can act as a private key distributor, and when there is a need to update the group's private key, a vehicle can distribute a small ciphertext, allowing all vehicles to transition to the new private key. This protocol doesn't solely depend on trusted dealers and facilitates efficient key updates. However, it is crucial for all vehicles to employ the same group private key for message encryption and decryption, which poses a security risk if the group private key is compromised.

The paper [92] suggests an upgraded Identity-Based Batch Verification (IBV) approach to address security and privacy concerns in VANETs. This method utilizes batch message verification, incorporating point multiplication and pairing operations to ensure security under the ROM (Random Oracle Model). Initially, the Trusted Authority (TA) stores the authentic identity, password, and private keys in each vehicle's Trusted Personal Device (TPD). The vehicle generates an anonymous identity and utilizes it to sign the message, which can be verified by the recipients within a reasonable timeframe. However, this scheme is complex due to the intricate process of anonymous identity generation, as well as message signing and verification.

In [93], the authors introduce an identity-based signcryption (IBSC) scheme that ensures security and employs bilinear pairing, decisional modified bilinear strong Diffie–Hellman (MBSDH), and modified bilinear Diffie–Hellman inversion (MBDHI) assumptions. In this scheme, the sender generates a private key using the Extract algorithm and creates a ciphertext CT for the message M by utilizing the sender's genuine identity, private key, and the receiver's identity. Upon receiving the ciphertext CT, the receiver verifies it using their private key and returns the message M if the verification is successful. Although this scheme is secure and resistant to forgery, it is intricate and computationally intensive due to its reliance on bilinear pairing.

The authors in [94] propose a hybrid cryptography scheme called Identity and HMAC-based Trust Management Hybrid Cryptography (TMHSC) to manage trust in VANETs. The trust value of each vehicle is calculated by the Agent Trusted Authority (ATA) based on reward points. To facilitate the registration of offline vehicles with the Regional Transport Office (RTO), a comprehensive process involving pre-authentication and trust-value updates is implemented. This process includes registering with an adjacent Online Authorized Trust Authority (ATA). The pre-authentication and trust-value updates are performed by leveraging the Road Side Unit (RSU),

which plays a vital role in ensuring the trustworthiness of the vehicle for Vehicle-to-Vehicle (V2V) communication. The scheme employs V2V authentication, trust evaluation, and the computation of a new trust value specifically for the sender. As a result, the scheme offers robust vehicle authentication, message authentication and integrity, traceability, non-repudiation, and unlinkability. It is important to note, however, that the scheme does not incorporate batch verification of messages and signatures.

The paper [95] introduces a privacy-preserving authentication scheme called SIPAR, which is based on identity verification and designed to support the efficient revocation of vehicles in VANETs. SIPAR enhances security by not storing the system's master key in the Trusted Personal Device (TPD) and eliminates the need for bilinear pairing operations to speed up verification. The message signing process involves the use of a private key and pseudonym. The receiver uses batch verification to check messages and signatures. SIPAR protects against modification attacks, replay attacks, and impersonation attacks while ensuring anonymity, non-repudiation, traceability, and authentication. It does not, however, provide any data on the message or packet loss ratio.

4. 2. 4. Machine Learning solutions

Machine learning is a widely adopted solution for enhancing the security of VANETs. By combining statistics and algorithms, machine learning models can learn from data to produce predictions or make decisions [96]. This allows them to detect patterns and behaviors that deviate from normal, making them useful for identifying legitimate vehicles and misbehaving nodes [97]. Machine learning approaches are commonly used for preventing Denial of Service (DoS) attacks and their variants.

Several machine learning uses Support Vector Machine (SVM) algorithms. Li et al. [98] proposed an Intrusion Detection System (IDS) for VANETs that uses the Support Vector Machine (SVM) algorithm to detect anomalous vehicles. The IDS employs both behavioral and contextual information to train the SVM classifier, which makes it resilient against various attack patterns and environmental changes. To facilitate nodes to reach a consensus on malicious nodes, the IDS is deployed on each node to analyze the behavior of neighboring nodes and exchange information with each other. The system employs the Dempster-Shafer theory to fuse data at each node, providing a broader view of the network. The performance evaluation includes measuring communication overhead, precision, and recall and comparing the results with previous works. The authors collected contextual information such as velocity, channel status, temperature and wind speed, GPS coordinates, and altitude, but they did not use this information in the evaluation. The binary classifier was built using the SVM algorithm, and its performance was evaluated using

precision and recall. A novel method to guarantee the prevention and detection of Jellyfish attacks in MANETs was put out by Doss Srinath et al. [99]. This method combines a support vector machine (SVM) for learning packet forwarding behavior with an authenticated routing-based architecture for identifying Jellyfish attacks. The suggested method chooses trusted nodes in a network for packet routing by using the hierarchical trust evaluation attribute of nodes. The outcomes of the simulation shows how well the method works to identify jellyfish attacks. The study in [100] focuses on investigating Distributed Denial of Service (DDoS) attacks in an SDN-based vehicular network caused by TCP flood, UDP flood, or ICMP flood. To efficiently identify and respond quickly to these attacks, the authors suggest using the SVM algorithm. The flow table entries are used as features for SVM training, and each new entry is forwarded to the controller by the data forwarding plane. The PACKET_IN trigger check is performed on the controller, and the PACKET_IN message rate is compared with the threshold rate. If an abnormality is detected, the attack detection module receives a warning message. The SVM recognition algorithm examines the flow characteristics to determine whether an attack is present. If an attack is confirmed, the attack warning system generates an alert and takes further steps. The framework was evaluated using the Mininet and Floodlight controller, and it demonstrated superior detection accuracy and a lower false positive rate. In [101], a hybrid approach based on the support vector machine (SVM) kernel is proposed to detect DDoS attacks. The approach utilizes jitters, packet drops, collisions, and other features to generate various types of data in a simulation that models a real-time scenario containing both regular communications and DDoS attacks. The performance of the hybrid model is evaluated using the suggested algorithm to determine its ability to differentiate between regular communications and DDoS attacks. However, this approach has some limitations, including an invalidated parametric evaluation, lack of coverage for vehicle-to-vehicle (V2V) communication, and high storage requirements.

Other machine learning solutions use decision tree algorithms. The authors in [102] suggested a secure cloud service for connected vehicles that employs machine learning to detect cyber attacks and ensure QoS and QoE for the user. They developed an intrusion detection mechanism that involves three stages, namely traffic data analysis, compression, and classification mechanisms. These phases aid in distinguishing between trustworthy and malicious service requests. The system's effectiveness was assessed by examining the results of simulations.

In their research paper [103], Grover et al. used the Random Forest algorithm to detect misbehaviors in vehicular ad hoc networks (VANETs). They utilized three inputs, including a proposed VANET model, an attack model, and the affected VANET application, to extract

different features and identify various misbehaviors. By conducting experiments with different attack combinations, observer nodes computed features such as speed deviation, distance, Received Signal Strength (RSS), and the number of generated/delivered/dropped/collided packets. However, the authors mentioned that the proposed approach may not be suitable for detecting temporal attacks like replay attacks in real VANET scenarios.

In their research, Kosmanos et al. [104] introduced a supervised machine learning method utilizing K-clustering to effectively identify position falsification attacks in vehicular networks. To train their model, they employed a dataset generated through the Veins network simulator, incorporating four key features: Signal Strength Indicator, Signal Quantity Indicator, PDR, and PVRS. The ML model was designed as a binary classifier, and two classification algorithms, KNN and RF, were evaluated. The authors utilized FPR (False Positive Rate), TPR (True Positive Rate), and ROC (Receiver Operating Characteristic) as evaluation metrics to assess the model's performance.

In [105], Ghaleb et al. presented a misbehavior detection model that uses Artificial Neural Network (ANN) techniques. They described how vehicles use sensors to collect information from their surroundings and share it through V2V communication, which is used to construct a Local Dynamic Map (LDM) of a vehicle. The LDM is then used to derive features for detecting misbehavior, and the historical values of these features are used to build a model. This model is then applied in real-time to classify new messages as legitimate or malicious. The features used in their work include plausibility and consistency checks from previous research, as well as some new features. The authors tested their model by injecting dynamic noise to simulate a data injection attack, where 20% of vehicles were malicious. Seven features were used to train the ML model: overlaying check, consistency of reported uncertainties, mobility message prediction error, communication-based feature, appearance position-based feature, average mobility messages prediction error, and the time since the last mobility message was received.

5. Gaps analysis and open issues

The security of Vehicular Ad-Hoc Networks (VANETs) is critical as it directly impacts the safety of people's lives. As the number of vehicles and applications in VANETs increases, along with the sophistication of attack processes, it becomes increasingly challenging to maintain the security of these networks. Therefore, it is essential to develop futuristic solutions to ensure the sustainability of security in VANETs. In this section, a GAP analysis between different security solutions is discussed, followed by a presentation of some open issues that require further research. By

analyzing the GAP between existing solutions and identifying open issues, researchers can work towards developing more effective and robust security mechanisms for VANETs, which can help ensure their secure and sustainable operation in the future.

5. 1. Existing solutions gaps analysis

Performing a gap analysis in the context of VANET involves identifying missing or necessary requirements in relation to desired outcomes. This involves comparing existing research to the desired goals and identifying any gaps between them. Once the gaps have been identified, potential solutions can be proposed to address them. Table 2.4 provides a comparison of selected solutions based on various criteria such as used approach, assured service, Covered attack, and research gaps. This comparison can help identify a compromised solution among different services.

Table 2.4. Comparison of security solutions in VANETs

Ref, year	Basic Mechanism	Covered Attack	Service	Gaps
[62], 2002	Reputation and trust	Man-in-the-middle	Integrity	Overhead increment in high mobility
[81], 2002	PKI	Eavesdropping, Replay, Impersonation	Authentication, Non-repudiation	Route discovery delays
[83], 2002	PKI	DoS, Replay attack	Availability, Non-repudiation	Greater effectiveness is needed in terms of PDR and reduction of computational overheads.
[82], 2003	PKI	DoS, Impersonation	Authentication, Availability	Improved performance is necessary to reduce latency and overhead
[63], 2005	Reputation and trust	DoS	Availability	Insufficient details about the model, continuously added overhead.
[90], 2006	ID-cryptography	Eavesdropping, Man-in-the-middle, Replay	Confidentiality, Integrity, Authentication, Non-repudiation	Validation is needed
[77], 2007	Reputation and trust	Eavesdropping	Authentication, Privacy	A bottleneck in terms of scalability, Increased complexity, centralized authorities
[67], 2008	Reputation and trust	DoS, Bogus information	Integrity, Availability	Not suitable for sparse environments
[79], 2008	PKI	Eavesdropping, replays	Privacy	Not tested in VANET dataset
[69], 2009	Reputation and trust	Bogus information	Authenticity	High vehicle speeds reduce the system's accuracy
[66], 2010	Reputation and trust	Bogus information	Authenticity	Suitable only for EEBL application
[68], 2010	Reputation and trust	Message modification, DoS	Integrity, Availability	Communication issues
[78], 2010	Reputation and trust	Bogus information	Authenticity, Privacy	A bottleneck in terms of scalability, Increased complexity, centralized authorities

[88], 2010	ID-cryptography	Bogus information, Replay attack, Man-in-the-middle	Confidentiality, Integrity, Authentication, Non-repudiation	Validation is needed
[85], 2011	PKI	Sybil attacks	Authenticity, ID traceability	Detecting is difficult in regions governed by different CAs
[103], 2011	Machine Learning	DoS, DDoS	Availability	Cannot be used for time-based attacks in a practical situation
[89], 2012	ID-cryptography	Eavesdropping, Man-in-the-middle	Authentication, Privacy	Verification delays, not appropriate for safety-critical applications
[84], 2013	PKI	Bogus information	Privacy	Validation is needed
[64], 2015	Reputation and trust	DoS	Availability	No security requirements to choose the forwarder node
[98], 2015	Machine Learning	DoS, DDoS	Availability	Validation is needed
[91], 2017	ID-cryptography	Eavesdropping, message modification, identity impersonation	Integrity, Authentication, Non-repudiation	Security breach if the group private key is compromised
[92], 2017	ID-cryptography	Identity impersonation, repudiation	Privacy	Convoluted identity generation, complex message signing and verification
[105], 2017	Machine Learning	DoS, DDoS	Availability	Validation is needed in DoS attack presence
[100], 2018	Machine Learning	DoS, DDoS	Availability	Appropriate only for SDN
[71], 2019	Reputation and trust	DoS, DDoS	Availability	Improved performance in terms of PDR and latency is necessary
[102], 2019	Machine Learning	DoS, DDoS	Availability	Not tested in the VANET dataset
[72], 2020	Reputation and trust	DoS, DDoS	Availability	Cheater attacks can be executed
[94], 2020	ID-cryptography	Eavesdropping, message modification, identity impersonation	Authentication, integrity, traceability, non-repudiation	Does not include batch verification of messages and signatures
[95], 2020	ID-cryptography	message modification, replay attacks, impersonation	Non-repudiation, traceability, authentication	No packet loss ratio information
[101], 2020	Machine Learning	DoS, DDoS	Availability	Evaluation not confirmed

After presenting and analyzing various solutions in VANET security, several emerging and open issues are raised that require further research and attention. These issues will be further elaborated in the following section.

5. 2. Open issues and emerging

Research on VANETs has been ongoing for many years, but there is still a need for more work to be done. Although previous research has tackled specific attacks, there are still vulnerabilities that need to be addressed. Furthermore, research is required in areas such as security resource consumption for different DOS attacks, secure routing protocols, robust key management, trust-

based systems, integrated approaches to routing security, data security at different levels, and cooperation enforcement.

Current routing protocols are susceptible to various attacks that can allow attackers to manipulate a victim's selection of routes or cause denial-of-service attacks. Cryptography is commonly used for security, but its strength is based on secure key management. The use of a centralized Certificate Authority (CA) in public cryptography schemes is a security vulnerability in VANETs because it creates a single point of failure. While symmetric cryptography is efficient, it is still susceptible to potential attacks on key distribution. Therefore, developing efficient key agreement and distribution methods is an ongoing research area in VANETs.

In addition to the above, future research could include building a trust-based system and integrating it into the current defensive approaches to address the node selfishness problem. To identify new security threats and develop new countermeasures, more research is necessary in VANETs.

Data verification about specific events is a significant issue that can be addressed by developing correlation mechanisms in vehicular nodes that receive related data messages. These techniques need to be evaluated in extremely dynamic VANETs where the quantity of vehicles and their speeds are constantly changing. In order to simulate VANETs in various contexts and take into account a variety of characteristics, including data traffic loads, the channels of communication employed, probabilities of reception, and latency, mathematical models can be used.

6. Conclusion

The safety of human lives is of utmost importance in VANETs, which makes them a popular target for various types of attacks that can range from insignificant to catastrophic. As a result, securing VANETs is a significant challenge that has been addressed through advanced research in the security field, and researchers have proposed a wide range of solutions to counter different types of attacks. In this chapter, we have provided an overview of the security requirements of VANETs and discussed the features, security challenges, and constraints associated with them. We have also presented a comprehensive discussion of the common attacks and threats that can be launched against VANETs and their classifications. Finally, we have reviewed various security solutions proposed in VANETs and compared them based on well-known security criteria in the field. This literature review can provide a comprehensive understanding of the current state of VANET security and help in developing more effective solutions to secure VANETs.

Chapter 3

Secure and Efficient routing protocol for Vehicle-to-Everything

Content

1. Introduction

In recent years, Vehicular Ad Hoc Networks (VANETs) have emerged as a promising technology for enhancing road safety, improving traffic management, and providing various other applications such as infotainment and smart navigation systems. However, security and privacy concerns have become major obstacles in the widespread deployment of VANETs. As VANETs are highly dynamic and distributed networks, they are vulnerable to various attacks such as Sybil attacks, message forgery, and denial-of-service attacks. Therefore, there is a need for effective security mechanisms to ensure the confidentiality, integrity, and authenticity of messages exchanged between vehicles in VANETs.

To address these challenges, this chapter proposes a combined approach of two security mechanisms: Reputation and Blockchain. The Reputation system is used to evaluate the behavior of the vehicles in VANETs based on their past interactions and to assign trust values to them. A high level of security is provided by integrating the Blockchain with the reputation system. Blockchain technology offers qualities that are widely desired for protecting communication in VANETs, including immutability, decentralization, distributed ledgers, consistency, security, integrity, and transparency.

This chapter also introduces a method for maintaining message integrity while also supplying the user with anonymous authentication using minimal computational resources. In order to do this, the elliptic curve's points are bilinearly paired, and blockchain integration is used. The elliptic curve, which offers great security and is challenging to break, is used in the proposed approach. The integrity of the message is verified using the hash function, which is also used to authenticate the authorized user using digital signatures and hashing. The proposed approach assures that a message gets considered if both hash values match; else, the message is discarded.

2. Preliminaries

In this section, we provide an overview of the key concepts and components that form the foundation of our proposed protocol. Specifically, we discuss Blockchain technology, the role of the Trusted Authority (TA), and the network entities involved in the protocol.

2. 1. Blockchain Integration in SecE-V2X

In the context of the SecE-V2X protocol, blockchain technology is seamlessly integrated to enhance the authenticity and efficiency of the VANET system. This novel approach leverages blockchain's inherent characteristics to facilitate secure and rapid authentication without relying on

a central Trusted Authority (TA). Below is a description of how blockchain is applied within the SecE-V2X approach:

2. 1. 1. Blockchain Structure

The fundamental building block of blockchain technology is the "block" (as represented in Figure 3.1). In our context, these blocks constitute a distributed ledger that captures and secures crucial information. Notably, the transactions recorded within each block are immutable and resistant to tampering, ensuring the integrity of data within the VANET system.

2. 1. 2. Interlinked Blocks

Each block in the blockchain is interlinked with the previous one through a cryptographic hash. This linkage forms a chain of blocks, creating a robust and unbreakable connection. Any attempt to modify the content of a single block reverberates throughout the entire blockchain, rendering it resistant to unauthorized alterations [106].

2. 1. 3. Transparency and Data Representation

The information loaded into the blocks of our VANET blockchain is marked by its transparency. Transactions and data are vividly represented through the utilization of SHA256 hash codes, making the system's operations clear and accountable [107].

2. 1. 4. Decentralization

One of the defining features of blockchain is its decentralization. There is no third-party authority governing the blockchain, ensuring that the system operates autonomously and free from centralized control. This decentralization fosters trust among network entities [107].

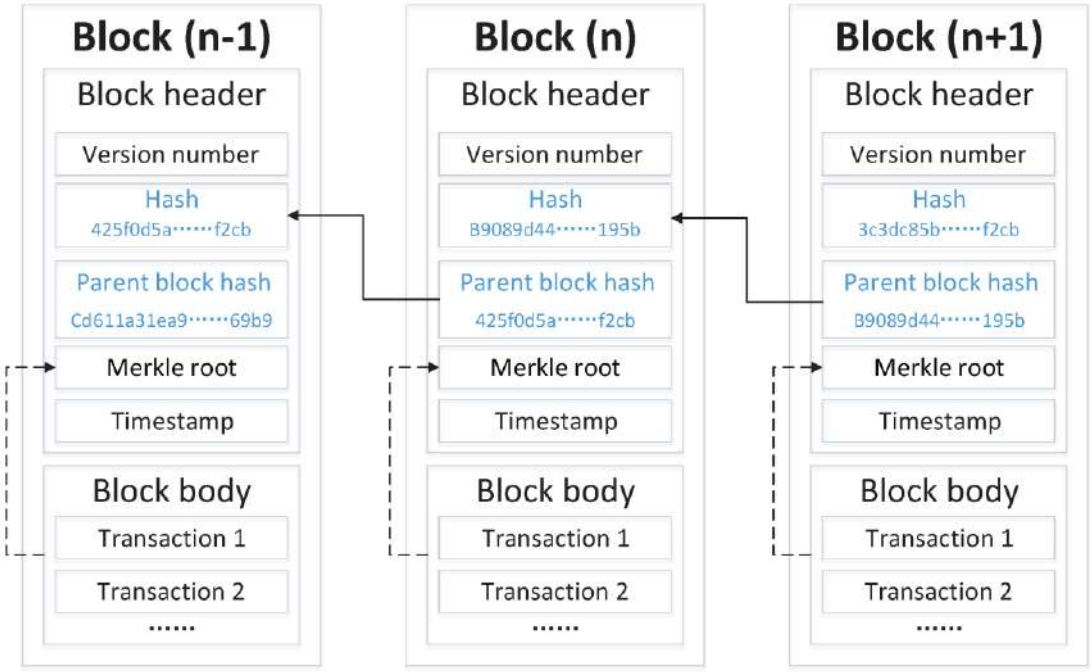

Figure 3.1. Blockchain architecture

2. 1. 5. Blockchain in VANETs

In the context of VANETs, the integration of blockchain technology revolutionizes the authentication process when vehicles transition between Road Side Unit (RSU) regions. Traditionally, authenticating vehicles at each new RSU would impose significant computational burdens, influencing VANET performance.

However, by incorporating blockchain into our proposed approach, the need for extensive TA involvement is mitigated. Initially, the TA computes vital public and private parameters, storing them in the blockchain. Among these parameters are DIDv (Dummy Identity), and A, where A = e(P, Q) γi. This dummy identity, mapped to the original identity, facilitates seamless communication without disclosing sensitive information.

When a vehicle moves between RSU regions, the RSU retrieves the dummy identity from the blockchain to generate an authentication receipt. This receipt is then shared with neighboring RSUs, eliminating the need for frequent re-authentication. The RSU validates the authenticity of AID1 (Authenticated ID1) by computing e(AID1P, VID1) and cross-referencing it with the blockchain, where it should match A.

By leveraging the blockchain, the SecE-V2X protocol verifies authenticity without the involvement of a centralized TA, significantly reducing re-authentication time and enhancing overall system efficiency.

2. 2. Elliptic Curve Cryptography (ECC) in SecE-V2X

In the context of our proposed SecE-V2X protocol, the utilization of Elliptic Curve Cryptography (ECC) plays a pivotal role in ensuring the security and integrity of vehicular ad-hoc networks. This section delves into ECC, shedding light on its security attributes and functionalities, and highlights its significance in our research.

2. 2. 1. The Foundations of ECC:

Elliptic Curve Cryptography, commonly referred to as ECC, emerged in 1985 through the pioneering work of Neal Koblitz and Victor S. Miller [108]. At its core, ECC is an asymmetric key cryptosystem grounded in the concept of elliptic curves. An elliptic curve comprises a set of points satisfying a specific mathematical equation, represented generically as [106]:

$$y^2 = x^3 + ax + b$$

Here, 'a' and 'b' represent constants that define the specific curve.

2. 2. 2. Distinctive Features of ECC:

ECC boasts several distinctive features that set it apart from other public cryptosystems:

- **Efficiency and Compactness**: One of ECC's most remarkable features is its ability to provide the same level of security as the RSA algorithm but with significantly smaller key sizes. For instance, a 160-bit ECC key offers equivalent security to a 1024-bit RSA key.

- **Resource-Friendly**: ECC is renowned for its suitability in resource-constrained environments, such as mobile devices, RFID systems, and cryptocurrencies. Its compact key size and computational efficiency make it an ideal choice for such applications.

- **Optimal Security-Performance Tradeoff**: ECC strikes a balance between high security and the use of short, efficient keys, making it a preferred cryptographic solution in scenarios where both security and performance are critical.

- **Versatility**: Thanks to its lightweight nature and effectiveness, ECC finds extensive use in various applications, including secure web browsing via SSL/TLS and the cryptocurrency realm, exemplified by its use in Bitcoin.

2. 2. 3. Elliptic Curve Cryptography (ECC) Algorithm:

Our research harnesses the potential of the Elliptic Curve Cryptography (ECC) algorithm, an asymmetric cryptographic scheme that underpins the security of SecE-V2X. Below, we provide an overview of ECC's core characteristics and its role within our protocol.

- **The ECC Equation:** ECC employs an asymmetric elliptic curve, defined by the mathematical expression:

$$y^2 = x^3 + ax + b \; mod \; q$$

Where q is the large prime number, and it is crucial that these integers adhere to ECC properties to avoid singular points. ECC incorporates a trapdoor function, characterized by its one-way computation process, which is computationally simple in one direction but formidable in the reverse direction.

- **Key Size and Efficiency:** ECC stands out for its efficiency, especially in terms of key size. ECC's agility in providing short, swift keys enhances its appeal in secure communication.

By integrating ECC into SecE-V2X, our protocol leverages the security, efficiency, and lightweight nature of ECC to fortify VANET communications, ensuring the confidentiality and authenticity of data exchanged among vehicles.

2. 3. Trusted Authority (TA)

In Vehicular Ad-Hoc Networks (VANETs) security, a Trusted Authority (TA) is an important entity that plays a significant role in ensuring the trustworthiness and security of the network. The TA is responsible for various security-related functions and acts as a trusted intermediary in VANETs.

One of the primary roles of the Trusted Authority in VANET security is the management of digital certificates. The TA is responsible for issuing and maintaining the certificates used by vehicles and infrastructure units in the network. These certificates are used to authenticate the identities of entities participating in VANET communication and ensure the integrity and confidentiality of the exchanged information.

The TA verifies the identity of vehicles and infrastructure units by conducting rigorous authentication procedures. Once the identities are validated, the TA issues digital certificates that

bind the entities' identities to their public cryptographic keys. These certificates serve as trusted proof of identity and enable secure communication and trust establishment within the VANET.

In addition to certificate management, the Trusted Authority may also play a role in key management. This includes the generation, distribution, and revocation of cryptographic keys used for secure communication and encryption in VANETs. The TA ensures that the keys are securely distributed to authorized entities and revoked when compromised or no longer needed.

Furthermore, the Trusted Authority may establish policies and enforce security measures within the VANET ecosystem. It defines rules and guidelines for secure communication, access control, and privacy protection. The TA also monitors the network for any suspicious activities or security breaches and takes appropriate actions to mitigate risks and maintain the security of the VANET.

The Trusted Authority enhances the security and trustworthiness of VANETs by serving as a central trusted entity. It provides a foundation for secure communication, identity verification, and protection against various attacks and threats that VANETs may face. The presence of a Trusted Authority helps to establish a secure and reliable environment for vehicular communication, ensuring the safety and integrity of the network and its participants.

2. 4. Network entities (RSUs and Vehicles)

In our proposed system, two key network members are Roadside Units (RSUs) and Vehicles. These entities play crucial roles in enabling communication and providing services within the VANET ecosystem. Roadside Units (RSUs) are devices strategically installed along roadsides or at specific locations such as parking lots or intersections within a Vehicular Ad-Hoc Network (VANET). Their primary function is to establish local connectivity with vehicles within their coverage area. RSUs are interconnected with each other and with the Trusted Authority (TA). The TA oversees the performance of RSUs and manages any necessary adjustments or concessions. Communication devices embedded in RSUs utilize IEEE 802.11p technology, enabling them to facilitate short-range dedicated communication. RSUs establish both wired connections with neighboring RSUs and the TA, as well as wireless connections with vehicles. They also provide location-based information to authenticated vehicles, and the TA is responsible for issuing the required credentials to RSUs. On-Board Units (OBUs) are essential components installed in every intelligent vehicle within a VANET. These OBUs consist of a set of hardware components that are assembled and programmed to enable intelligent communication. An OBU is securely installed on each vehicle and serves as the communication interface for interacting with other vehicles in the network.

Additionally, OBUs are equipped with Global Positioning System (GPS) technology, which provides real-time data such as latitude, longitude, and time-based information for each vehicle. Furthermore, OBUs may include data recorders that capture and store vehicle crash information, similar to the concept of a black box in aircraft.

3. Security Presumptions

In designing our SecE-V2X protocol, the following presumptions are established in order to guarantee the security and dependability of data:

- Only authenticated and honest nodes are allowed to take part in the routing path.
- Nodes that are the source and destination are regarded as legal.
- The absence of Honesty equivalence between nodes is referred to as a lack of symmetry. Node U being honest with node V does not imply that node V is similarly honest with node U.
- The trustworthy connection between two nodes might be offered to other nodes as a source of recommendations
- Composite: An integrated honest value can be created by combining the honest values gathered through many different paths

4. Proposed protocol

Secure and efficient routing protocol for vehicle-to-everything (SecE-V2X) is a secured routing protocol based on the greedy perimeter stateless protocol (GPSR) by integrating Blockchain technology with the reputation mechanism. The Blockchain technology for maintaining message integrity while also supplying the user with anonymous authentication using minimal computational resources and the reputation mechanism for choosing the trusted next forwarding node from the current node's neighbors to defend the VANET against different attacks categories. The accompanying Table 3.1 provides a list of the symbols' and variables' definitions.

Public and private keys are utilized in the approach we propose to improve security. The corresponding private key is used to create the public key. Due to its reduced key size, Elliptic Curve Cryptography (ECC) is employed in this work to improve system performance.

Table 3.1. List of notations.

Notation	Definition
TA	Trusted Authority
PubK	Public Key
TAver	Verification Key of TA
H	Hash Function
v	Vehicle
ID_v	Identity of vehicle *v*
$CE_{v\text{-}TA}$	Ciphertext Encrypted by *v* with *TA* public key
PSD	The Pseudonym
PrvK	Private Key
EX	The Expiration of *PSD*
$CE_{TA\text{-}v}$	Ciphertext Encrypted by *TA* with *v* public key
Sgn	Signature
SK	Session Key
(S)	Source Node
(D)	Destination Node
t_r	Reception Time
t_s	Sending Time
LR	Packet Loss Rate
PSC	Successfully Sent Packets
PRC	Correctly Received Packets
PSD	Correctly Sent Packets that were Dropped
PRD	Correctly Received Packets that were Dropped
PR	Packet Reordering
D[j]	The Displacement of the Packet *j*
RI[j]	Received Index of the Packet *j*
AS[j]	Arrival Sequence of the Packet *j*
RD	Reordering Density
s_p	Successfully Forwarded Packets
f_p	Unsuccessfully Sent Packets
C_{com}	Communication Confidence
C_{frd}	Forward Packet Confidence
C_{fct}	Uncertainty Confidence Factors
C_{direct}	Direct confidence
w_{com}	Weight Adjustment for Communication Confidence
w_{Delay}	Weight Adjustment for Delay
C_{rcd}	Recommended Confidence
H_{ovr}	Overall Honesty
a and β	Weights

4. 1. Initialization of the system

In the system initiation phase, the Trusted Authority (TA) undertakes a series of steps to establish the foundational parameters and keys essential for secure communications within the VANET

network. To maintain originality and avoid plagiarism, we provide a rephrased version of this process:

- **Elliptic Curve Selection:** The first step involves the TA's selection of a finite elliptic curve defined as: $y^2 = x^3 + ax + b \ mod \ q$, where 'q' denotes a large prime number. This elliptic curve, represented by P and Q, serves as the foundation for cryptographic operations.
- **Key Generation:** Following curve selection, the TA proceeds to generate random values α and β from the multiplicative group Z_q^*, where Z_q^* represents the multiplicative group of size 'q'. These random values are utilized to derive the TA's public key (*PubK)* and verification key (*TAver*). Specifically, *PubK* is computed as *PubK* = αP, while *TAver* is calculated as Tver = βP.
- **Hash Function Specification:** As part of the initialization process, the TA designates a hash function denoted as '*H*'. This hash function plays a critical role in ensuring data integrity and security within the VANET network.
- **Publication of Parameters:** To facilitate secure communication among all users and Roadside Units (RSUs) within the VANET network, the TA publishes the following parameters: (*PubK, TAver, H,* $P, Q, e(P, Q), q$).

By disseminating these essential parameters to all network participants, the system initializes the foundation for secure and authenticated communication within the VANET network.

4. 2. Network entities registration

Using the Identity-Based Signature presented in [109], the network entities are required to provide their real identities to the *TA* for the purpose of registering. Code Segment 3.1 below describes the registration process in detail.

For vehicles, vehicle *v* uses *PubK* and its ID_v to compute the ciphertext encrypted $CE_{v\text{-}TA}$, where $CE_{v\text{-}TA}$ is the ciphertext encrypted by vehicle *v* with *TA* public key *PubK*. Finally, $CE_{v\text{-}TA}$ is sent to *TA*. When *TA* receives the registration message from vehicle *v*, *TA* decrypts $CE_{v\text{-}TA}$ and extracts ID_v. Then, *TA* calculates the authentication information and generates *n* pseudonym PSD_i and calculates the associated public key $PubK_i$, private key $PrvK_i$, where $i \in \{1,2,\ldots,n\}$, EX_i is the expiration of PSD_i. Finally, *TA* computes the ciphertext encrypted $CE_{TA\text{-}v}$ and sends it to the vehicle.

For RSUs, TA determines and provides the private key $PrvK_{rsu}$ to the RSU. As soon as RSU receives $PrvK_{rsu}$, it is able to create a signature that is deemed valid and take part in the authentication.

Code Segment 3.1. Steps involved when registering

Vehicle side

```
Enc_ID←encrypt(ID_v);  // Encrypts the real ID
sendToTA(Enc_ID); // Send the encrypted real ID
WaitForResponse;
receiveFromTA(Enc_msg); // Receive the encrypted message
decrypt(Enc_msg); // Decrypt the received message
extract(PSD_v[i], EXP_v[i], PrvK_v[i]); // Extract the parameters
validity←check(PSD_v[i], EXP_v[i], PrvK_v[i]); // Check the parameters validity
if validity=true then
store(PSD_v[i], EXP_v[i], PrvK_v[i]); // Store the parameters
end if
```

Trusted Authority side

```
receive(Enc_ID); // Receive the encrypted real ID
ID_v←decrypt(Enc_ID);  // Decrypt the vehicle real ID
for i←1 to n do
        generate(PSD_v[i]); // Generate pseudonym
        generate(EXP_v[i]); // Generate the expiration of pseudonym
        PubK_v[i] ←H(PSD_v[i], EXP_v[i]); // Calculate corresponding public key
        PrvK_v[i] ← PrvK_TA(H(PSD_v[i], EXP_v[i])); // Calculate corresponding private key
end for
Enc_msg← encrypt(PSD_v[i], EXP_v[i], PrvK_v[i]); // Encrypt the parameters
sendToVehicle(Enc_msg); // Send the encrypted message to the vehicle
storeToBlockchain(ID_v, PSD_v[i], EXP_v[i]); // Store the parameters in the Blockchain
```

4. 3. Vehicle to RSU Authentication

This process entails confirming the legitimacy of vehicles' identities when they connect to Roadside Units (RSUs). It guarantees that only authorized vehicles can safely communicate with RSUs and utilize the network services that they offer. The vehicle-to-RSU authentication process typically involves the following steps:

- Immediately as a vehicle reaches the RSU's signal coverage area, the vehicle generates a signature Sgn_v and sends it to RSU.
- The RSU checks the validity of the received Sgn_v, if so, the vehicle is regarded as legitimate. For legitimate vehicle, the RSU generate signature Sgn_{rsu} and session key SK_{rsu-v} and send them to the vehicle.
- Vehicle checks confirm the validity of Sgn_{rsu}. Then, the vehicle calculates session SK_{v-rsu} and a secure channel is established between the vehicle and the RSU.

4. 4. Network entities communications

Nodes gather information that is used in the decision-making process when communicating with their neighbors by keeping track of each neighbor node's packet loss rate, latency, and packet reordering. Following that, they calculate their Honesty and keep it locally in their table of neighbors. A node neighbors table is shown in Table 3.2.

Table 3.2. Node neighbors' table in SecE-V2X.

Node-Id	Position Information (x,y)	Overall-Honesty	Timestamp
1	6218.79, 2363.39	0.75	22.75
2	6552.16, 2286.45	0.63	23.87
3	6536.72, 2181.74	0.86	23.42

Along with the GPSR beacon that is periodically broadcast, the proposed protocol also sends an honesty beacon message that includes the updated data after each neighbor node's honesty update. As part of our proposed strategy, Figure 3.2 depicts the Honesty update beacon structure.

Node-Id	Position-Information	Neighbor-Id	Neighbor-Honesty	Timestamp

Figure 3.2. The Honesty beacon format in SecE-V2X.

When a node wants to forward data to an intended destination that is not in its coverage area, it initially encrypts the message and the message's creation timestamp with a signature, selects the best-honesty forwarding node using data stored in its neighbors' tables, and then delivers the data to that node. The selected node then sends the data it has just received in the same manner to the following node, and so on until the destination node has been reached. The malicious nodes will be prevented from taking part in the routing process in this way. On the basis of the proposed next-hop selection strategy, the chosen relay node is depicted in Figure 3.3. The source *(S)* chooses node 3 even though it is not the closest to the destination *(D)* in this case. As can be observed in Table 3.2, node 3 has a greater value of Honesty when compared to other neighbors of *(S)*.

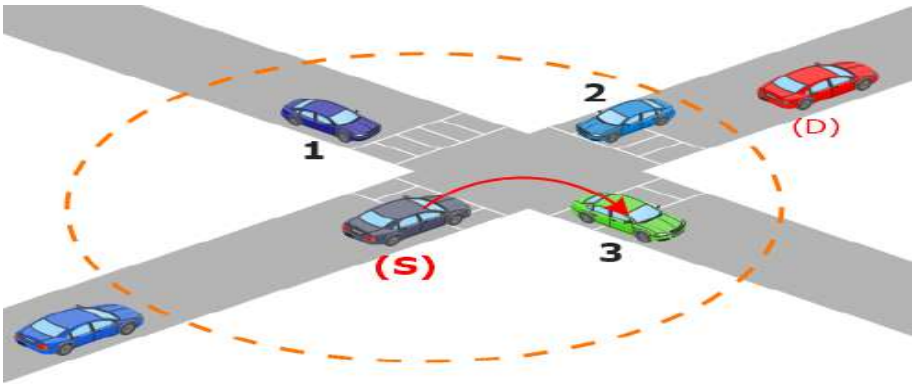

Figure 3.3. Choosing the next hop in the SecE-V2X protocol.

The overall Honesty value of a node is calculated using the communication behavior of packet delivery between nodes, which is expressed in the packet loss rate, latency, packet reordering, and confidence of the node. On a scale from *0* to *1*, where *0* represents the totality of dishonesty and *1*, the totality of honesty, the honesty value is established. Due to the dynamic topology and self-organizing nature of the network, nodes will randomly join and depart the network. This necessitates real-time updating of node honesty. The transactions between nodes determine how this value is changing at any given moment. Figure 3.4. displays an overview of the proposed protocol's entire sequence of work. The remaining part of this section describes the Honesty factors and the next-hop selection strategy used in the proposed approach.

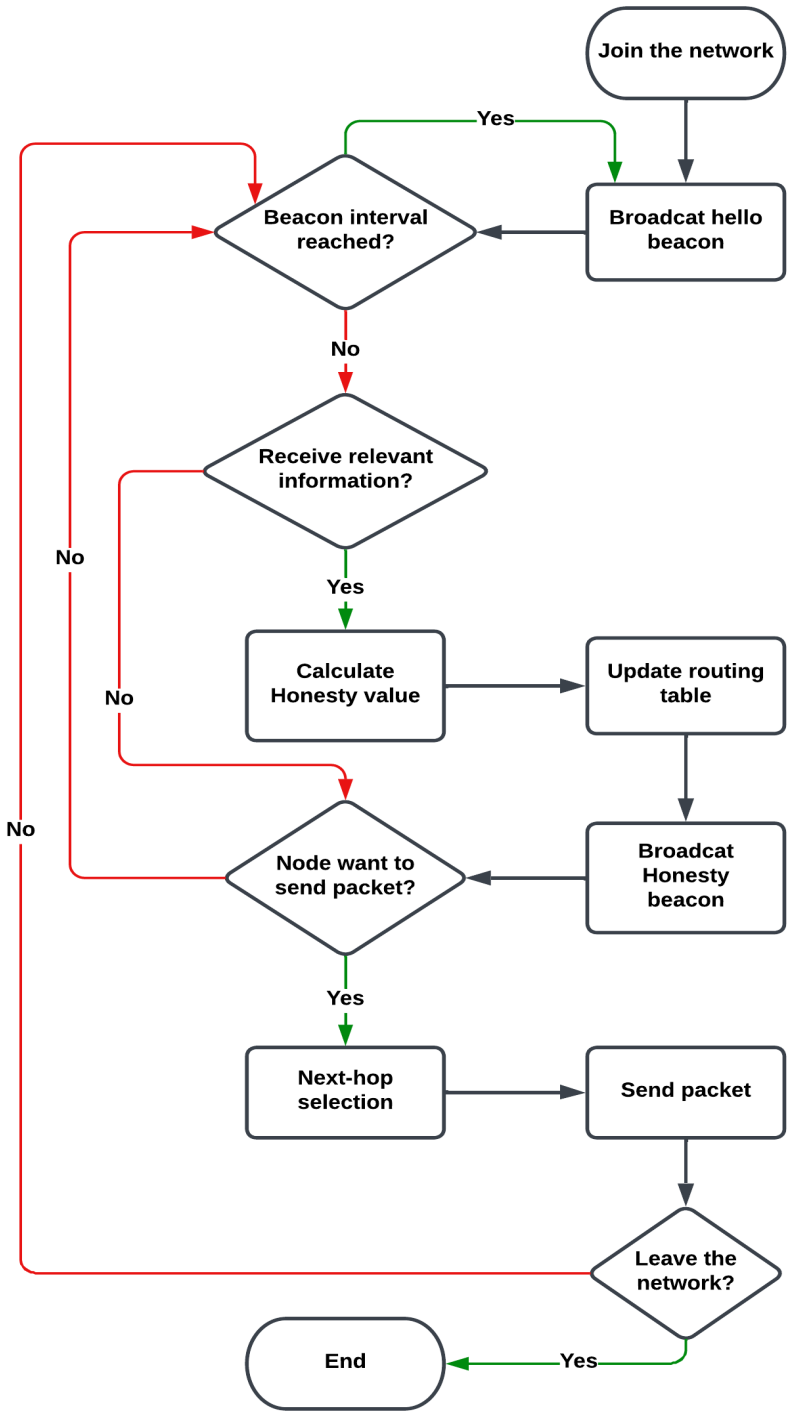

Figure 3.4. The flow chart of the entire working of the proposed protocol.

4. 5. Honesty Metrics

Three Honesty metrics are employed in the proposed protocol: packet latency, packet loss rate, and packet reordering. By calculating these metrics, our suggested technique prevents the attacker nodes from taking part in the routing process. These metrics are fully explained in the subsections that follow.

- **Packet Delay**

In a network with an unstable environment, a node's delay rate is largely stable after sending or receiving data, or it fluctuates within a specified tolerance; however, the rate of delay is substantially higher for attacker nodes like DoS than it is for legitimate nodes. Delay is known as the average amount of time required to send data to nearby nodes. Equation (3.1), presented in [110], can be used to calculate it:

$$D = t_r - t_s \tag{3.1}$$

Where the times a packet is received and sent, respectively, are denoted by t_r and t_s.

- **Packet Loss Rate**

A given time period's total number of transmitted or received packets divided by the number of lost packets is known as the packet loss rate. Equation (3.2), presented in [111], is used to calculate the packet loss rate *LR(n)* of a node n:

$$LR(n) = \frac{PRD + PSD}{PRC + PSC + PRD} \tag{3.2}$$

where *PSC* is the quantity of successfully sent packets, *PRC* is the quantity of correctly received packets, *PSD* is the quantity of correctly sent packets that were dropped, and *PRD* is the quantity of correctly received packets that were dropped.

- **Packet Reordering**

The proportion of packets that arrive late compared to their predicted position is the definition of this metric, as indicated by the received index, to capture the packets' displacements from their original locations. Equation (3.3), presented in [112], is used to determine the packet reordering, *PR*:

$$\begin{cases} PR = \sum_{i=+1}^{i=D_r} RD[i] \\ D[j] = RI[j] - AS[j] \end{cases} \tag{3.3}$$

where *D[j]* denotes the displacement of the packet, j, *RI[j]* denotes its received index, and *AS[j]* is its arrival sequence. Referring to the distribution of packet displacements from their sending positions, reordering density *(RD)*, as proposed in [113], is used. As a result, $PR = 0$ represents the scenario in which every packet is in the proper order, whereas $PR > 0$ represents the scenario in which the packet sequence has been changed.

4. 6. Calculating the Node Honesty

A node computes the direct and recommended confidences to determine the honesty of a neighbor node. We demonstrate how to determine the node Honesty in the proposed approach in this subsection.

- **Communication Confidence**

The quantity of transferred packets can be used to calculate communication confidence. Equation (3.4) from [114] is used to calculate communication confidence:

$$\begin{cases} C_{com} = \dfrac{2C_{frd} + C_{fct}}{2} \\ C_{frd} = \dfrac{2s_p + 1}{2\gamma} \\ C_{fct} = \dfrac{1}{\gamma} \\ \gamma = s_p + f_p + 1 \end{cases} \tag{3.4}$$

Where s_p represents the number of packets that were successfully forwarded, f_p represents the number of packets that were unsuccessfully sent, C_{com} represents communication confidence, C_{frd} represents forward packet confidence, and C_{fct} represents factors of uncertainty confidence.

- **Direct Confidence**

By combining the communication confidence, delay, and packet reordering, PR, Equation (3.5) is used to compute the direct confidence:

$$C_{direct} = \begin{cases} w_{com} \times C_{com} + \dfrac{w_{Delay}}{D} & PR = 0 \\ 0 & PR > 0 \end{cases} \tag{3.5}$$

The terms C_{com}, D, and PR, respectively, stand for communication confidence from Equation (3.4), delay from Equation (3.1), and packet reordering determined from Equation (3.3). The weight adjustment factors for communication confidence and delay, respectively, are w_{com} and w_{Delay}, with $w_{com} + w_{Delay} = 1$ and $w_{com}, w_{Delay} \in [0, 1]$.

- **Recommended Confidence**

Both direct and recommended confidence are taken into account when calculating confidence. Equation (3.6) is used to determine the recommended confidence depending on the information provided by the suggestions of the nearby nodes:

$$C_{rcd} = \frac{\sum_{i=1}^{n} H_{ovr}(i)}{n} \tag{3.6}$$

where H_{ovr} *(i)* is the honesty suggested by neighbor i and n is the total number of neighbors (Equation (3.7)).

- **Overall Honesty**

Based on the specific effects of each type of recommended and direct confidence, integrated Honesty (H_{ovr}) is determined. We used a and β weights to understate the impact of erroneous confidence on the nodes that provide misleading recommended confidence. Equation (3.7) is used to determine the overall Honesty, where $a, \beta \in [0,1]$, $a + \beta = 1$, *and* $a > \beta$:

$$H_{ovr} = \alpha \times C_{direct} + \beta \times C_{rcd} \quad (3.7)$$

4. 7. SecE-V2X Routing Algorithm

During the next-hop selection phase, the SecE-V2X algorithm isolates malicious nodes to prevent them from taking part in packet routing and ensuring security in VANET networks. To make sure of this, each time a node communicates with a neighbor, the neighbor's Honesty value is calculated in Equation (3.7) using the packet loss rate, delay, and packet reordering. Following every Honesty update, the beacon broadcasts the updated Honesty. The steps required to determine the neighbor's Honesty value are described in Code Segment 3.2 below.

Code Segment 3.2. Steps involved when calculating the Honesty value for the neighbor (N).

```
// The steps taken to calculate the neighbor Honesty value
With the neighbor N do
D←getDelay(N);          // Getting the delay of the node N
Sp←getSp(N);  // Getting the number of successfully forwarded packets of the node N
Fp←getFp(N);  // Getting the number of unsuccessfully forwarded packets of the node N
y←Sp+Fp+1;
Cfrd←(2*Sp +1)/(2*y);   // Calculating the forward packet confidence
Cfct←1/y;          // Calculating the factor of uncertainty confidence
Ccom← (2*Cfrd+Cfct)/2          // Calculating the communication confidence
PR←getPR(N);            // Getting the packet reordering value of the node N
if PR=0 then
Cdirect← Wcom*Ccom+Wdelay/D        // Calculating the direct confidence
else
Cdirect←0;
```

```
end if
Crcd←0;
for all i in neighborsTable do
        Crcd←Crcd+getHovr(N)[i];        // Calculating the sum of Recommended confidences
end for
Crcd←Crcd/neighborsTable.length;      // Calculating the average of the Recommended confidences
Hovr←a*Cdirect+b*Crcd;          // Calculating the overall honesty
My-id ← getNodeId();  //Getting current node id
Neighbor-id ← getNodeId(N);   //Getting neighbor Node id
MyPosition← getPosition();      // Getting current node position
NPosition← getPosition(N);      // Getting neighbor N position
t ← now();        //Getting current time
neighborsTable.upDateHonesty(Neighbor-id, NPosition, Hovr, t);        //Updating neighborsTable
sendHonestyBeacon (My-Id, MyPosition, Neighbor-id, Hovr, t);
```

Following receipt of an Honesty beacon message, the recommended confidence and the Honesty value are calculated, and the table of neighbors is then updated. Code Segment 3.3 shows the steps taken in response to receiving an Honesty beacon message.

Code Segment 3.3. Steps required when receiving an Honesty beacon message.

```
// The steps taken when receiving an Honesty beacon message
extractHonestyBeacon(Id, Position, Neighbor-id, Neighbor-Honesty, Timestamp);        //Extracting information from the beacon message
setHovr(N)[Id]←Neighbor-Honesty;     // Setting the honesty of the node N provided from the node Id
Crcd←0;
for all i in neighborsTable do
        Crcd←Crcd+getHovr(N)[i]; // Calculating the Recommended confidence by getting the honesty of the node N provided from the node i
end for
Hovr←a*Cdirect+b*Crcd;          // Calculating the overall honesty
neighborsTable.upDateHonesty(Neighbor-id, Neighbor-Position, Hovr, now()); //Updating node information in neighborsTable
```

When a node wants to send data to another node, it first encrypts the message with a signature. The integrity of the message will be maintained because of the uniqueness of this signature, which

means that it cannot be altered or modified by anyone. Then, the sender node looks at all of its neighbors, chooses the one with the greatest Honesty rating, and sends the packet to that node. The intended receiver follows the same method until the packet reaches its destination. By receiving, the receiver node checks the signature; if it is appropriate and it has been received in the necessary set time interval, the message is received; if not, it is rejected and the precedent-hop is considered as a dishonest node. The following forwarding nodes from the source to the destination are chosen as indicated in Figure 3.5.

Figure 3.5. Next-hop selection from source to destination.

The current node sends the identical packet to the two best neighbors at the same time to maximize the likelihood that the message will be appropriately received when the Honesty values of all the neighbors fall below the desired weight level. The SecE-V2X next-hop selection process is described in Code Segment 3.4.

Code Segment 3.4. The next-hop selection steps in the SecE-V2X protocol.

```
// The steps taken while deciding to deliver a data packet
selfPosition ← getPosition();     // Getting current node position
destinationId ← getNodeId(destination);                    // Getting the ID of the destination node
destinationPosition ← getPosition(destination);            // Getting the position of the destination node
myDistance ← (destinationPosition - selfPosition).length();       //Calculate the distance between the current node and the destination
if destinationId in neighborsTable then //Check if the destination in neighborsTable
```

```
        sendPacketTo(destinationId); //Sending the packet to the destination
else
        bestHonesty ← 0;
        for all i in neighborsTable do
                neighborDistance ← (destinationPosition - neighborPosition).length(); // Calculate the distance between the neighbor node and the destination
                if (myDistance > neighborDistance) then
                        if (neighborHonesty > bestHonesty) then // Choosing the neighbor with the best honesty
                                bestHonesty ← neighborHonesty;
                                bestNeighbor ← getNodeId(neighbor);
                        end if
                end if
        end for
        if (bestHonesty > weightLevel) then //Check the weight level
                sendPacketTo(bestNeighbor); // Transferring the packet to the chosen next-hop node
        else
                chooseTwoBestNeighbors(bestNeighbor1, bestNeighbor2); // choosing the best-honesty pair of nodes
                sendPacketTo(bestNeighbor1);
                sendPacketTo(bestNeighbor2);
        end if
end if
```

5. Security Analysis

Ensuring the security and integrity of the proposed protocol is of utmost importance in the context of Vehicular Ad-Hoc Networks (VANETs). In this section, we conduct a comprehensive security analysis to evaluate the effectiveness of our approach in mitigating various security threats and maintaining the confidentiality, authenticity, and integrity of the system.

- **Authentication:** To ensure authentication within our protocol, when a vehicle user enters an RSU region, the authenticated vehicle securely transmits authentication information to the RSU. This information, thoughtfully selected by the Trusted Authority (TA), is

communicated to the authenticated vehicle in an offline manner, adding an extra layer of security to the process. In parallel, the RSU forwards a dummy identity to the vehicle user. This dummy identity, originating from the TA, is meticulously safeguarded, making it a formidable challenge for any potential adversary to compromise the TA and gain access to these critical credentials. This multi-faceted approach significantly bolsters our scheme's ability to fend off authentication attacks.

- **Message Integrity:** By utilizing the hash function and comparing hash values, our approach ensures message integrity in the VANET. This guard against message tampering or modification during transmission. Any alteration to the message content would result in a mismatch between the computed hash values, alerting the receiving nodes to the tampering attempt.
- **Anonymity:** During the phase of anonymous authentication, only dummy identities are brought into play. These dummy identities, whether belonging to vehicle users or RSUs, are thoughtfully provided by the Trusted Authority (TA) during their initial offline registration. Moreover, throughout the entire data transfer process, these dummy identities remain the sole point of reference. It's important to note that these dummy identities are meticulously mapped to their corresponding real identities within the TA's domain. This strategic mapping ensures that, even if an adversary were to intercept or compromise a dummy identity, they would remain utterly ignorant of the actual, real-world identity. Consequently, our approach preserves the utmost privacy for vehicle users and RSUs, upholding the fundamental principle of anonymity while bolstering the security framework.
- **Resistance to Drop Attacks:** Our protocol incorporates advanced measures to detect and effectively mitigate drop attacks, a deceptive strategy employed by malicious nodes to intentionally discard or selectively forward messages. The cornerstone of our defense mechanism lies in the intelligent use of the Packet Loss metric within our reputation system. By meticulously analyzing the consistency of message dissemination and harnessing reputation scores that intricately consider Packet Loss as a vital component, our approach emerges as a robust guardian against nodes participating in drop attacks. This innovative approach empowers the system to discern irregularities in message propagation patterns and promptly take appropriate actions, such as isolating or flagging nodes engaged in these malicious activities.
- **Resistance to Delay Attacks:** To execute a delay or replay attack, an adversary typically attempts to send a counterfeit or altered message to the end entity within a predefined

timeframe. However, our proposed scheme employs a robust countermeasure in the form of timestamping for each transmitted message. The sender dispatches the message within a specific time window to the end entity. If, by any chance, the adversary intercepts the message and attempts to modify its content or even replace it entirely, the altered message will not reach the destination within the expected time frame. Should the received message exceed the designated time window, the end entity promptly recognizes it as invalid and subsequently discards the message. This inherent feature of our proposed approach serves as a formidable defense mechanism against delay attacks.

- **Resistance to Repudiation Attacks:** During the initial offline registration process, both vehicle users and RSUs undergo a rigorous and secure registration procedure with the Trusted Authority (TA). In this process, the entities confidentially submit their original credentials to the TA. Once the TA successfully authenticates and verifies the identities of these entities, it issues the requisite credentials, which are subsequently employed during anonymous authentication and message transfer phases. This meticulous registration and credential provisioning process ensures that vehicle users and RSUs are bound to the authenticity of their actions within the VANET system. Any attempts at repudiation are effectively nullified, as the entities' credentials, verified by the TA, serve as indisputable proof of their involvement in the system.

6. Conclusion

In this chapter, we presented a comprehensive overview of our proposed protocol for enhancing the security of Vehicular Ad-Hoc Networks (VANETs). By combining the strengths of reputation systems and blockchain technology, our protocol aims to address various security challenges and provide a robust framework for secure and trustworthy communication among vehicles. We began by introducing the key components of our protocol, including the Trusted Authority (TA), Road Side Units (RSUs), and On-Board Units (OBUs). These entities play critical roles in facilitating secure communication, authentication, and access control within the VANET environment. We then delved into the detailed design and operation of our protocol. The integration of reputation systems and blockchain technology allows us to achieve significant advancements in security. The reputation system provides a mechanism for evaluating and establishing trust among vehicles, while the blockchain ensures tamper-proof storage of reputation scores, transactional data, and other critical information. Through our protocol, we address key security concerns such as authentication, data integrity, privacy, and resistance to attacks. By leveraging cryptographic techniques, digital signatures, hashing, and decentralized consensus mechanisms, we establish a

strong foundation for secure and reliable communication within VANETs. Furthermore, we conducted a comprehensive security analysis, which demonstrated the effectiveness of our protocol in mitigating various security threats and preserving the integrity and confidentiality of data. The integration of reputation systems and blockchain technology enhances the overall resilience of the network and reduces the risk of malicious activities. It is important to note that while our proposed approach shows promise in addressing security challenges in VANETs, it is essential to conduct thorough testing, simulations, and analysis to validate its effectiveness against various known attack scenarios. The next chapter will focus on the implementation details and performance evaluation of our protocol, providing valuable insights into its feasibility and efficiency in real-world scenarios.

Chapter 4

Performance Evaluation and Simulation Results

Content

1. Introduction

This chapter is dedicated to providing a comprehensive evaluation of our proposed protocol through detailed implementation and performance analysis. Our objective is to assess the behavior and effectiveness of the protocol under different scenarios and validate its capability to enhance security in vehicular ad-hoc networks (VANETs).

To begin, we carefully examine various Network Simulators available in the literature to select the most suitable one for our specific environment. We consider factors such as flexibility, scalability, and accuracy to ensure reliable simulation outcomes. The chosen simulator will serve as the foundation for our performance evaluation. Next, we explore the widely used Mobility Generators and VANET frameworks that are instrumental in creating realistic mobility patterns and simulating VANET environments. These tools provide us with the means to generate realistic vehicular mobility scenarios, capturing the dynamics and complexities of real-world traffic situations. Once the simulation infrastructure is established, we present our Simulation Model, which encompasses the network topology, communication protocols, and security mechanisms implemented in our proposed protocol. We also introduce the models for simulating various attacks, enabling us to assess the protocol's resilience in the face of security threats.

The heart of this chapter lies in the evaluation of our protocol's performance. Through extensive simulations, we analyze its behavior under diverse scenarios, considering different network parameters, traffic conditions, and attack scenarios. We measure and evaluate key performance metrics such as packet delivery ratio, end-to-end delay, and throughput, providing quantitative insights into the protocol's performance characteristics. Furthermore, we conduct a comparative analysis by benchmarking our proposed protocol against other existing secure protocols for VANETs. This allows us to assess the protocol's effectiveness in mitigating security threats and highlight its advantages over alternative solutions.

By thoroughly examining the simulation results, we gain valuable insights into the strengths and limitations of our proposed protocol. We identify its performance under varying conditions and validate its ability to enhance security and ensure reliable communication in VANETs.

2. Network simulation

As we delve into the evaluation of vehicular networks, it is important to choose a network simulator that supports the specific characteristics and requirements of vehicular networks. In this

subsection, we compare several network simulators to identify the most appropriate choice for our evaluation.

2. 1. Network Simulators

Among the available network simulators, we specifically focus on simulators that are capable of effectively simulating vehicular networks based on the IEEE 802.11p standard. These simulators give the ability to replicate realistic vehicular communication scenarios and evaluate how well the protocols perform in different scenarios. There are various VANET simulators available, both commercial and open-source. We list the most popular simulation tools and compare their key properties in Table 4.1.

2. 1. 1. Network Simulator 2 (NS-2)

NS-2 [115], or Network Simulator Version 2, is a widely used and highly regarded open-source network simulator that has been instrumental in the field of network research. Initially developed by the VINT project research group at the University of California, Berkeley, NS-2 provides a powerful platform for simulating diverse network scenarios, both wired and wireless. The simulator is implemented using a combination of C++ and OTCL (Object-oriented Tool Command Language), where the core simulation functionality is defined in C++, while OTCL is utilized for assembling and configuring simulation objects and scheduling discrete events.

One of the notable features of NS-2 is its extensive support for various network protocols and applications. It includes a wide range of TCP variants and detailed models for specific applications like HTTP traffic. Furthermore, NS-2 offers comprehensive support for wireless network modeling, encompassing crucial aspects such as node mobility, radio propagation modeling, and routing algorithm models. It also incorporates the MAC (Medium Access Control) protocol model based on the IEEE 802.11p specification, which is specifically designed for vehicular communication.

Although NS-2 has proven to be a valuable tool in network simulation, it does have certain limitations. Scalability is one such limitation, as large-scale simulations may encounter performance challenges. Additionally, NS-2 lacks a Graphical User Interface (GUI), which can make the simulation setup and visualization more challenging for users. However, these drawbacks are outweighed by the simulator's robust capabilities and the vast amount of research conducted using NS-2 [51].

2. 1. 2. Network Simulator 3 (NS-3)

NS-3 [116], a discrete-event network simulator and open-source software, has emerged as a prominent tool for networking research and simulation. It serves as a successor to the widely used NS-2 simulator and is licensed under the GNU GPLv2, making it freely available for research and development purposes. The NS-3 project aims to provide a preferred and open simulation environment for networking research, fostering innovation and advancements in the field. NS-3 is compatible with Linux, Mac OS, and MS Windows (via cygwin), and offers scripting capabilities in both C++ and Python. Leveraging the power of C++ and Python bindings, NS-3 allows researchers to design and simulate complex network scenarios effectively.

Compared to its predecessor NS-2, NS-3 boasts several notable improvements and features. It offers enhanced scalability, performance, and realism in network simulations, allowing researchers to model a wide range of wired and wireless networks with greater accuracy. NS-3 provides a comprehensive set of network protocols, mobility models, and traffic generators, enabling researchers to analyze and evaluate the behavior and performance of network systems under various conditions.

The architecture of NS-3 is built on a modular and extensible design, allowing users to add or modify components according to their specific research requirements. It incorporates well-defined APIs and interfaces that facilitate easy integration of new modules and extensions. The simulator employs a discrete-event simulation model, where events are scheduled and executed in sequential order, enabling precise control over the simulation timeline.

While NS-3 is a powerful and widely used network simulator, it is important to be aware of its limitations. These include the learning curve associated with programming and network protocols, the lack of a built-in graphical user interface, longer execution times for complex simulations, limited protocol support, resource-intensive requirements, and the need for additional documentation and community support.

2. 1. 3. OPtimized Network Engineering Tools (OPNET)

OPNET [117], which stands for Optimized Network Engineering Tool, is a widely used commercial network simulation software. It is designed to simulate both wired and wireless networks and offers a comprehensive environment for designing, analyzing, and optimizing communication networks. OPNET provides a user-friendly graphical user interface (GUI) that enables users to define network topologies, configure parameters, and model various networking technologies. It supports a range of wireless standards, including IEEE 802.11, IEEE 802.15.1,

IEEE 802.20, and satellite networks. With OPNET, users can simulate switches, routers, servers, and protocols at different levels of the network stack. The software also includes features for application performance management, network planning and engineering, and network research and development. However, its ease of use, interactive GUI, and comprehensive documentation make it a popular choice for professionals and researchers in the field of network simulation and modeling. While OPNET is a powerful tool with extensive capabilities, it is important to note that OPNET has a few limitations worth considering. Firstly, it is commercial software, which means it may come with a cost associated with its usage. This can limit its accessibility, particularly for individuals or organizations with budget constraints. Additionally, OPNET's complexity can pose a challenge when it comes to developing specific components or customizing the simulation environment. It may require a steep learning curve and expertise in working with the software. Furthermore, as a commercial tool, the availability of technical support and updates may depend on the licensing agreement and level of customer support provided by the vendor.

2. 1. 4. Global Mobile Information system Simulator (GloMoSim)

GloMoSim is an open-source network simulation tool that was developed at the parallel computing laboratory of the University of California, Los Angeles (UCLA). It is specifically designed to simulate both wired and wireless networks with a high level of scalability. GloMoSim employs PARSEC (Parallel Simulation Environment for Complex System), a simulation language based on the C programming language, which enables the execution of discrete-event simulation models in parallel and sequential modes. With GloMoSim, users can simulate networks comprising thousands of nodes with diverse communication capabilities, including multicast, ad-hoc networking, direct satellite broadcasts, and traditional Internet protocols. It follows the OSI layer model and provides support for multiple protocols and templates at each layer. Although GloMoSim offers advanced features and the potential for parallel processing, it has some limitations, such as the lack of extensive documentation and no longer being actively supported. However, its ability to handle large-scale network simulations makes it a valuable tool for researchers and developers interested in studying wireless network scenarios and exploring various communication technologies [118].

2. 1. 5. Quality Networking (QualNET):

QualNet is a cutting-edge network simulator developed by Scalable Network Technologies, designed to simulate large and complex networks. It serves as a comprehensive tool for evaluating network performance and analyzing various network scenarios. Built on the foundation of GloMoSim, QualNet offers enhanced capabilities and runs seamlessly on multiple operating systems such as UNIX, Windows, MAC, and Linux. Its simulation models are implemented in

C++, providing efficient and accurate network evaluations. QualNet supports a wide range of network types, including Wi-Fi, sensor networks, MANET, WiMAX, and more, enabling researchers to explore different network architectures and protocols. The simulator is equipped with a rich set of model libraries, including wireless libraries like 802.11ac, 802.11ax, and 802.11n, as well as cellular libraries for GSM and UMTS networks. Additionally, QualNet incorporates advanced features such as parallel execution and optimized lookahead, ensuring high-fidelity simulations. However, it should be noted that as a commercial simulator, QualNet may incur costs for users, and the availability of updated models could be a limitation [119].

2. 1. 6. (JIST/SWANS)

JIST/SWANS, an acronym for Java in Simulation Time/Scale Wireless Ad Hoc Network Simulator, is a powerful and efficient discrete event simulator. It utilizes a Java virtual machine to execute simulations, enabling simulation code to be written in the widely-used Java programming language. This approach simplifies the development process as it eliminates the need for specialized languages or system calls. JIST/SWANS offers high-performance simulation capabilities, surpassing many existing simulation runtimes in terms of both time and memory consumption. It serves as a prototype for building general-purpose discrete event simulators and employs a virtual machine-based simulation paradigm.

SWANS, which stands for Scale Wireless Ad Hoc Network Simulator, is built on top of the JIST platform and focuses on simulating wireless networks. It addresses the limitations of previous network simulation tools and provides a scalable solution for simulating large-scale wireless networks. SWANS consists of independent software components that can be combined to form complete wireless networks or sensor networks. With SWANS, researchers and developers can simulate networks of significantly larger sizes compared to other simulators like GloMoSim and ns-2, while maintaining simulation throughput and memory requirements.

By leveraging the capabilities of JIST, SWANS achieves superior simulation performance, enabling the execution of standard Java network applications over simulated networks. This integration of JIST and SWANS offers researchers and developers a versatile and efficient platform for wireless network simulations. It provides an effective environment for studying and analyzing the behavior of wireless networks, facilitating the development and evaluation of novel protocols and algorithms.

The limitations of JIST/SWANS are primarily related to its lack of updates and limited documentation availability. The absence of recent updates may hinder its compatibility with newer

technologies and protocols, while the lack of a mobility model for VANET restricts its applicability in vehicular communication scenarios. Additionally, the outdated status of the simulator may result in limited technical support and scalability constraints, although SWANS is designed to simulate larger networks compared to other simulators [120].

2. 1. 7. JAVA SIMULATOR (J-SIM)

J-SIM is an object-oriented and open-source simulator that follows the Autonomous Component Architecture (ACA) and is primarily implemented using Java and scripting languages like TCL, Perl, and Python. It provides a powerful framework for building and analyzing network simulations. Models in J-SIM are developed using Java, allowing for the creation of robust and extensible simulations, while scripting languages are employed for simulation configuration and control during execution. This combination of languages enables J-SIM to offer a wide range of modeling capabilities and flexibility.

J-SIM stands out by incorporating features from popular simulators such as NS-2 and OMNeT++, providing a comprehensive solution for network simulation. The autonomous component architecture in J-SIM allows for the creation of reusable and modular components, promoting code reusability and enhancing the efficiency of simulation development. The simulator's object-oriented approach, coupled with its support for scripting languages, facilitates the creation and customization of network models, making J-SIM a versatile tool for simulating complex network scenarios.

One of the notable strengths of J-SIM is its ability to handle large-scale simulations. The autonomous component architecture, combined with the performance of the underlying Java virtual machine, ensures efficient execution and scalability. Additionally, J-SIM offers a wide range of modeling capabilities, including support for various network protocols and simulation scenarios.

However, it is worth mentioning that J-SIM has some limitations. The availability of comprehensive documentation and user support for J-SIM may be limited compared to other commercial simulators. Additionally, while J-SIM provides a robust framework, it may require a steep learning curve for users who are not familiar with Java or scripting languages [51].

2. 1. 8. OBJECTIVE MODULAR NETWORK TESTBED in C++ (OMNeT++)

OMNeT++ [121] is a powerful and versatile discrete event simulation framework built on the C++ programming language. It serves as a modular and extensible simulation library primarily designed for developing network simulators but also finds applications in other domains such as complex

IT model simulation, multiprocessor modeling, and queuing systems. The framework was developed by Andras Varga at the Budapest University of Technology and aims to bridge the gap between open-source simulators like NS-2 and commercial options like OPNET. One of the notable strengths of OMNeT++ is its component-based and hierarchical architecture, allowing developers to create simulations using reusable modules and building blocks. This modular approach enhances code reusability and scalability, making it easier to model and simulate complex systems.

OMNeT++ employs the concept of Network Description (NED) files to define the structure and behavior of simulation models. These files, which can be edited using text or graphical interfaces, represent the relationships between modules and communication links. The framework also provides a Graphical Network Editor (GNED) equipped with a NED compiler, command-line interface (Cmdenv), graphical interface (Tkenv), and performance analysis tools (Plove). This comprehensive toolset enables users to visualize, debug, and analyze simulation models efficiently.

To enhance its functionality, OMNeT++ offers the INET Framework as its primary protocol model library. INET consists of a wide range of pre-built models and components representing the Internet protocol stack, including TCP, IP, Ethernet, and more. The INET Framework is actively maintained by the OMNeT++ team and benefits from contributions and patches from the community, ensuring its continuous evolution and adaptability to emerging networking technologies and standards.

OMNeT++ is known for its broad user base and vibrant ecosystem. Researchers and developers worldwide have contributed various simulation models and frameworks over the years, resulting in a rich collection of open-source projects and independent modules. This community-driven approach has led to the development of specialized frameworks like the Mobility Framework, which focuses on wireless and mobile networks, and other frameworks for specific domains like vehicular networks, overlay/peer-to-peer networks, and LTE networks.

Supported on multiple platforms such as Windows, Linux, and macOS, OMNeT++ leverages standard C++ and is compatible with modern C++ compilers. Its Integrated Development Environment (IDE) provides a user-friendly interface for model development, simulation execution, and result analysis. The simulation capabilities of OMNeT++ have been extensively validated, showcasing its efficiency in terms of throughput, simulation time, and memory usage compared to other popular simulators like NS-2.

The combination of OMNeT++'s modular architecture, specialized frameworks, comprehensive libraries, community support, and visualization capabilities make it a preferred choice for simulating VANETs. It provides researchers and developers with a powerful and flexible platform to investigate and analyze various aspects of VANET protocols, mobility patterns, and communication strategies in a realistic and scalable simulation environment. OMNeT++ is often preferred for simulating Vehicular Ad hoc Networks (VANETs) due to its several advantages and capabilities that make it well-suited for this domain:

- **Modular and Extensible Framework:** OMNeT++ follows a modular and component-based architecture, allowing developers to easily create and customize simulation models specifically tailored for VANETs. The framework's flexibility enables the incorporation of various VANET-specific protocols, mobility models, and communication patterns.
- **INET Framework:** OMNeT++ provides the INET Framework, which includes a comprehensive set of models and components for simulating the Internet protocol stack. This library offers pre-built models for protocols commonly used in VANETs, such as TCP, IP, and IEEE 802.11p, the dedicated wireless standard for vehicle-to-vehicle (V2V) communication. These ready-to-use models significantly simplify the process of VANET simulation development.
- **Mobility Framework:** OMNeT++ offers the Mobility Framework, specifically designed for wireless and mobile network simulations. This framework provides detailed models for mobility patterns, radio propagation, and MAC protocols, essential components in accurately representing the movement and communication behavior of vehicles in VANETs.
- **Community Support and Development:** OMNeT++ benefits from a vibrant and active community of researchers and developers. This community actively contributes to the development of VANET-related simulation models and frameworks, expanding the capabilities of OMNeT++ for VANET simulations. Furthermore, this collective effort ensures that the framework remains up-to-date with the latest VANET research advancements and standards.
- **Visualization and Analysis Tools:** OMNeT++ provides a range of visualization and analysis tools that facilitate the interpretation and evaluation of VANET simulation results. These tools enable users to visualize the network behavior, analyze performance metrics, and gain insights into the effectiveness of VANET protocols and algorithms.

- **Performance and Scalability:** OMNeT++ is known for its efficient simulation engine, which allows large-scale simulations with numerous vehicles in VANET scenarios. It offers options for optimizing simulation performance, such as parallel execution and distributed simulation, enabling researchers to simulate realistic VANET deployments efficiently.
- **Integration and Interoperability:** OMNeT++ supports integration with external tools and libraries, allowing researchers to leverage existing VANET simulation frameworks, mobility traces, or traffic generators. This interoperability enhances the versatility of OMNeT++ for VANET simulation and facilitates comparisons and collaborations with other simulation environments.

Table 4.1. Networking simulators comparison

	NS-2	NS-3	OPNET	GloMoSim	QualNET	JIST/SWANS	J-SIM	OMNeT++
License	Open-source	Open-source	Commercial	Open-source	Commercial	Open-source	Open-source	Open-source
Simulation language	C++ & OTCL	C++ & Python	C++ & OTCL	C	Parsec C++	Java	Java	C++
GUI Support	Poor	Poor	Excellent	Poor	Excellent	Poor	Poor	Good
Operating System	Linux, macOS, Windows	Linux, macOS, Windows, FreeBSD	Linux, macOS, Windows	Linux, macOS, Windows	Linux, Windows	Linux, macOS, Windows	Linux, macOS, Windows	Linux, macOS, Windows
802.11p support	NS-2.33	Yes	Yes	No	Yes	No	No	Yes
Ease of Use	Moderate	Hard	EASY	Moderate	Moderate	Hard	Hard	Easy
Continuous development	NS-3	Yes	Yes	No	Yes	Yes	No	Yes
Parallel Processing	No	Yes	Yes	Yes	Yes	Yes	No	Yes
Scalability	Poor	high	High	High	High	High	Moderate	high

2. 2. Mobility Generators

The Mobility Generators in network simulators allow researchers and developers to simulate dynamic environments realistically. By incorporating accurate mobility patterns, they can evaluate the performance of routing protocols, mobility management schemes, handover mechanisms, and other mobility-dependent algorithms in different scenarios. In network simulators, the Mobility Generators offer different techniques and models to simulate node movements and mobility patterns. These generators provide flexibility in defining node mobility characteristics such as speed, direction, pause times, and mobility models. These models enable the simulation of various mobility scenarios such as vehicular networks, pedestrian movements, or animal migrations. They

can be customized to reflect specific mobility patterns or real-world traces obtained from mobility datasets.

Many variables, including vehicle density, road topology, traffic signals, number of lanes, vehicle speed, obstructions on the road, and driver behavior and habits, affect the mobility generation in VANET. To provide real-time mobility for vehicles, many traffic simulations or mobility generators are available. However, not all network simulators support the results that these mobility generators produce. A comparison of the discussed mobility generators' properties is shown in Table 4.2.

2. 2. 1. Simulation of Urban Mobility (SUMO)

Simulation of Urban Mobility, or SUMO [122], is an open-source microscopic traffic simulator created to simulate and examine road networks. It offers a range of powerful features that make it a valuable tool for traffic simulation. With SUMO, users can create detailed simulations of urban traffic scenarios, including various types of vehicles, pedestrians, and public transport. The simulator supports the import of network data from popular formats such as OpenStreetMap, VISUM, VISSIM, NavTeq, and more, allowing users to replicate real-world road networks accurately.

One of the key strengths of SUMO is its ability to handle large-scale road networks efficiently. Whether it's a small neighborhood or a complex city layout, SUMO can handle networks of any size, enabling researchers and practitioners to study traffic dynamics in various urban environments. The simulator also provides advanced features like collision avoidance, lane changing, and right-of-way rules, ensuring realistic vehicle behavior and interactions.

SUMO offers a user-friendly Graphical User Interface (GUI) that simplifies the simulation setup and provides an intuitive visualization of the simulated traffic. Users can easily configure parameters, define vehicle routes, set traffic flow patterns, and monitor the simulation in real-time through the GUI. Furthermore, SUMO provides additional tools and APIs for route finding, network import, emission calculation, and remote control of simulations, enhancing its versatility and adaptability to different research needs.

Being an open-source project, SUMO benefits from an active and supportive community that continuously contributes to its development. It is implemented in C++ and Python, utilizing portable libraries, which ensures its cross-platform compatibility across Windows, Linux, and macOS operating systems. Additionally, SUMO can be seamlessly integrated with other simulation

frameworks and network simulators like OMNeT++, NS-2, and NS-3, allowing users to combine traffic simulations with network dynamics for comprehensive analysis.

2. 2. 2. MObility model generator for VEhicular networks (MOVE)

It is a powerful tool that provides researchers and practitioners with the ability to rapidly generate realistic mobility models for VANET (Vehicular Ad hoc Network) simulations. Built on top of the widely used SUMO (Simulation of Urban MObility) micro-traffic simulator, MOVE offers an efficient solution for creating mobility trace files that accurately capture the movement patterns of vehicles in a realistic manner [123].

One of the key strengths of MOVE is its integration with SUMO, which allows users to leverage the advanced traffic simulation capabilities provided by SUMO while focusing on generating mobility models specifically tailored for VANET scenarios. By utilizing the rich features of SUMO, MOVE ensures that the generated mobility traces are based on real-world traffic dynamics, taking into account factors such as road networks, traffic flow, and vehicle behavior.

MOVE also provides a user-friendly Graphical User Interface (GUI) that simplifies the process of generating simulation scenarios. With the GUI, users can easily define various parameters, including road topologies, vehicle routes, departure times, destinations, and other mobility-related attributes. This eliminates the need for manual scripting and allows researchers to quickly generate complex and realistic mobility models without delving into the intricacies of the underlying simulation tools.

Furthermore, MOVE is compatible with popular network simulation tools such as ns-2 and GloMoSim, enabling seamless integration of the generated mobility models into larger network simulations. This facilitates comprehensive evaluations of VANET protocols, algorithms, and applications, providing insights into their performance under realistic mobility conditions.

2. 2. 3. Vehicular Ad hoc Network Mobility Simulator (VANETMobiSim)

VANETMobiSim [124] is a feature-rich traffic generator designed specifically for VANET (Vehicular Ad hoc Network) simulations. Built on the foundation of the CANU Mobility Simulation Environment (CanuMobiSim), VANETMobiSim extends its capabilities by offering advanced mobility modeling for vehicular networks.

Developed in Java, VANETMobiSim enables the generation of realistic traffic patterns at both the microscopic and macroscopic levels. It supports the import of real geographical maps, allowing

users to simulate traffic scenarios in real-world locations. This feature enhances the authenticity and applicability of the simulation results to practical VANET deployments.

One of the standout features of VANETMobiSim is its support for pre-compiled intelligent driving models. These models, such as the Intelligent Driving Model with Lane Changing and Intelligent Driving Model with Intersection Management, enhance the behavior of vehicles in the simulation by incorporating realistic driving patterns and interactions.

The output generated by VANETMobiSim is compatible with popular network simulators, including NS-2, GloMoSim, and QualNet. This seamless integration enables researchers to seamlessly incorporate the generated traffic scenarios into their preferred simulation environment, facilitating comprehensive evaluations of VANET protocols, algorithms, and applications.

At the macroscopic level, support for multi-lane roads, distinct directional flows, differentiated speed limits, and intersection traffic signs are provided by VANETMobiSim. This level of detail allows for the accurate representation of complex road networks and traffic dynamics.

Moreover, VANETMobiSim introduces new mobility models that capture realistic car-to-car and car-to-infrastructure interactions. These models consider factors such as nearby moving vehicles, overtaking maneuvers, and adherence to traffic signals, ensuring a high level of realism in the simulation.

2. 2. 4. City Mobility (CityMob)

CityMob is a versatile traffic generator and mobility model generator specifically designed for Vehicular Ad hoc Networks (VANETs). It offers various mobility models, including the Simple Model (SM), Manhattan Model (MM), and Downtown Model (DM). The Simple Model provides a straightforward representation of vehicle movements, while the Manhattan Model emulates a grid-based road network similar to that found in Manhattan. The Downtown Model aims to simulate a realistic urban environment by incorporating factors such as traffic density, two-way lanes, and traffic signals. CityMob allows for the customization of parameters such as blocking size, traffic distribution, vehicle movement speed, and vehicle queuing. It supports the generation of mobility traces compatible with the NS-2 network simulator, enabling researchers and developers to evaluate and analyze VANET scenarios effectively. CityMob is an open-source simulator implemented in C programming language, providing flexibility and extensibility for further enhancements and research in the field of vehicular networks [118].

2. 2. 5. Street Random Waypoint (STRAW)

STRAW, which stands for STreet RAndom Waypoint, is a traffic generator that aims to provide realistic simulation results by incorporating the mobility patterns observed in real US cities. It utilizes a vehicular mobility model that takes into account actual vehicle traffic behavior, allowing for accurate simulations of network scenarios. The current implementation of STRAW is specifically designed for the JiST/SWANS discrete event simulator, making it compatible with this particular network simulator. However, in order to use STRAW with other network simulators such as NS-2, modifications are required to incorporate its output effectively. STRAW is an essential component of the C3 (Car-to-Car Cooperation) project, which focuses on enhancing communication and cooperation among vehicles. By utilizing a realistic mobility model that considers real-world traffic conditions, STRAW enables researchers and network simulation practitioners to study and evaluate network protocols and algorithms in scenarios that closely resemble actual urban environments [125].

2. 2. 6. FreeSim

FreeSim is a versatile and customizable traffic simulator that operates at both the microscopic and macroscopic levels. It allows users to generate realistic traffic scenarios by incorporating various traffic algorithms for single and multiple vehicles on multiple lanes. FreeSim also supports the utilization of real-time data, enabling the simulation of dynamic traffic patterns based on actual conditions. One of the key advantages of FreeSim is its integration with Intelligent Transportation Systems (ITS), enabling vehicles to communicate with highway monitoring systems. This functionality enhances the simulation of advanced traffic management and control strategies. FreeSim is an open-source tool released under the GNU General Public License, providing users with free access to the source code for customization and further development. With its adaptability, comprehensive features, and support for ITS, FreeSim is a valuable tool for researchers and practitioners in the field of transportation planning and analysis [123].

Table 4.2. Mobility generators comparison.

	SUMO	**MOVE**	**VANETMobiSim**	**CityMob**	**STRAW**	**FreeSim**
License	Open source	Open source	Open source	Open source	Open source	Open source
GUI	Yes	Yes	Yes	Yes	Yes	Yes
Continuous development	Yes	No	No	Yes	No	-
Ease of use	Moderate	Moderate	Easy	Easy	Moderate	Easy
Available examples	Yes	Yes	Yes	No	No	Yes

Trafic Model Type	Microscopic	Microscopic	Microscopic	Microscopic	Microscopic	Macroscopic, Microscopic
Map Support	Real, User defined, Random	Real, User defined, Random	Real, User defined, Random	Random	Real	Real
Lane change	Yes	Yes	Yes	Yes	Yes	No
Simulator Compatibility	NS2, NS3, and OMNeT++	NS-2, GloMoSim, and QualNet.	NS-2, OMNeT++,GloMoSim, and QualNet.	NS-2	SWANS	-
Features	collision avoidance, multiple vehicles types, Traffic intersections, speed control	collision avoidance, multiple vehicles types, Traffic intersections, speed control	speed control, Traffic intersections	collision avoidance, multiple vehicles types, speed control	speed control	Speed control

2. 3. Vehicular network simulators and frameworks

Mobility patterns and network simulators are integrated with vehicular network simulators and frameworks to provide a unified platform for simulations. By using these tools, running various software applications and managing their dependencies separately is no longer necessary. Researchers can analyze the behavior and performance of VANET networks in a realistic way by adding models for vehicle movements, traffic patterns, and communication protocols. These simulators include features like building network topologies, creating realistic mobility traces, and displaying simulation results. They simplify the simulation process, allowing researchers to concentrate on specific network aspects and create effective solutions for future vehicular communication systems. The main characteristics comparison of the widely used vehicular network simulators and framework are shown in Table 4.3.

2. 3. 1. Vehicles in Network Simulation (VEINS)

VEINS (Vehicles in Network Simulation) [126] is an open-source framework specifically designed for simulating vehicular networks with a high degree of realism and accuracy. It combines the SUMO (Simulation of Urban Mobility) traffic simulator with the OMNeT++ network simulator, creating a powerful and integrated platform for conducting comprehensive vehicular network simulations.

The integration of SUMO and OMNeT++ in VEINS allows for the seamless modeling of both the mobility patterns of vehicles on the road and the communication protocols and algorithms

of the network. This bidirectional coupling enables the framework to accurately capture the dynamic influence of road traffic on network traffic, as well as vice versa. For example, vehicles in the simulation can dynamically adjust their routes or alter their mobility patterns in response to network events, such as the reception of warning messages from neighboring vehicles or roadside infrastructure.

VEINS utilizes TCP communication and the TraCI (Traffic Control Interface) protocol to establish efficient and reliable communication between the SUMO mobility generator and the OMNeT++ network simulator. This integration ensures synchronized and real-time interaction between the two components, facilitating a realistic and dynamic simulation environment.

One notable advantage of VEINS is its ability to utilize real maps for road layout plans. This means that researchers can incorporate actual road networks and urban environments into their simulations, enhancing the realism and accuracy of the results. Additionally, VEINS offers an environmental application that monitors the emission of carbon dioxide gas by the vehicles in the simulation, allowing for the analysis of environmental impacts and sustainability aspects.

With VEINS, researchers and developers have access to a wide range of models and tools for simulating various aspects of vehicular networks, including vehicle mobility, communication protocols, traffic control, and environmental factors. The framework supports online reconfiguration and rerouting of vehicles, enabling the evaluation of dynamic scenarios and adaptive network behavior. Furthermore, VEINS is compatible with multiple operating systems, such as Linux, Mac, and Windows, and benefits from the user-friendly GUI provided by the OMNeT++ simulator, making it accessible and intuitive for simulation setup, execution, and analysis. With its compatibility across multiple operating systems and the benefits of the OMNeT++ GUI, VEINS offers a user-friendly and powerful solution for realistic vehicular network simulations, contributing to advancements in intelligent transportation systems and network research. The architecture of the VEINS framework is shown in Figure 4.1 following.

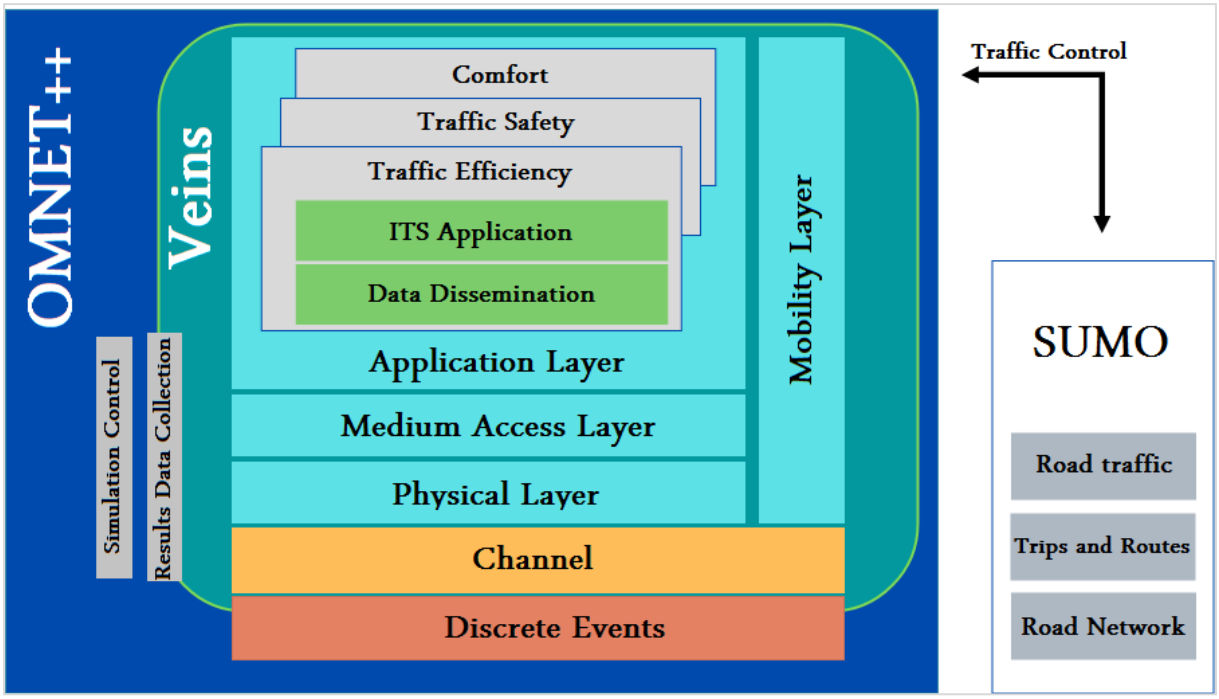

Figure 4.1. Veins Framework architecture

2. 3. 2. Traffic and Network Simulation Environment (TraNS)

TraNS is a comprehensive and open-source simulation environment designed for creating realistic simulations of Vehicular Ad-Hoc Networks (VANETs). It integrates two powerful components: SUMO, a mobility generator tool, and NS-2, a network simulator, to facilitate the development and execution of VANET simulations. By combining the mobility generator and network simulator, TraNS enables bidirectional feedback between vehicle activity and the mobility model, ensuring a more accurate representation of real-world experiments [120].

One of TraNS's key objectives is to bridge the gap between simulation results and real-world implementations by closely mimicking real VANET environments. It achieves this by using SUMO to generate realistic mobility patterns and NS-2 to simulate the network behavior. This integration allows for the exchange of information at runtime within VANETs, influencing the behavior of vehicles based on network events and vice versa.

Implemented in Java and C++, TraNS offers compatibility with both Linux and Windows platforms, providing flexibility for users. It supports both network-centric and application-centric operational modes. There is no direct feedback between the SUMO mobility generator and the NS-2 network simulator in the network-centric mode. Instead, mobility traces are transferred through a parser from SUMO to NS-2, enabling the simulation of large-scale networks with

thousands of vehicles. The parser converts the SUMO traces into a format compatible with NS-2, facilitating seamless communication between the two components.

TraNS also boasts visualization capabilities by leveraging the Google Earth API, allowing users to visualize and analyze simulation results in a geospatial context. This feature enhances the understanding of VANET dynamics and facilitates the identification of potential issues and optimization opportunities.

However, it's important to note that TraNS is not currently under active development, and its last release (TraNS V1.2) dates back to 2009. Consequently, it may not be compatible with the latest versions of SUMO and NS-2, limiting its applicability to specific environments and use cases. Nonetheless, TraNS remains a valuable tool for researchers and practitioners interested in conducting VANET simulations, offering a foundation for investigating traffic patterns, evaluating network protocols, and exploring the impact of various mobility models and applications on VANET performance.

2. 3. 3. Groove-based Vehicular Network Simulation Environment (GrooveNet)

GrooveNet is an open-source hybrid simulator that offers a unique and powerful platform for simulating vehicular networks. It combines both a mobility generator and a network simulator, allowing for seamless communication between simulated vehicles and real vehicles. The simulator is designed with a modular event-based architecture and provides a user-friendly graphical user interface (GUI), making it easy to configure and execute simulations [118].

One notable feature of GrooveNet is its ability to the Tiger/Line database's collection of real street maps. This capability enhances the realism of the simulations by providing an accurate representation of the road network. Researchers can leverage this feature to study and analyze various scenarios in real-world environments.

The modular architecture of GrooveNet includes models for trip generation, message broadcasting, and mobility patterns. It also incorporates a variety of link and physical layer communication models, enabling the simulation of different communication protocols and technologies.

One of the key strengths of GrooveNet is its support for Inter-Vehicular Communication (IVC). By modeling IVC through a practical street map topology, the simulator enables researchers to design and evaluate communication protocols for vehicular networks. This includes the

deployment of in-vehicle communication systems and the analysis of their performance under realistic conditions.

However, it's important to note that GrooveNet does have certain limitations. Firstly, it does not support the latest versions of the underlying simulation tools it integrates, which may limit its compatibility with newer features and updates. Additionally, GrooveNet is not actively under development, which means it may not receive regular updates or improvements.

Furthermore, while GrooveNet offers visualization capabilities, the level of detail and realism in the visual representations may not be as advanced as in some other simulators. Researchers seeking highly detailed visualizations may find other simulators more suitable for their needs.

Lastly, while GrooveNet supports the simulation of a large number of vehicles, it may have scalability limitations when simulating extremely large-scale vehicular networks with a significant number of vehicles and complex communication scenarios.

Despite these limitations, GrooveNet remains a valuable tool for simulating vehicular networks. Its integration of mobility generation, network simulation, and real vehicle communication provides researchers and practitioners with a versatile platform for studying and evaluating different aspects of vehicular networks. By acknowledging its limitations and working within its capabilities, researchers can leverage GrooveNet effectively to gain insights into the behavior and performance of vehicular networks.

2. 3. 4. National Chiao Tung University network simulator (NCTUns)

NCTUns is an integrated network simulator and emulator that has been widely used since its development in 2002. One of the key features of NCTUns is its novel kernel reentering simulation, which allows for the implementation of a real-life TCP/IP protocol stack in the Linux kernel. This enables high-fidelity simulation outcomes. In its 6.0 version, NCTUns implemented the IEEE 802.11P protocol and offers various capabilities such as agent-based and module-based mobility control for vehicles, different vehicle mobility models, road network construction, and simulation and emulation of devices equipped with various radio technologies [125].

NCTUns supports the simulation of roadside units (RSUs) and on-board units (OBUs) with radios including IEEE 802.11 infrastructure/ad-hoc mode, GPRS, DVB-RCST, and IEEE 802.11p/1609. It implements vehicle mobility models such as a strolling vehicle model where vehicles randomly turn at road intersections, as well as landmark-based models where the map is divided into grids and vehicles probabilistically move towards landmarks. The road network can

be user-defined or imported from a SHAPE file. Additionally, NCTUns supports wireless infrastructure on both vehicles and RSUs.

NCTUns is known for its integration of the network simulator and mobility generator into a single module, extending the capabilities of the network simulator. It utilizes the Linux TCP/IP stack, ensuring high-fidelity simulation data. With its GUI interface, users can customize network topologies, modify modules within nodes, and visualize packet transmission through animation and graph plotting. It also has the capability to execute parallel simulations on multicore computing machines.

However, NCTUns does have some limitations. Firstly, it operates primarily on the Linux Fedora platform, limiting its compatibility with other operating systems. Additionally, NCTUns is based on C++ programming, which may present challenges for users who are not familiar with this programming language. Furthermore, while NCTUns offers a graphical user interface (GUI) with features such as a packet animation player, performance monitor, node editor, and topology editor, the GUI may not be as user-friendly or intuitive as other network simulation tools.

Another aspect to consider is that NCTUns underwent a transformation in 2011 when it transitioned into commercial software and was renamed EstiNet. This shift to commercialization may have implications for the availability of support and updates for the open-source version of NCTUns, potentially limiting its long-term development and compatibility with newer technologies.

Despite these limitations, NCTUns remains a valuable tool for network-related research and simulation. It offers a wide range of functionalities and the ability to simulate real-life scenarios, making it a useful resource for network protocol design and evaluation.

2. 3. 5. EstiNet

EstiNet is a robust and versatile network simulator and emulator that originated from the renowned NCTUns platform. With a focus on network-related research and development, EstiNet provides a comprehensive set of tools for testing, evaluating, and analyzing various networking scenarios. Its user-friendly GUI environment offers an intuitive interface for constructing and managing simulated networks, making it accessible to both researchers and practitioners. EstiNet's capabilities extend beyond simulation, as it also supports emulation, allowing connections to real-world network devices for a more realistic assessment of network behavior. By simulating all five layers of the TCP/IP protocol stack, EstiNet enables in-depth analysis of network protocols,

performance, and behavior. It adopts Linux-based programs, leveraging the reliability and popularity of Linux as an operating system for networking equipment, to accurately replicate real-world network dynamics. EstiNet is a versatile tool that can be utilized for a wide range of network scenarios, including vehicular networks, wireless communication, and Internet of Things (IoT) applications. However, it's important to note that EstiNet, like any simulator, has certain limitations, such as potential scalability constraints, computational resource requirements, and the need for carefully crafted simulation models to accurately represent complex network behaviors. Nonetheless, EstiNet remains a valuable asset in the field of network simulation and emulation, empowering researchers and practitioners to explore, analyze, and optimize network designs and protocols [120].

2. 3. 6. MobiREAL

It is a network simulator that places a strong emphasis on simulating realistic mobility in mobile ad-hoc networks (MANETs). It utilizes a rule-based model to depict the behavior and movement of nodes, allowing for dynamic and realistic simulations. This rule-based probabilistic mobility model is particularly suited for simulating VANET networks and is easier to implement compared to other simulators. MobiReal consists of two independent simulators: the MobiREAL Behavioral simulator, which simulates the behavior of mobile nodes, and the MobiREAL network simulator, which facilitates data exchange among the nodes. These simulators communicate through a TCP channel, enabling a comprehensive simulation of MANETs and recently extending support to vehicular network simulations [123].

By employing a realistic mobility model, MobiReal provides a new approach to modeling and simulating the movement of nodes in MANETs. This allows for the evaluation of MANET applications in more realistic environments, giving insight into infrastructure, routing protocols, and network application performance that is difficult for other simulators to analyze. MobiReal's rule-based model, inspired by human behavior cognitive modeling, enables researchers to define how mobile nodes adjust their destinations, roads, speeds, and other factors based on their location, surroundings, and information collected from applications.

Furthermore, MobiReal is capable of simulating both vehicular and human mobility. It incorporates mobility support facilities into the Georgia Tech Network Simulator (GTNetS), allowing for the inclusion of mobile node dynamics such as joining/leaving, movement, and packet collisions.

One limitation of MobiReal is its dependency on external components and proprietary software. For vehicular mobility, it relies on NETSREAM, a non-open-source release from TOYOTA Motors, which may restrict its usage and availability. Additionally, incorporating other traffic simulators for vehicular mobility functionality requires additional integration efforts and compatibility considerations.

Another limitation is the scope of its simulation capabilities. While MobiReal excels at simulating the mobility and behavior of nodes, it may not provide the same level of depth and complexity in other aspects, such as network protocols, infrastructure, or specific application performance evaluation. Researchers using MobiReal should be aware that its primary focus is on mobility modeling, and additional tools or simulations may be needed to address other aspects of network evaluation comprehensively.

2. 3. 7. VEhicular NeTwork Open Simulator (VENTOS)

VENTOS is an advanced and versatile simulator specifically designed for studying Vehicular Ad-hoc Networks (VANETs). It integrates the SUMO mobility generator and the OMNeT++ network simulator to provide a comprehensive simulation environment for evaluating VANET traffic flow and vehicle-to-infrastructure interactions. VENTOS supports DSRC (Dedicated Short Range Communication) enabled wireless communication, enabling seamless V2I (Vehicle-to-Infrastructure) communication. This allows for efficient and reliable exchange of messages using protocols like SNMP (Simple Network Management Protocol). The simulator incorporates dynamic traffic routing algorithms that leverage real-time traffic information within the VANET, aiming to reduce average delays and fluctuations in average vehicle speeds across the network. VENTOS also includes a robust adversary module, enabling the simulation of security attacks to assess the network's resilience and evaluate potential countermeasures. Additionally, VENTOS features tools for automatic incident detection on highways and utilizes Matlab scripts for visualizing and analyzing simulation scenarios. It provides researchers and practitioners with a powerful platform to investigate VANET behavior, test various protocols and strategies, and gain insights into the performance and effectiveness of VANET applications. However, it's important to consider the specific limitations and constraints of VENTOS in terms of scalability, computational requirements, and the scope of its simulation models and scenarios [127].

Table 4.3. VANET Networking frameworks and simulators comparison.

VANET Simulators	VEINS	TraNS	GrooveNet	NCTUns	EstiNet	MobiREAL	VENTOS
License	Open Source	Open Source	Open Source	Open Source	Commercial	Proprietary	Open Source
GUI	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Continuous development	Yes	No	No	No	Yes	No	Yes
Ease of use	Moderate	Moderate	Hard	Hard	Hard	Hard	Moderate
Network simulator	OMNET++	NS-2	-	-	-	GTNets	OMNET++
Mobility Generators	SUMO	SUMO	Inbuilt	Inbuilt	Inbuilt	Inbuilt	SUMO
Traffic Model	Microscopic	Microscopic	Microscopic	Microscopic	Microscopic	Microscopic	Microscopic & Mesoscopic

3. Simulation Experiments

In the Simulation Experiments, we employed several tools and parameters to create a realistic network scenario and evaluate the effectiveness of our proposed SecE-V2X protocol. Using OMNeT++, we were able to define the network topology, node configurations, and communication protocols within the simulated environment.

3. 1. Network model

To establish a reliable and accurate simulation environment, we carefully selected the appropriate parameters. These parameters include network size, node density, and communication range. By fine-tuning these parameters, we aimed to create a network scenario that closely resembles real-world VANET communication scenarios.

We utilized the Veins framework to configure the network simulation with vehicles. The Veins framework provides an integration of OMNeT++ and SUMO, allowing for realistic vehicular network simulations. We created nodes in the form of standard wireless INET nodes, using the regular INET network card. For the VANET scenario, we employed the IEEE802.11p network card, which is specifically designed for vehicular communication.

We conducted each simulation for a duration of 500 seconds, both with and without attacks. The node density in our experiments was set to 100 nodes, representing a realistic scenario with a moderate number of vehicles in the network. The main parameters of our simulation are summarized in Table 4.4.

Table 4.4. Simulation parameters.

Parameter	Value
OMNET++ version	5.2.6
INET version	4.2.2
Veins version	5.0
Environment	Urban
SUMO version	1.3.1
Simulation area	2500 m × 2500 m
Simulation time	500 s
Number of nodes	100
Number of attackers	0, 3, 6, and 9
Max node speed	14 m/s
Mobility model	Erlangen
MAC protocol	IEEE 802.11p
Transmission range	250 m

By configuring the simulation parameters in this manner, we were able to evaluate the performance of our proposed protocol under various traffic conditions and the presence of attacks. This allowed us to assess the protocol's effectiveness in ensuring secure and reliable communication in VANETs.

3. 2. Mobility model

To generate realistic mobility and traffic scenarios for our simulation, we employed the simulator for urban mobility (SUMO). SUMO provided us with the capability to create accurate urban mobility patterns for the nodes in our simulation. Specifically, we designed our simulation scenario to represent an urban environment in Erlangen with dimensions of 2500 m × 2500 m.

The mobility model in our simulation relied on SUMO-generated mobility, which closely emulates the characteristics of roadways in the city map. The nodes, representing vehicles in the network, traveled within this urban environment based on the mobility patterns generated by SUMO (See Figure 4.2).

The speeds of the nodes varied from 0 to 14 m/s, encompassing a range of different velocities commonly observed in urban traffic scenarios. This variation in speeds added realism to the simulation, allowing us to study the impact of different vehicle velocities on the performance of our proposed protocol.

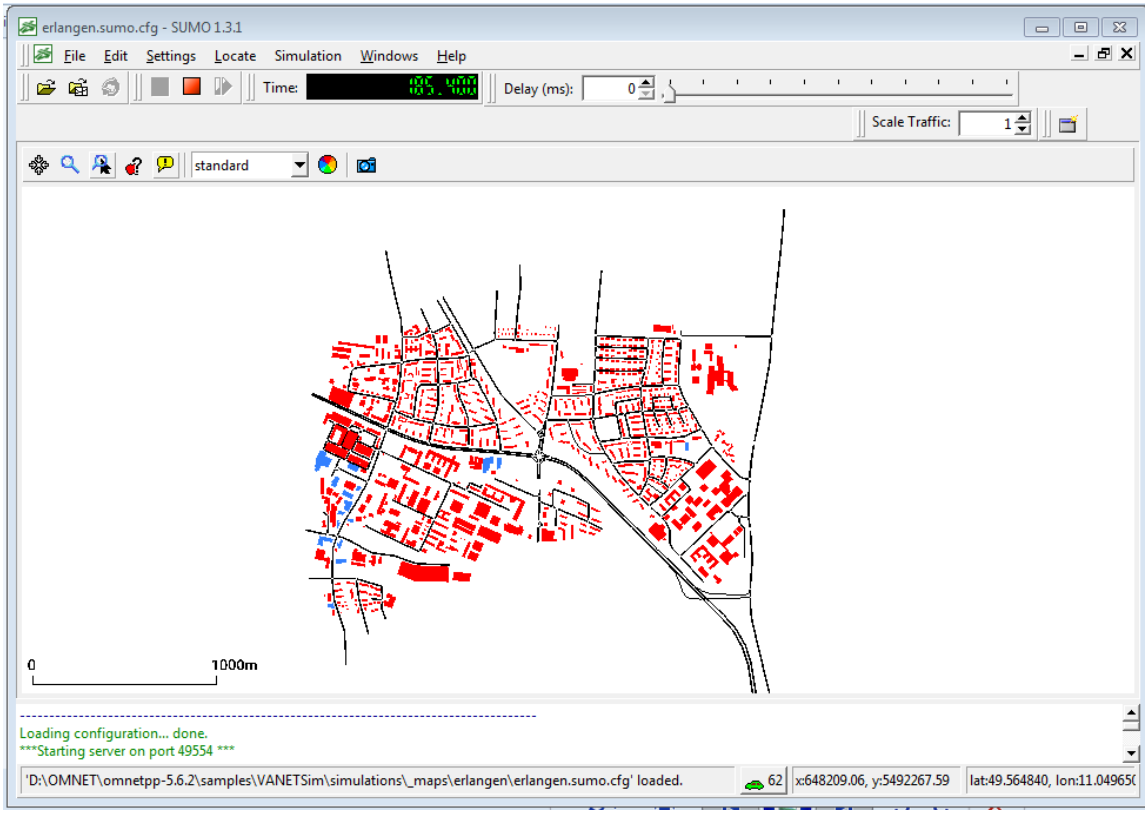

Figure 4.2. Mobility traffic scenario.

By utilizing SUMO's mobility model, we were able to simulate the movement of vehicles in an urban setting, capturing the intricacies and dynamics of real-world traffic scenarios. This enabled us to evaluate the behavior and performance of our proposed SecE-V2X protocol in a realistic mobility environment.

4. Attack modeling

The proposed protocol underwent testing against three different attack types, considering various numbers of malicious nodes (0, 3, 6, and 9). The malicious nodes were placed within the node pools, ensuring no collusion. In Figure 4.3, a real-time simulation scenario with a duration of 200 seconds is depicted, where the malicious nodes are highlighted in red. Evaluating the impact of these malicious nodes on the network, we compared the performance of our proposed protocol with GPSR, as well as two other secure routing approaches (Mustikawati, E. et al. (2017) and Doss, S. et al. (2018)).

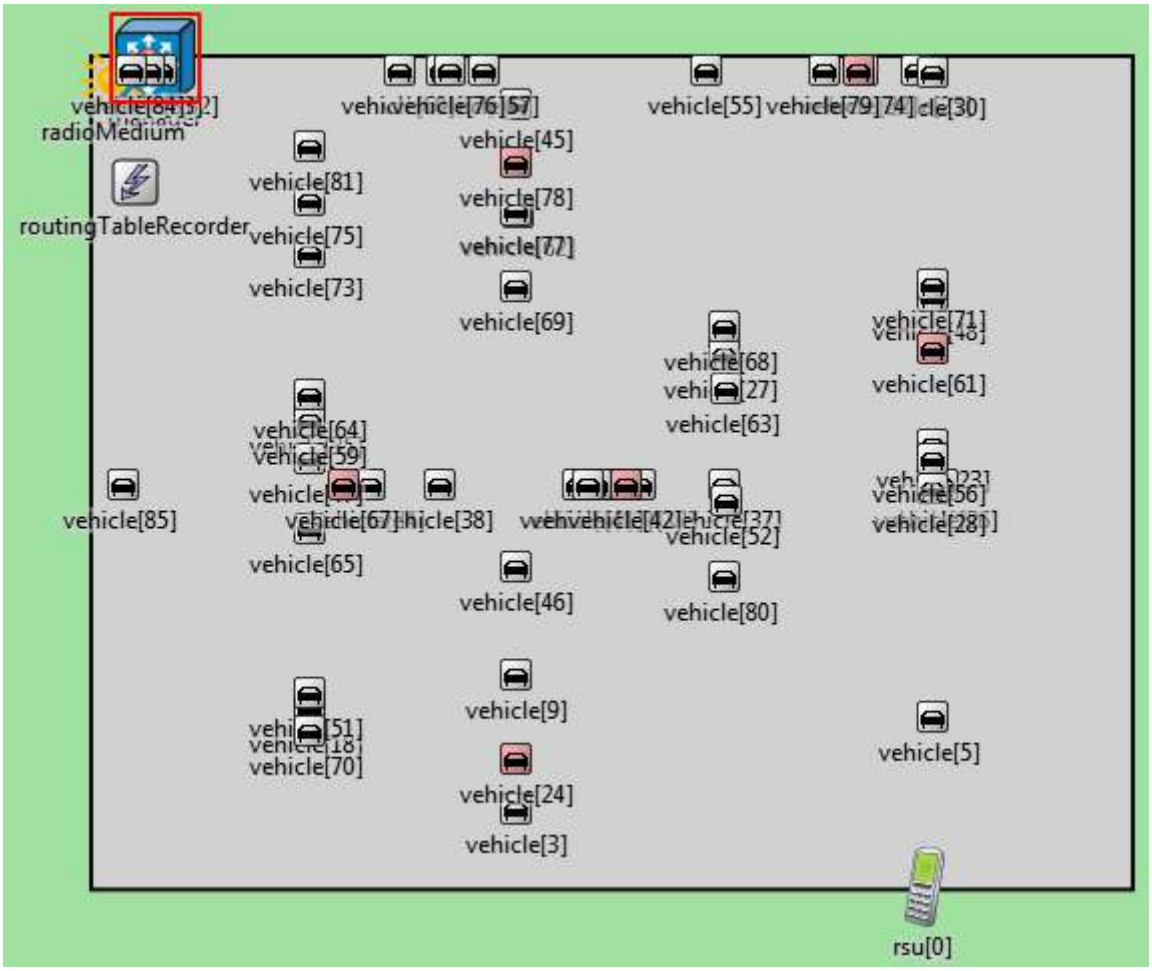

Figure 4.3. Simulation scenario.

4. 1. Blackhole Attack

In our simulation study, we incorporated Blackhole attacks to assess the effectiveness of our recently developed protocol in the presence of such network disruptions. The specific implementation involves dropping all packets passing through the attacker node. By introducing these targeted packet drop scenarios, we aimed to evaluate the protocol's resilience and performance under adverse conditions.

The primary objective of incorporating Blackhole attacks was to evaluate how well our protocol handles packet loss and ensures reliable packet delivery in the face of packet drops. This evaluation provides valuable insights into the protocol's robustness and suitability for real-world deployment, particularly in environments where drop attacks may occur.

To assess the impact of the attacks, we analyzed key performance metrics including Packet Delivery Ratio (PDR), End-to-End Delay (EED), and Throughput. These metrics allow us to quantify the protocol's performance in the presence of Blackhole attacks.

4. 2. Replay attack

To determine their effect on the network's security and evaluate how well our proposed protocol works, we incorporated replay attacks in our simulation experiments. The objective of Replay attacks was to introduce a random delay, ranging from 5 to 10, before transmitting packets.

By introducing Replay attacks with random delays, we aimed to simulate malicious entities attempting to disrupt the timeliness and integrity of the communication within the network. The introduced delays were designed to mimic the unauthorized replaying of previously captured packets, potentially leading to message duplication or outdated information being reintroduced into the network.

During the evaluation, we measured performance metrics such as Packet Delivery Ratio (PDR), End-to-End Delay (EED), and Throughput to quantify the impact of Replay attacks on the protocol's functionality. This allowed us to assess the protocol's ability to detect and mitigate such attacks, ensuring that duplicated or outdated packets were effectively identified and discarded.

4. 3. Jellyfish reordering

In our simulation experiments, we incorporated Jellyfish reordering attacks to evaluate their impact on the network's security and the effectiveness of our proposed protocol. The objective of Jellyfish reordering attacks was to select packets at random before transmitting them to the next-hop.

By introducing Message Jellyfish reordering, we aimed to assess the protocol's ability to detect and mitigate unauthorized modifications to packet order.

During the evaluation, we measured performance metrics such as Packet Delivery Ratio (PDR), End-to-End Delay (EED), and Throughput to quantify the impact of Jellyfish reordering attacks on the protocol's functionality. This allowed us to assess the effectiveness of the protocol in the presence of such attacks.

5. Performance Metrics

To assess the performance of the proposed solution in terms of security and achievement, we employed several key performance metrics. These metrics allowed us to measure and evaluate various aspects of the solution's performance. By considering these performance metrics, we were able to comprehensively evaluate the proposed solution's performance in terms of security and achievement. These metrics provided us with valuable insights into the solution's packet delivery

capability, end-to-end delay, and overall throughput. Analyzing these metrics allowed us to assess the solution's effectiveness and determine its suitability for the intended use case.

5. 1. Packet Delivery Ratio (PDR)

The percentage of successfully delivered packets to the total number of packets sent can be used to compute the packet delivery ratio. PDR provides insights into the solution's ability to reliably transmit packets without loss or errors. A higher PDR indicates a more effective and robust solution in terms of packet delivery. This metric is measured as follows:

$$PDR = \frac{\Sigma\ Packets_{received}}{\Sigma\ Packets_{sent}} * 100 \quad (4.1)$$

where $Packets_{sent}$ is the total number of packets sent by the source node, and $Packets_{received}$ is the total number of packets the destination node has successfully received.

5. 2. End-to-End Delay

The end-to-end delay gauges the typical amount of time it takes a packet to get from its source to its destination node. EED reflects the overall latency or delays experienced by packets in the network. A lower EED signifies faster data transmission and reduced latency, indicating improved efficiency and responsiveness of the solution. It can be determined by dividing the total number of packets by the sum of the times for all packets received, as indicated in Equation (4.2):

$$EED = \frac{\Sigma(Packet_{received\ time} - Packet_{sent\ time})}{Nbr_{Packets}} \quad (4.2)$$

where $Nbr_{Packets}$ is the number of forwarded packets, $Packet_{sent\ time}$ is the time at which the packet was sent, and $Packet_{received\ time}$ is the time at which the packet was received.

5. 3. Throughput

Throughput metric, which quantifies the amount of data successfully transmitted per unit of time. Throughput is a critical measure of the solution's capacity and efficiency in handling data traffic. Higher throughput values indicate a greater volume of data being processed and transmitted within a given timeframe, highlighting the solution's ability to handle data effectively. It can be calculated as follows:

$$Throughput = \frac{Nbr_{Packets}}{T_{Period}} \quad (4.3)$$

where T_{Period} is a specific amount of time and $Nbr_{Packets}$ represents the number of forwarded packets.

6. Proposed protocol evaluation

In this section, we assess the performance of the SecE-V2X protocol in the presence of Blackhole Attacks, Replay Attacks, and Jellyfish reordering. We consider three key performance metrics: Packet Delivery Ratio (PDR), End-to-End Delay (EED), and Throughput. Firstly, we evaluate the three key performance metrics of the proposed protocol under the presence of Blackhole Attacks. Next, we examine our proposed protocol in the presence of Replay Attacks in terms of PDR, EED, and Throughput. Lastly, we analyze the performance in the face of the Jellyfish reordering attack.

6. 1. Under Blackhole Attacks

The effectiveness of our recently developed protocol in the face of Blackhole attacks is the main focus of this section. Table 4.5 provides the performance comparison values.

Table 4.5. Performance values comparison under Blackhole attacks.

Number of Attackers	Metrics	SecE-V2X	GPSR	[99], 2018	[73], 2017
0	PDR (%)	95.34	90.22	93.80	91.60
	EED (ms)	6.80	8.04	7.11	8.64
	Throughput (packets/s)	638.15	523.72	596.33	534.55
3	PDR (%)	93.21	74.34	88.76	79.63
	EED (ms)	7.48	12.51	7.94	12.21
	Throughput (packets/s)	608.22	385.11	556.65	452.11
6	PDR (%)	90.46	57.50	76.72	64.67
	EED (ms)	8.31	19.80	9.64	16.83
	Throughput (packets/s)	589.41	331.86	460.54	346.86
9	PDR (%)	77.43	48.71	69.75	55.29
	EED (ms)	10.48	25.62	13.29	21.34
	Throughput (packets/s)	433.53	273.38	398.41	282.38

6. 1. 1. Packet Delivery Ratio (PDR)

Figure 4.4 contrasts the differences in the PDR of the proposed SecE-V2X with the GPSR and the aforementioned security methods according to the number of attacker nodes.

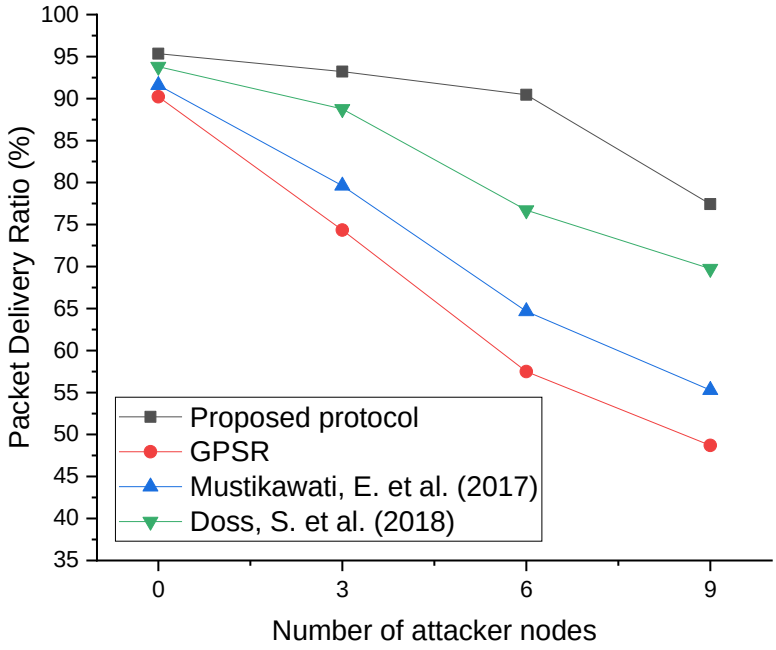

Figure 4.4. Packet delivery ratio under Blackhole attacks.

The results show a clear correlation between the number of attacker nodes and the degradation of PDR for all protocols. As the number of malicious nodes increases, the PDR decreases, indicating the impact of the attacks on the network performance. However, our proposed SecE-V2X protocol demonstrates superior resilience compared to the other schemes.

In a network without any malicious nodes, the SecE-V2X protocol achieves an impressive PDR of approximately 95.34%. This showcases the effectiveness of our approach in maintaining high packet delivery rates when the network is free from attacks. When three malicious nodes are introduced, the proposed protocol still performs strongly with a PDR of 93.21%, outperforming the GPSR (74.34%) and both Doss et al. (88.76%) and Mustikawati et al. (79.63%).

As the number of attacker nodes increases to six, the PDR of the proposed protocol experiences a slight decline to 90.46%. Nevertheless, it remains significantly higher than the PDR values of GPSR (57.50), Doss et al. (76.72%), and Mustikawati et al. (64.67%). Finally, with nine malicious nodes, the PDR of our SecE-V2X protocol drops to 77.43%, emphasizing the impact of

an exaggerated number of attacker nodes on network performance. Despite this reduction, the proposed protocol still outperforms the comparison algorithms in terms of PDR.

These evaluation results highlight the effectiveness of the SecE-V2X protocol in mitigating the impact of Blackhole Attacks on packet delivery. It showcases the protocol's ability to maintain a high level of successful data transmission even in the presence of a considerable number of malicious nodes.

6. 1. 2. End-to-End Delay

Figure 4.5 compares the SecE-V2X algorithm's end-to-end delay performance with that of the GPSR and the security algorithms listed above.

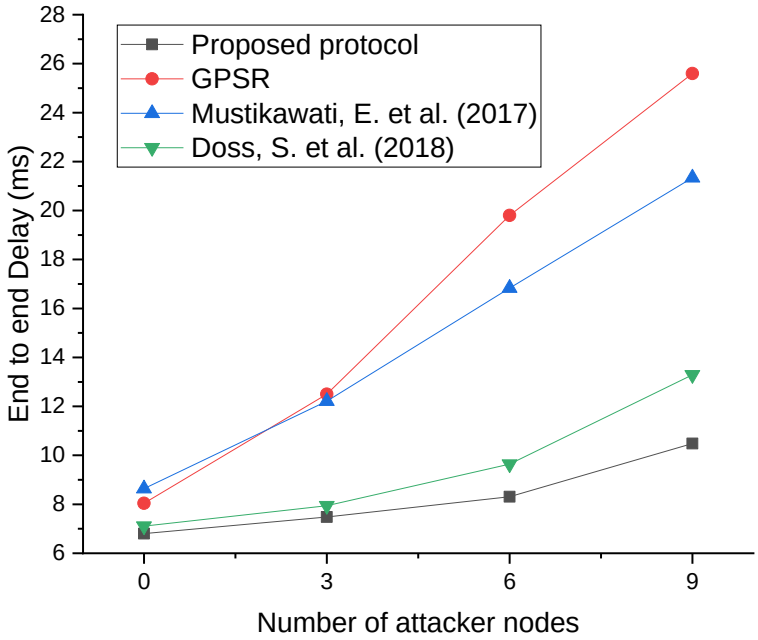

Figure 4.5. End-to-end delay under Blackhole attacks.

The evaluation demonstrates a direct correlation between the number of attacker nodes and the increase in EED for all protocols. As the number of malicious nodes grows, the delay in end-to-end communication rises, indicating the impact of the attacks on network performance. However, the proposed SecE-V2X protocol exhibits favorable EED results compared to the other schemes.

For the network without any attacker nodes (0 attacker nodes), the proposed protocol achieves an EED of 6.8 units, while GPSR exhibits a slightly higher EED of 8.04 units. Doss et al. (2018)

achieves an EED of 7.11 units, and Mustikawati et al. (2017) achieves an EED of 8.64 units. These results indicate that the proposed protocol performs well in terms of minimizing the end-to-end delay compared to the other algorithms under normal network conditions. As the number of attacker nodes increases to three, the EED of the proposed protocol remains relatively low at 7.48 units. GPSR experiences a significant increase in EED to 12.5 units, while Doss et al. (2018) achieves an EED of 7.94 units, and Mustikawati et al. (2017) achieves an EED of 12.21 units. This suggests that the proposed protocol maintains better efficiency in forwarding packets within a reasonable time frame compared to the other algorithms in the presence of a moderate number of attacker nodes. When there are six attacker nodes, the EED of the proposed protocol slightly increases to 8.31 units. In contrast, GPSR experiences a notable increase to 19.8 units, Doss et al. (2018) achieves an EED of 9.64 units, and Mustikawati et al. (2017) achieves an EED of 16.83 units. These findings highlight the ability of the proposed protocol to mitigate the impact of attacker nodes on end-to-end delay, resulting in more efficient packet delivery compared to the other algorithms. With nine attacker nodes, the EED of the proposed protocol further increases to 10.48 units. GPSR exhibits a significantly higher EED of 25.6 units, while Doss et al. (2018) achieves an EED of 13.29 units, and Mustikawati et al. (2017) achieves an EED of 21.34 units. Despite the presence of a larger number of attacker nodes, the proposed protocol maintains a relatively lower end-to-end delay compared to the other algorithms.

These evaluation results highlight the effectiveness of the SecE-V2X protocol in mitigating the increase in End-to-End Delay caused by Blackhole Attacks. The protocol demonstrates its capability to maintain relatively low and stable communication delays, even in the presence of a considerable number of attacker nodes.

6. 1. 3. Throughput

Figure 4.6 displays the generated throughput of the proposed SecE-V2X based on the number of attacker nodes. The objective is to measure the protocol's ability to maintain a high level of data transfer capacity even in the presence of such attacks. We compare the throughput results of our approach with the GPSR and two existing security algorithms, Doss et al. (2018) and Mustikawati et al. (2017).

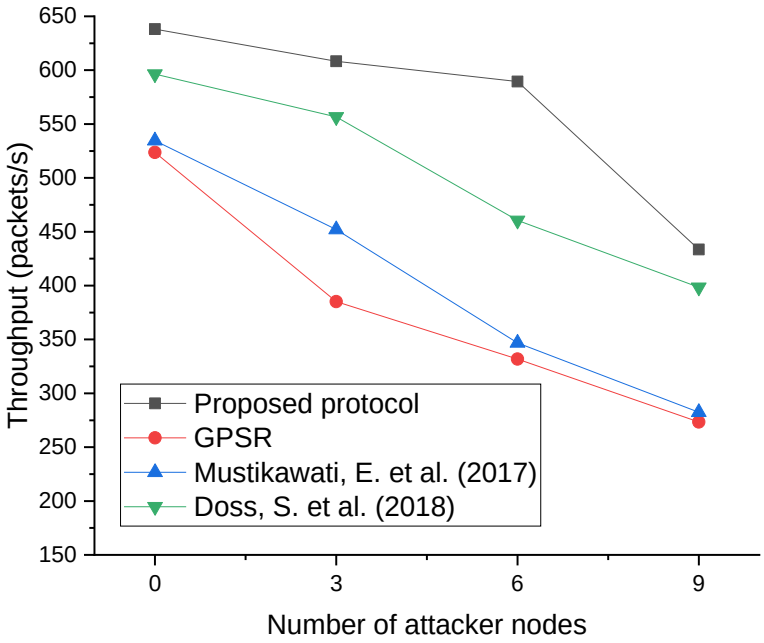

Figure 4.6. Nodes' data throughput under Blackhole attacks.

The evaluation reveals a significant relationship between the number of attacker nodes and the degradation of throughput for all protocols. As the number of malicious nodes increases, the overall data transfer capacity decreases due to the impact of the attacks. However, the proposed SecE-V2X protocol showcases superior throughput performance compared to the other algorithms.

In a network without any attacker nodes, the SecE-V2X protocol achieves a commendable throughput of 638.15 units, demonstrating its ability to maintain efficient data transfer capacity under normal operating conditions. When three malicious nodes are introduced, the proposed protocol experiences a slight decline in throughput to 608.22 units, outperforming GPSR which achieves a throughput of 385.11 units and both Doss et al. (556.65 units) and Mustikawati et al. (452.11 units).

As the number of attacker nodes increases to six, the throughput of the proposed protocol further decreases to 589.41 units. Despite this decline, it remains significantly more efficient than the throughput values of the GPSR (331.86 units), Doss et al. (460.54 units), and Mustikawati et al. (346.86 units). Finally, with nine malicious nodes, the throughput of our SecE-V2X protocol drops to 433.53 units, indicating the impact of a larger number of attacker nodes on data transfer capacity. Nevertheless, even in this scenario, the proposed protocol outperforms the comparison algorithms in terms of throughput.

These evaluation results highlight the effectiveness of the SecE-V2X protocol in maintaining a high level of throughput even in the presence of Blackhole Attacks.

6. 2. Under Replay attack

To examine its resilience and effectiveness in mitigating the impact of such attacks, this section primarily focuses on the performance of our proposed protocol in the face of Replay attacks. The performance comparison values are shown in Table 4.6.

Table 4.6. Performance values comparison under Replay attacks.

Number of Attackers	Metrics	SecE-V2X	GPSR	[99], 2018	[73], 2017
0	PDR (%)	95.34	90.22	93.80	91.60
	EED (ms)	6.80	8.04	7.11	8.64
	Throughput (packets/s)	638.15	523.72	596.33	534.55
3	PDR (%)	93.12	88.47	91.38	89.20
	EED (ms)	6.94	68.72	22.34	25.84
	Throughput (packets/s)	617.42	483.31	543.61	494.76
6	PDR (%)	90.03	84.16	88.78	86.13
	EED (ms)	6.63	74.37	26.14	28.91
	Throughput (packets/s)	608.23	457.74	506.23	469.41
9	PDR (%)	85.53	81.33	84.94	83.32
	EED (ms)	6.77	78.11	31.05	33.18
	Throughput (packets/s)	562.27	421.12	491.14	421.38

6. 2. 1. Packet Delivery Ratio (PDR)

According to the quantity of attacker nodes, Figure 4.7 compares the PDR of the proposed SecE-V2X with the GPSR and previously mentioned security approaches.

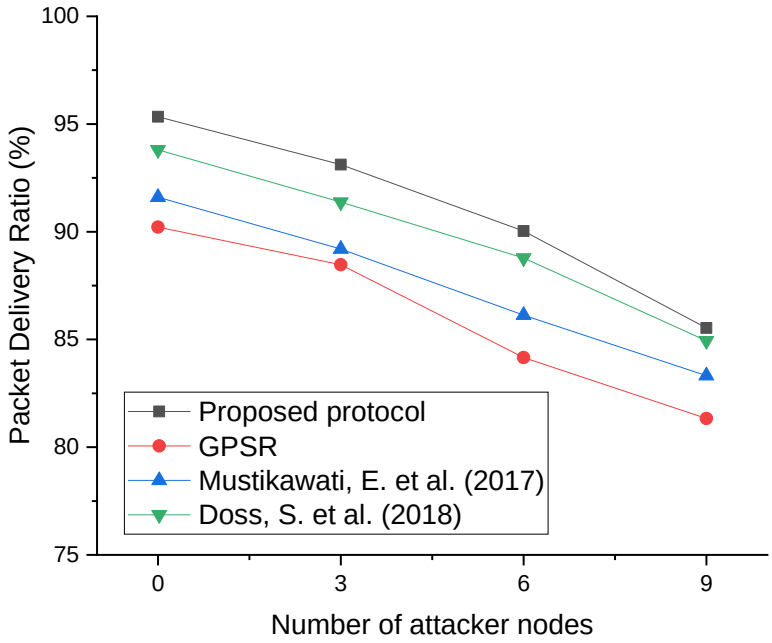

Figure 4.7. Packet delivery ratio under Replay attacks.

When there are no attacker nodes in the network (0 attacker nodes), the proposed protocol demonstrates a superior PDR of 95.34%, outperforming GPSR with a PDR of 90.22%. Doss et al. (2018) achieves a PDR of 93.8%, while Mustikawati et al. (2017) achieves a PDR of 91.6%. These results highlight the effectiveness of the proposed protocol in maintaining a high packet delivery rate, indicating its robustness against the Replay attack.

As the number of attacker nodes increases to three, the proposed protocol maintains a consistently high PDR of 93.12%. In comparison, GPSR achieves a PDR of 88.47%, Doss et al. (2018) achieves a PDR of 91.38%, and Mustikawati et al. (2017) achieves a PDR of 89.2%. These findings underscore the resilience of the proposed protocol in mitigating the impact of the Replay attack, ensuring reliable packet delivery even in the presence of malicious nodes.

Furthermore, as the number of attacker nodes further increases to six and nine, the proposed protocol continues to exhibit superior performance with PDR values of 90.03% and 85.53% respectively. In contrast, GPSR experiences a decline in PDR to 84.16% and 81.33% under the same conditions. Similarly, Doss et al. (2018) and Mustikawati et al. (2017) demonstrate lower PDR values of 88.78% and 86.13%, and 84.94% and 83.32% respectively. These results reaffirm the effectiveness of the proposed protocol in maintaining reliable communication and forwarding packets accurately, even when faced with an increasing number of attacker nodes in the network.

6. 2. 2. End-to-End Delay

One of the key features of our proposed protocol is the inclusion of timestamps associated with each transferred message. This allows the protocol to track the age of the packets and discard outdated packets, ensuring that only the most relevant and up-to-date information is forwarded. By discarding outdated packets, our protocol reduces the potential delay caused by processing and transmitting unnecessary or obsolete data. Figure 4.8 illustrates the end-to-end delay ratio for various numbers of Replay attacker nodes.

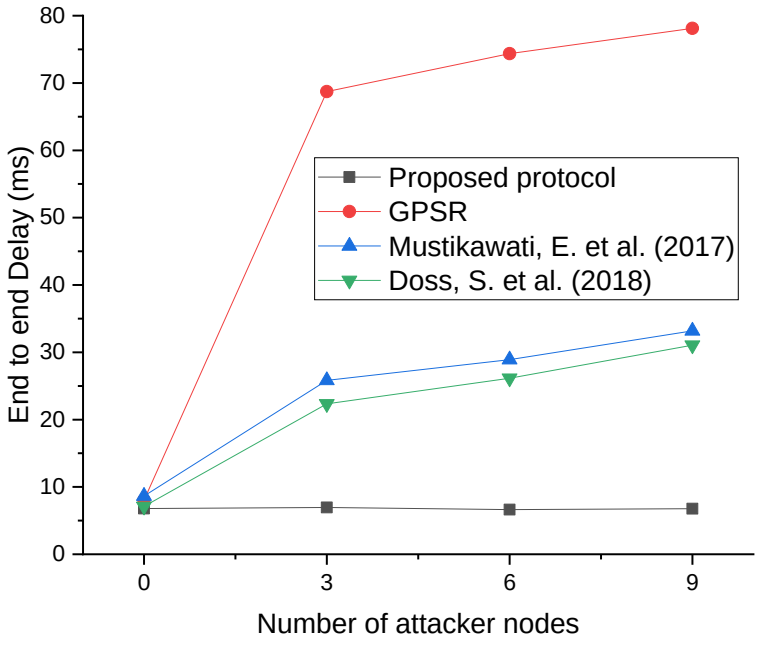

Figure 4.8. End-to-end delay under Replay attacks.

The evaluation results highlight the effectiveness of our proposed protocol in managing EED under Replay attack scenarios. When the number of attacker nodes is zero, our protocol achieves an impressive EED of 6.8 milliseconds, outperforming the comparison protocols such as GPSR (8.04 milliseconds), Doss, S. et al. (2018) (7.11 milliseconds), and Mustikawati, E. et al. (2017) (8.64 milliseconds).

As the number of attacker nodes increases, our protocol maintains a consistent and low EED compared to the other protocols. Even with three attacker nodes, our protocol achieves an EED

of 6.94 milliseconds, while the other protocols experience significantly higher delays, ranging from 22.34 milliseconds to 25.84 milliseconds.

With six and nine attacker nodes, our proposed protocol continues to demonstrate its effectiveness by maintaining low EED values of 6.63 milliseconds and 6.77 milliseconds, respectively. In contrast, the comparison protocols experience much higher delays, ranging from 74.37 milliseconds to 78.11 milliseconds, and from 26.14 milliseconds to 33.18 milliseconds, respectively.

These results indicate that our proposed protocol successfully mitigates the impact of Replay attacks on EED, ensuring efficient and timely message delivery even in the presence of malicious nodes.

6. 2. 3. Throughput

A comparison of throughput based on different Replay attacker node counts is shown in Figure 4.9.

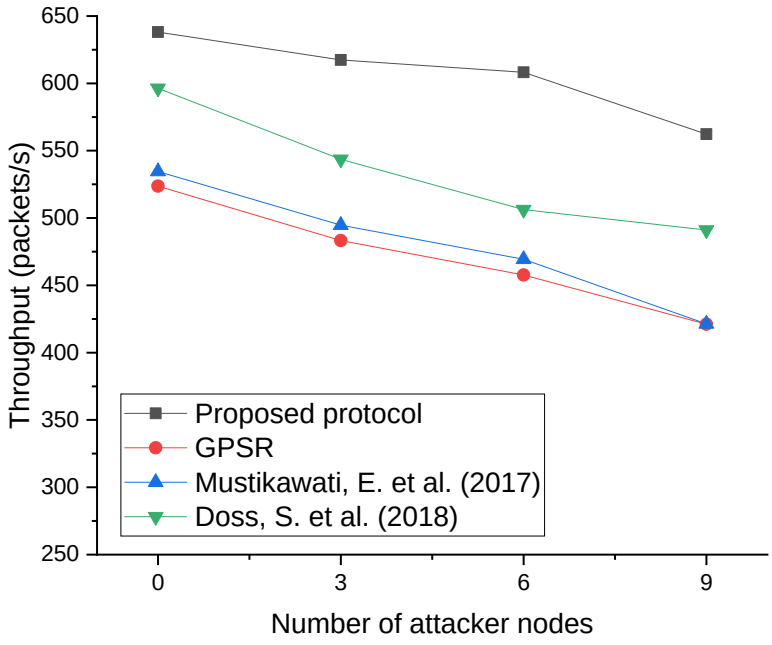

Figure 4.9. Nodes' data throughput under Replay attacks.

The evaluation results demonstrate the effectiveness of our proposed protocol in maintaining a good throughput even under Replay attack scenarios. When there are no attacker nodes, our protocol achieves a high throughput of 638.15 units, surpassing the comparison protocols such as

GPSR (523.72 units), Doss, S. et al. (2018) (596.33 units), and Mustikawati, E. et al. (2017) (534.55 units).

As the number of attacker nodes increases, our proposed protocol continues to exhibit a competitive throughput compared to the other protocols. Even with three attacker nodes, our protocol achieves a throughput of 617.42 units, while the comparison protocols experience slightly lower throughputs ranging from 483.31 units to 543.61 units.

With six and nine attacker nodes, our proposed protocol maintains a respectable throughput of 608.23 units and 562.27 units, respectively. In contrast, the comparison protocols exhibit lower throughputs, ranging from 457.74 units to 506.23 units and from 421.12 units to 491.14 units, respectively.

6. 3. Under Jellyfish reordering

This section mainly concentrates on the performance of our proposed protocol in the face of Jellyfish reordering attacks to analyze its resilience and effectiveness in reducing the effects of such attacks. Table 4.7 displays the performance comparison values.

Table 4.7. Performance values comparison under Jellyfish reordering attacks.

Number of Attackers	Metrics	SecE-V2X	GPSR	[99], 2018	[73], 2017
0	PDR (%)	95.34	90.22	93.80	91.60
	EED (ms)	6.80	8.04	7.11	8.64
	Throughput (packets/s)	638.15	523.72	596.33	534.55
3	PDR (%)	93.92	74.38	89.57	79.92
	EED (ms)	7.70	12.05	8.64	13.03
	Throughput (packets/s)	612.82	385.11	559.42	453.76
6	PDR (%)	90.50	61.42	79.19	72.07
	EED (ms)	8.71	19.82	11.49	17.21
	Throughput (packets/s)	556.02	331.86	472.08	402.10
9	PDR (%)	78.18	48.73	74.84	56.86
	EED (ms)	10.88	25.06	14.19	22.24
	Throughput (packets/s)	441.15	273.38	415.64	290.37

6. 3. 1. Packet Delivery Ratio (PDR)

The results presented in Figure 4.10 demonstrate the performance of different protocols, including the proposed protocol (SecE-V2X), GPSR, Doss et al. (2018), and Mustikawati et al. (2017), in terms of Packet Delivery Ratio (PDR) under varying numbers of attacker nodes.

As the number of attacker nodes increases, it is observed that the PDR decreases across all protocols, indicating the impact of malicious nodes on the delivery of packets in the network.

Comparing the results, the proposed protocol (SecE-V2X) exhibits a higher PDR compared to GPSR, Doss et al. (2018), and Mustikawati et al. (2017). For example, with three malicious nodes, the PDR of the proposed protocol is 93.915%, while GPSR achieves a PDR of 74.38%, Doss et al. (2018) achieves 89.57%, and Mustikawati et al. (2017) achieves 79.915%. When the number of attacker nodes increases to six, the PDR of the proposed protocol decreases to 90.5%, while GPSR drops to 61.42%, Doss et al. (2018) decreases to 79.185%, and Mustikawati et al. (2017) declines to 72.07%. Furthermore, with nine attacker nodes, the PDR of the proposed protocol remains the highest among the compared protocols, reaching 78.18%. In contrast, GPSR drops to 48.73%, Doss et al. (2018) achieves 74.84%, and Mustikawati et al. (2017) experiences the lowest PDR of 56.855%.

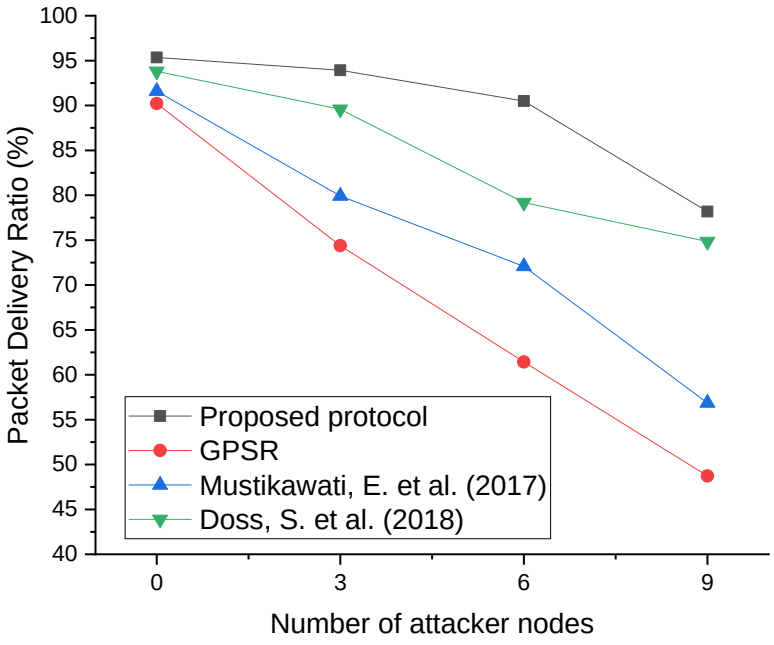

Figure 4.10. Packet delivery ratio under Jellyfish reordering attacks.

6. 3. 2. End-to-End Delay

Figure 4.11 provides insights into the results obtained from evaluating the performance of different protocols, including the proposed protocol (SecE-V2X), GPSR, Doss et al. (2018), and Mustikawati et al. (2017), in terms of end-to-end delay under varying numbers of attacker nodes.

For three malicious nodes, the proposed protocol demonstrates the shortest end-to-end delay of 7.7 ms, outperforming the results of Doss et al. (2018) with 8.64 ms and Mustikawati et al. (2017) with 13.02 ms. When the number of attacker nodes increases to nine, the end-to-end delay of the proposed protocol remains the best among the compared protocols, measuring 10.88 ms. In contrast, Doss et al. (2018) experiences an end-to-end delay of 14.19 ms, and Mustikawati et al. (2017) encounters a considerably higher delay of 22.24 ms.

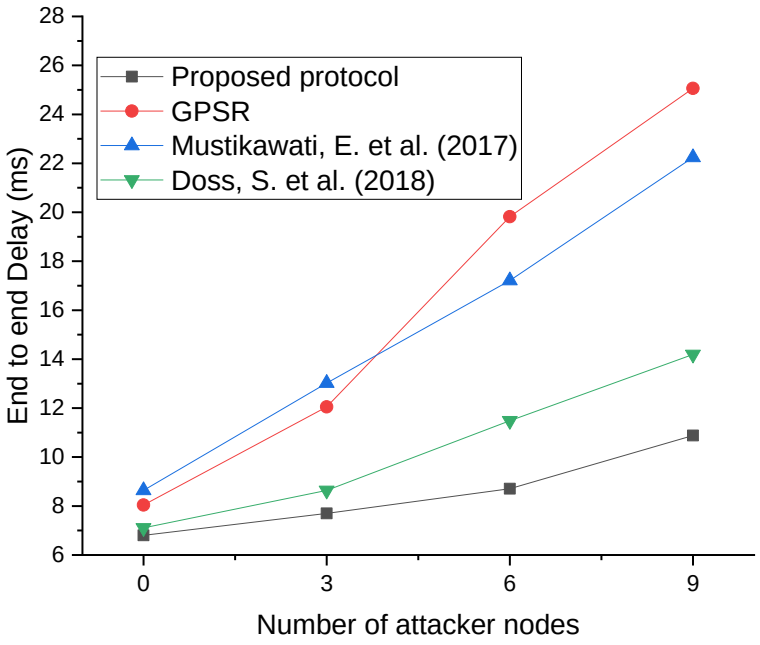

Figure 4.11. End-to-end delay under Jellyfish reordering attacks.

6. 3. 3. Throughput

Figure 4.12 displays the results of the performance evaluation of different protocols, in terms of Nodes' data throughput for varying numbers of attacker nodes.

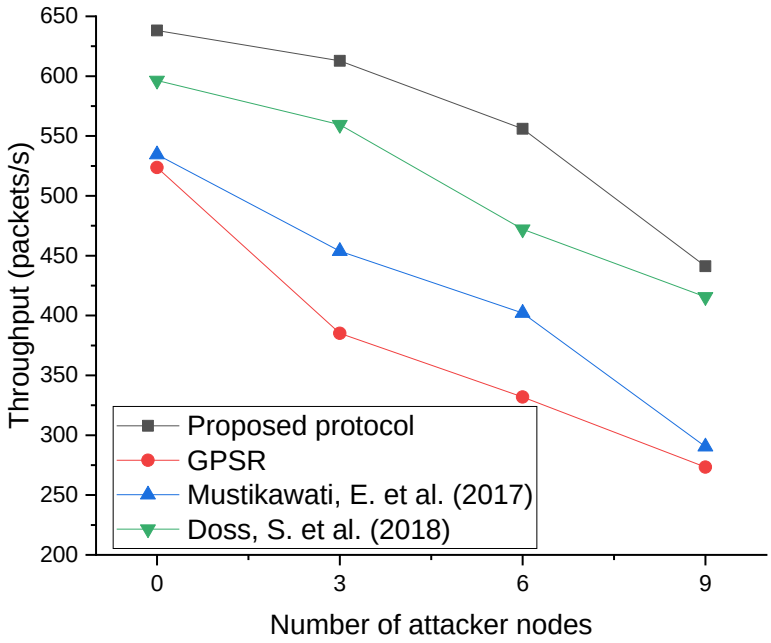

Figure 4.12. Nodes' data throughput under Jellyfish reordering attacks.

When there are no attacker nodes (0), the proposed protocol achieves a throughput of 638.15 packets/s, while GPSR achieves 523.72 packets/s, Doss et al. (2018) achieves 596.33 packets/s, and Mustikawati et al. (2017) achieves 534.55 packets/s. As the number of attacker nodes increases to three, the throughput decreases for all protocols. The proposed protocol achieves a throughput of 612.82 packets/s, GPSR achieves 385.11 packets/s, Doss et al. (2018) achieves 559.42 packets/s, and Mustikawati et al. (2017) achieves 453.76 packets/s. With six attacker nodes, the throughput continues to decrease for all protocols. The proposed protocol achieves a throughput of 556.02 packets/s, GPSR achieves 331.86 packets/s, Doss et al. (2018) achieves 472.08 packets/s, and Mustikawati et al. (2017) achieves 402.1 packets/s. When the number of attacker nodes increases to nine, the throughput further decreases. The proposed protocol achieves a throughput of 441.145 packets/s, GPSR achieves 273.38 packets/s, Doss et al. (2018) achieves 415.635 packets/s, and Mustikawati et al. (2017) achieves 290.37 packets/s.

The results indicate that as the number of attacker nodes increases, the throughput of all protocols decreases. However, the proposed protocol (SecE-V2X) consistently outperforms the other protocols in terms of throughput across all scenarios. It demonstrates the effectiveness of the proposed protocol in maintaining a higher data throughput, even under the presence of attacker

nodes. This suggests that the proposed protocol provides better communication efficiency and can support higher data transfer rates in VANET environments compared to the other evaluated protocols.

7. Conclusion

In this chapter, we have provided a comprehensive evaluation of the network performances by explaining the methodology employed. We have presented the tools used for the simulations, namely OMNeT++, VEINS, and SUMO, along with the parameters configured for the simulation experiments. The simulation environment was meticulously built to closely resemble real-world conditions in VANETs. This included capturing the network traffic pattern, mobility pattern, and incorporating a fading propagation model. By replicating these factors, we aimed to create a realistic simulation environment that reflects the complexities and challenges of VANETs.

To assess the performance of our proposed protocol, we selected three routing behaviors that are associated with common problems encountered in VANET routing environments: Blackhole, Replay, and Message tampering attacks. These attack scenarios allowed us to evaluate the protocol's robustness and effectiveness in the face of security threats.

By allowing each node to select the most trusted neighbor towards the destination, our protocol outperforms alternative solutions. Although the results may not guarantee 100% packet delivery, they are deemed acceptable given the inherent unstable connectivity in VANETs. Additionally, any dropped packets can be resolved by employing higher-layer protocols for re-sending.

The outcomes of our performance evaluation provide valuable insights into the strengths and capabilities of our proposed protocol. The results validate its ability to enhance trust and reliability in communication among vehicles in VANETs. Furthermore, they highlight the significance of our protocol in mitigating common routing issues and addressing security challenges.

.

Chapter

5

Conclusion and Future Works

Content

1. General Conclusion

In this final chapter, we provide a comprehensive conclusion to the research contributions made throughout our thesis. We begin by presenting an overview of our work and summarizing the key contributions we have made to the field. Subsequently, we discuss the challenges and technical obstacles that we encountered during our extensive evaluation of the proposed solutions. Finally, we outline potential directions for future work, highlighting areas that warrant further investigation and development.

1.1. Realized study review

The focus of this thesis revolves around the security aspects of VANET, specifically addressing the mitigation of attacks. Our primary objective is to establish secure and trustworthy communication within VANET. To accomplish this, we have devised a novel protocol that builds upon the GPSR routing protocol and incorporates elements from blockchain cryptography and reputation mechanisms. By adapting and integrating various algorithms and mechanisms, we tailor them to meet the specific requirements and challenges of VANET, resulting in a secure and efficient protocol.

In this thesis, we have undertaken a comprehensive study of the security aspects of VANET networks, focusing specifically on attack mitigation. We began by providing an overview of VANET networks, discussing their main characteristics, and contextualizing our research within this field.

The second chapter delved into the topic of security in VANETs. We explored the fundamental concepts of security services and classified various types of attacks that commonly occur within this network category. Additionally, we surveyed and classified existing proposed solutions aimed at mitigating these attacks.

In the third chapter, we introduced our novel routing protocol, SecE-V2X, designed to enhance security in VANETs. We provided a detailed description of the protocol, highlighting its key features and advantages. Furthermore, we conducted a thorough security analysis of the proposed protocol, assessing its resilience against potential attacks.

Moving to the fourth chapter, we implemented and tested the proposed protocol, conducting extensive simulations and experiments. We analyzed the collected data and discussed the outcomes of our experiments, evaluating the performance of our secured routing protocol across various VANET scenarios. Additionally, we compared our results with those of other existing secured

routing protocols, aiming to assess the effectiveness of our protocol. Throughout this chapter, we engaged in a comprehensive discussion of our findings.

1.2. Thesis Contribution

The contributions of this thesis are multifold. Firstly, we reviewed and analyzed existing vulnerabilities within VANETs, focusing on various attack types. Secondly, we evaluated the effectiveness of existing security solutions in vehicular networks, identifying their limitations. Thirdly, we proposed a new secured routing protocol, offering a more secure and reliable communication infrastructure for VANETs. Moreover, we implemented and tested our protocol using simulation tools, comparing its performance against other existing secured protocols.

1.3. Limitations and extensions

Throughout our research, we encountered certain limitations and obstacles that are important to acknowledge. These limitations provide opportunities for further exploration and improvement in the field of VANET security. Firstly, the integration of diverse domains proved challenging, as we combined concepts from wireless networks, cryptography, reputation systems, and blockchain. Integrating these areas of knowledge required a deep understanding and effective combination of methodologies.

Scalability is another concern that surfaced during our experiments and simulations. While we evaluated the performance of our proposed protocol in controlled scenarios, the scalability of the protocol in real-world VANET deployments remains a significant consideration. As the number of vehicles and nodes increases, the protocol's scalability may be affected.

Practical implementation challenges are also noteworthy. Although our protocol showed promising results in simulations, its practical implementation in real-world VANETs may face obstacles. Factors such as hardware limitations, varying network conditions, and resource constraints can impact the protocol's performance. Moreover, VANETs involve human behavior and interactions, making it challenging to predict and mitigate attacks effectively. Human factors, including driver behavior, trust dynamics, and compliance with security measures, have a significant impact on network security.

2. Future works

Looking ahead, there are still several avenues for further exploration in the field of VANET security. Future research and development efforts can focus on refining and optimizing our protocol, with a specific emphasis on the following areas:

- **Real-World Evaluations:** Conducting real-world evaluations to validate the effectiveness and practicality of our proposed protocol is crucial. By deploying the protocol in actual vehicular networks, we can assess its performance under real-life conditions and gather valuable insights into its practical implementation challenges, scalability, and adaptability to diverse environments.

- **Expanded Performance Metrics:** In addition to the performance metrics already considered, such as packet delivery ratio and end-to-end delay, further evaluation of our protocol can include other important metrics. These may include resource consumption, such as CPU and memory usage, as well as metrics such as Attack Detection Rate, False Positive Rate, and False Negative Rate. By examining these metrics, we can gain a more comprehensive understanding of the protocol's overall performance.

- **Cooperative Attacks:** Investigating the case of cooperative attacks, where multiple nodes collaborate to disrupt routing services, presents an interesting direction for future research. By studying the techniques employed by these malicious nodes and defining a model to understand their behaviors and impact on routing services, we can enhance our protocol's resilience against such sophisticated attacks. This research can provide valuable insights into the design of more robust security mechanisms and countermeasures to effectively combat cooperative attacks in VANETs.

Bibliography

1. Taneja, K.; Patel, R.B. Mobile Ad Hoc Networks: Challenges and Future. In Proceedings of the Proceedings of National Conference on Challenges & Opportunities in Information Technology; Citeseer, 2007; pp. 133–135.
2. Rikitianskaia, M. "The Real Ethernet": The Transnational History of Global Wi-Fi Connectivity. *New Media & Society* 2022, 14614448221103533, doi:10.1177/14614448221103533.
3. Frodigh, M.; Johansson, P.; Larsson, P. Wireless Ad Hoc Networking: The Art of Networking without a Network. *Ericsson review* 2000, *4*, 249.
4. Zeadally, S.; Hunt, R.; Chen, Y.-S.; Irwin, A.; Hassan, A. Vehicular Ad Hoc Networks (VANETS): Status, Results, and Challenges. *Telecommun Syst* 2012, *50*, 217–241, doi:10.1007/s11235-010-9400-5.
5. Wang, J.; Shao, Y.; Ge, Y.; Yu, R. A Survey of Vehicle to Everything (V2X) Testing. *Sensors* 2019, *19*, 334, doi:10.3390/s19020334.
6. Osman, R.A.; Abdelsalam, A.K. A Novel Adaptive Approach for Autonomous Vehicle Based on Optimization Technique for Enhancing the Communication between Autonomous Vehicle-to-Everything through Cooperative Communication. *Applied Sciences* 2021, *11*, 9089, doi:10.3390/app11199089.
7. Nabwene, R.N. Review on Intelligent Internal Attacks Detection in VANET. In Proceedings of the 2018 4th Annual International Conference on Network and Information Systems for Computers (ICNISC); IEEE, 2018; pp. 1–6.
8. Chatterjee, T.; Karmakar, R.; Kaddoum, G.; Chattopadhyay, S.; Chakraborty, S. A Survey of VANET/V2X Routing from the Perspective of Non-Learning-and Learning-Based Approaches. *IEEE Access* 2022, *10*, 23022–23050.
9. Malik, N.; Puthal, D.; Nanda, P. An Overview of Security Challenges in Vehicular Ad-Hoc Networks. In Proceedings of the 2017 International Conference on Information Technology (ICIT); December 2017; pp. 208–213.
10. Abassi, R. VANET Security and Forensics: Challenges and Opportunities. *WIREs Forensic Science* 2019, *1*, e1324, doi:10.1002/wfs2.1324.
11. Sharma, S.; Kaul, A.; Ahmed, S.; Sharma, S. A Detailed Tutorial Survey on VANETs: Emerging Architectures, Applications, Security Issues, and Solutions. *International Journal of Communication Systems* 2021, *34*, e4905, doi:10.1002/dac.4905.
12. Baharlouei, H.; Makanju, A.; Zincir-Heywood, N. Exploring Realistic VANET Simulations for Anomaly Detection of DDoS Attacks. In Proceedings of the 2022 IEEE 95th Vehicular Technology Conference: (VTC2022-Spring); June 2022; pp. 1–7.
13. Khekare, G.S.; Sakhare, A.V. A Smart City Framework for Intelligent Traffic System Using VANET. In Proceedings of the 2013 International Mutli-Conference on Automation, Computing, Communication, Control and Compressed Sensing (iMac4s); March 2013; pp. 302–305.
14. Azzaoui, N. Towards a Smart City: Data Dissemination in IoV Networks, Ouargla university.
15. Benguenane, M.; Korichi, A.; Brik, B.; Azzaoui, N. Towards Mitigating Jellyfish Attacks Based on Honesty Metrics in V2X Autonomous Networks. *Applied Sciences* 2023, *13*, 4591, doi:10.3390/app13074591.
16. Srivastava, A.; Verma, S.; Kavita; Jhanjhi, N.Z.; Talib, M.N.; Malhotra, A. Analysis of Quality of Service in VANET. *IOP Conf. Ser.: Mater. Sci. Eng.* 2020, *993*, 012061, doi:10.1088/1757-899X/993/1/012061.
17. Anwer, M.S.; Guy, C. A Survey of VANET Technologies. 2014.
18. Rasheed, A.; Gillani, S.; Ajmal, S.; Qayyum, A. Vehicular Ad Hoc Network (VANET): A Survey, Challenges, and Applications. In Proceedings of the Vehicular Ad-Hoc Networks for Smart Cities; Laouiti, A., Qayyum, A., Mohamad Saad, M.N., Eds.; Springer: Singapore, 2017; pp. 39–51.

19. Kiela, K.; Barzdenas, V.; Jurgo, M.; Macaitis, V.; Rafanavicius, J.; Vasjanov, A.; Kladovscikov, L.; Navickas, R. Review of V2X–IoT Standards and Frameworks for ITS Applications. *Applied Sciences* 2020, *10*, 4314, doi:10.3390/app10124314.
20. Yang, Y.; Bagrodia, R. Evaluation of VANET-Based Advanced Intelligent Transportation Systems. In Proceedings of the Proceedings of the sixth ACM international workshop on VehiculAr InterNETworking; Association for Computing Machinery: New York, NY, USA, September 25 2009; pp. 3–12.
21. Paul, B.; Ibrahim, M.; Bikas, M.A.N. VANET Routing Protocols: Pros and Cons 2012.
22. Brik, B. La Collecte et l'agrégation de Données Dans Les Réseaux Mobiles., Université Amar Telidji de Laghouat , Département d'Informat, 2017.
23. Diab, T. Security Management in Vehicular Ad Hoc Networks. phdthesis, Université de Haute Alsace - Mulhouse, 2020.
24. Consortium, D.I. *DSRC Technology and the DSRC Industry Consortium (DIC) Prototype Team*; White Paper. Tech. rep, 2005;
25. Association, I.S. Wireless Lan Medium Access Control (Mac) and Physical Layer (Phy) Specifications Amendment 6: Wireless Access in Vehicular Environments. *Standard IEEE* 2010, *802*.
26. Kim, J.-W.; Kim, J.-W.; Jeon, D.-K. A Cooperative Communication Protocol for QoS Provisioning in IEEE 802.11p/Wave Vehicular Networks. *Sensors* 2018, *18*, 3622, doi:10.3390/s18113622.
27. Rana, S.; Rana, S.; Purohit, K.C. A Review of Various Routing Protocols in VANET. *International Journal of Computer Applications 96*.
28. Qureshi, K.N.; Abdullah, A.H. TOPOLOGY BASED ROUTING PROTOCOLS FOR VANET AND THEIR COMPARISON WITH MANET. . *Vol.* 2005, *58*.
29. Johnson, D.B. The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks. *draft-ietf-manet-dsr-09. txt* 2003.
30. Belding-Royer, E.; Chakeres, I.; Johnson, D.; Perkins, C. DYMO–Dynamic MANET on-Demand Routing Protocol. *Proceedings of the Sixty-First Internet Engineering Task Force, Washington, DC, USA* 2004.
31. Royer, E.M.; Perkins, C.E. Ad-Hoc on-Demand Distance Vector Routing. In Proceedings of the Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications; 1999; Vol. 2, pp. 90–100.
32. Clausen, T.; Jacquet, P. *Optimized Link State Routing Protocol (OLSR)*; 2003;
33. Perkins, C.E.; Bhagwat, P. Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers. *SIGCOMM Comput. Commun. Rev.* 1994, *24*, 234–244, doi:10.1145/190809.190336.
34. Beijar, N. Zone Routing Protocol (ZRP). *Networking Laboratory, Helsinki University of Technology, Finland* 2002, *9*, 12.
35. Srivastava, A.; Prakash, A.; Tripathi, R. Location Based Routing Protocols in VANET: Issues and Existing Solutions. *Vehicular Communications* 2020, *23*, 100231.
36. Karp, B.; Kung, H.-T. GPSR: Greedy Perimeter Stateless Routing for Wireless Networks. In Proceedings of the Proceedings of the 6th annual international conference on Mobile computing and networking; 2000; pp. 243–254.
37. Lochert, C.; Mauve, M.; Füßler, H.; Hartenstein, H. Geographic Routing in City Scenarios. *ACM SIGMOBILE mobile computing and communications review* 2005, *9*, 69–72.
38. Naumov, V.; Gross, T.R. Connectivity-Aware Routing (CAR) in Vehicular Ad-Hoc Networks. In Proceedings of the IEEE INFOCOM 2007-26th IEEE International Conference on Computer Communications; IEEE, 2007; pp. 1919–1927.
39. Devangavi, A.D.; Gupta, R. Routing Protocols in VANET — A Survey. In Proceedings of the 2017 International Conference On Smart Technologies For Smart Nation (SmartTechCon); August 2017; pp. 163–167.

40. Blum, J.; Eskandarian, A.; Hoffman, L. Mobility Management in IVC Networks. In Proceedings of the IEEE IV2003 Intelligent Vehicles Symposium. Proceedings (Cat. No. 03TH8683); IEEE, 2003; pp. 150–155.
41. Fotohi, R.; Ebazadeh, Y.; Geshlag, M.S. A New Approach for Improvement Security against DoS Attacks in Vehicular Ad-Hoc Network. *ijacsa* 2016, *7*, doi:10.14569/IJACSA.2016.070702.
42. Lyamin, N.; Kleyko, D.; Delooz, Q.; Vinel, A. AI-Based Malicious Network Traffic Detection in VANETs. *IEEE Network* 2018, *32*, 15–21, doi:10.1109/MNET.2018.1800074.
43. Bangui, H.; Ge, M.; Buhnova, B. A Hybrid Machine Learning Model for Intrusion Detection in VANET. *Computing* 2022, *104*, 503–531, doi:10.1007/s00607-021-01001-0.
44. Blum, J.; Eskandarian, A. The Threat of Intelligent Collisions. *IT Professional* 2004, *6*, 24–29, doi:10.1109/MITP.2004.1265539.
45. Verma, A.; Saha, R.; Kumar, G.; Kim, T. The Security Perspectives of Vehicular Networks: A Taxonomical Analysis of Attacks and Solutions. *Applied Sciences* 2021, *11*, 4682, doi:10.3390/app11104682.
46. Mahmood, J.; Duan, Z.; Yang, Y.; Wang, Q.; Nebhen, J.; Bhutta, M.N.M. Security in Vehicular Ad Hoc Networks: Challenges and Countermeasures. *Security and Communication Networks* 2021, *2021*, 1–20.
47. Mokhtar, B.; Azab, M. Survey on Security Issues in Vehicular Ad Hoc Networks. *Alexandria Engineering Journal* 2015, *54*, 1115–1126, doi:10.1016/j.aej.2015.07.011.
48. Hasrouny, H.; Samhat, A.E.; Bassil, C.; Laouiti, A. VANet Security Challenges and Solutions: A Survey. *Vehicular Communications* 2017, *7*, 7–20, doi:10.1016/j.vehcom.2017.01.002.
49. Sakiz, F.; Sen, S. A Survey of Attacks and Detection Mechanisms on Intelligent Transportation Systems: VANETs and IoV. *Ad Hoc Networks* 2017, *61*, 33–50, doi:10.1016/j.adhoc.2017.03.006.
50. Parno, B.; Perrig, A. Challenges in Securing Vehicular Networks. In Proceedings of the Workshop on hot topics in networks (HotNets-IV); Maryland, USA, 2005; pp. 1–6.
51. Darraji, E. Secure Opportunistic Routing Protocol Based on Reputation Systems in VANETs. thesis, Loughborough University, 2022.
52. Mejri, M.N. Securing Vehicular Networks Against Denial of Service Attacks. phdthesis, Université Sorbonne Paris Cité ; École nationale d'ingénieurs de Tunis (Tunisie), 2016.
53. Kushwaha, D.; Shukla, P.K.; Baraskar, R. A Survey on Sybil Attack in Vehicular Ad-Hoc Network. *International Journal of Computer Applications* 2014, *98*.
54. Leinmüller, T.; Schoch, E. Greedy Routing in Highway Scenarios: The Impact of Position Faking Nodes. In Proceedings of the Proceedings of Workshop On Intelligent Transportation (WIT 2006)(Mar. 2006); 2006.
55. Nguyen-Minh, H. Contribution to the Intelligent Transportation System : Security of Safety Applications in Vehicle Ad Hoc Networks. phdthesis, Université d'Avignon, 2016.
56. Keerthika, M.; Shanmugapriya, D. Wireless Sensor Networks: Active and Passive Attacks - Vulnerabilities and Countermeasures. *Global Transitions Proceedings* 2021, *2*, 362–367, doi:10.1016/j.gltp.2021.08.045.
57. Alnasser, A.; Sun, H.; Jiang, J. Recommendation-Based Trust Model for Vehicle-to-Everything (V2X). *IEEE Internet of Things Journal* 2020, *7*, 440–450, doi:10.1109/JIOT.2019.2950083.
58. Ben Salem, N.; Buttyan, L.; Hubaux, J.-P.; Jakobsson, M. Node Cooperation in Hybrid Ad Hoc Networks. *IEEE Transactions on Mobile Computing* 2006, *5*, 365–376, doi:10.1109/TMC.2006.1599405.
59. Laberteaux, K.; Hartenstein, H. *VANET: Vehicular Applications and Inter-Networking Technologies*; John Wiley & Sons, 2009;
60. Engoulou, R.G.; Bellaïche, M.; Pierre, S.; Quintero, A. VANET Security Surveys. *Computer Communications* 2014, *44*, 1–13, doi:10.1016/j.comcom.2014.02.020.
61. Srinivasan, A.; Teitelbaum, J.; Wu, J.; Cardei, M.; Liang, H. Reputation-and-Trust-Based Systems for Ad Hoc Networks. *Algorithms and protocols for wireless and mobile ad hoc networks* 2009, *375*, 375–404.

62. Buchegger, S.; Boudec, J.Y.L. Cooperation of Nodes Fairness in Dynamic Ad-Hoc Networks. In Proceedings of the Proceedings of IEEE/ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHOC); 2002.
63. Dotzer, F.; Fischer, L.; Magiera, P. Vars: A Vehicle Ad-Hoc Network Reputation System. In Proceedings of the Sixth IEEE International Symposium on a World of Wireless Mobile and Multimedia Networks; IEEE, 2005; pp. 454–456.
64. Jeyaprakash, T.; Mukesh, R. An Optimized Node Selection Routing Protocol for Vehicular Ad-Hoc Networks – A Hybrid Model. *Journal of Communications Software and Systems* 2015, *11*, 80–85, doi:10.24138/jcomss.v11i2.106.
65. Ma, J.; Yang, C. A Trust-Based Stable Routing Protocol in Vehicular Ad-Hoc Networks. *Int J Secur Appl* 2015, *9*, 107–116.
66. Kim, T.H.-J.; Studer, A.; Dubey, R.; Zhang, X.; Perrig, A.; Bai, F.; Bellur, B.; Iyer, A. Vanet Alert Endorsement Using Multi-Source Filters. In Proceedings of the Proceedings of the seventh ACM international workshop on VehiculAr InterNETworking; 2010; pp. 51–60.
67. Raya, M.; Papadimitratos, P.; Gligor, V.D.; Hubaux, J.-P. On Data-Centric Trust Establishment in Ephemeral Ad Hoc Networks. In Proceedings of the IEEE INFOCOM 2008-The 27th Conference on Computer Communications; IEEE, 2008; pp. 1238–1246.
68. Dhurandher, S.K.; Obaidat, M.S.; Jaiswal, A.; Tiwari, A.; Tyagi, A. Securing Vehicular Networks: A Reputation and Plausibility Checks-Based Approach. In Proceedings of the 2010 IEEE Globecom Workshops; IEEE, 2010; pp. 1550–1554.
69. Lo, N.-W.; Tsai, H.-C. A Reputation System for Traffic Safety Event on Vehicular Ad Hoc Networks. *EURASIP Journal on wireless communications and networking* 2009, *2009*, 1–10.
70. Guo, Y.; Schildt, S.; Morgenroth, J.; Wolf, L. A Misbehavior Detection System for Vehicular Delay Tolerant Networks. *INFORMATIK 2012* 2012.
71. Poongodi, M.; Hamdi, M.; Sharma, A.; Ma, M.; Singh, P.K. DDoS Detection Mechanism Using Trust-Based Evaluation System in VANET. *IEEE Access* 2019, *7*, 183532–183544.
72. Nandy, T.; Noor, R.M.; Idris, M.Y.I.B.; Bhattacharyya, S. T-BCIDS: Trust-Based Collaborative Intrusion Detection System for VANET. In Proceedings of the 2020 National Conference on Emerging Trends on Sustainable Technology and Engineering Applications (NCETSTEA); IEEE, 2020; pp. 1–5.
73. Mustikawati, E.; Perdana, D.; Negara, R.M. Network Security Analysis in Vanet against Black Hole and Jellyfish Attack with Intrusion Detection System Algorithm. *CommIT (Communication and Information Technology) Journal* 2017, *11*, 77–83.
74. Harit, S.K.; Singh, G.; Tyagi, N. Fox-Hole Model for Data-Centric Misbehaviour Detection in Vanets. In Proceedings of the 2012 Third International Conference on Computer and Communication Technology; IEEE, 2012; pp. 271–277.
75. Ruj, S.; Cavenaghi, M.A.; Huang, Z.; Nayak, A.; Stojmenovic, I. On Data-Centric Misbehavior Detection in VANETs. In Proceedings of the 2011 IEEE Vehicular Technology Conference (VTC Fall); Ieee, 2011; pp. 1–5.
76. Wasef, A.; Jiang, Y.; Shen, X. ECMV: Efficient Certificate Management Scheme for Vehicular Networks. In Proceedings of the IEEE GLOBECOM 2008-2008 IEEE Global Telecommunications Conference; IEEE, 2008; pp. 1–5.
77. Calandriello, G.; Papadimitratos, P.; Hubaux, J.-P.; Lioy, A. Efficient and Robust Pseudonymous Authentication in VANET. In Proceedings of the Proceedings of the fourth ACM international workshop on Vehicular ad hoc networks; 2007; pp. 19–28.
78. Vulimiri, A.; Gupta, A.; Roy, P.; Muthaiah, S.N.; Kherani, A.A. Application of Secondary Information for Misbehavior Detection in VANETs. In Proceedings of the NETWORKING 2010: 9th International IFIP TC 6 Networking Conference, Chennai, India, May 11-15, 2010. Proceedings 9; Springer, 2010; pp. 385–396.
79. Lin, X.; Lu, R.; Zhang, C.; Zhu, H.; Ho, P.; Shen, X. Security in Vehicular Ad Hoc Networks. *IEEE Communications Magazine* 2008, *46*, 88–95, doi:10.1109/MCOM.2008.4481346.

80. Bariah, L.; Shehada, D.; Salahat, E.; Yeun, C.Y. Recent Advances in VANET Security: A Survey. In Proceedings of the 2015 IEEE 82nd Vehicular Technology Conference (VTC2015-Fall); September 2015; pp. 1–7.
81. Sanzgiri, K.; Dahill, B.; Levine, B.N.; Shields, C.; Belding-Royer, E.M. A Secure Routing Protocol for Ad Hoc Networks. In Proceedings of the 10th IEEE International Conference on Network Protocols, 2002. Proceedings.; November 2002; pp. 78–87.
82. Hu, Y.-C.; Johnson, D.B.; Perrig, A. SEAD: Secure Efficient Distance Vector Routing for Mobile Wireless Ad Hoc Networks. *Ad Hoc Networks* 2003, *1*, 175–192, doi:10.1016/S1570-8705(03)00019-2.
83. Hu, Y.-C.; Perrig, A.; Johnson, D.B. Ariadne: A Secure on-Demand Routing Protocol for Ad Hoc Networks. In Proceedings of the Proceedings of the 8th annual international conference on Mobile computing and networking; Association for Computing Machinery: New York, NY, USA, September 23 2002; pp. 12–23.
84. A Security Credential Management System for V2V Communications. In Proceedings of the 2013 IEEE Vehicular Networking Conference.
85. Rahbari, M.; Jamali, M.A.J. Efficient Detection of Sybil Attack Based on Cryptography in Vanet 2011.
86. Shamir, A. Identity-Based Cryptosystems and Signature Schemes. In Proceedings of the Advances in Cryptology: Proceedings of CRYPTO 84 4; Springer, 1985; pp. 47–53.
87. Boneh, D.; Franklin, M. Identity-Based Encryption from the Weil Pairing. In Proceedings of the Advances in Cryptology — CRYPTO 2001; Kilian, J., Ed.; Springer: Berlin, Heidelberg, 2001; pp. 213–229.
88. Sun, J.; Zhang, C.; Zhang, Y.; Fang, Y. An Identity-Based Security System for User Privacy in Vehicular Ad Hoc Networks. *IEEE Transactions on Parallel and Distributed Systems* 2010, *21*, 1227–1239, doi:10.1109/TPDS.2010.14.
89. Lu, H.; Li, J.; Guizani, M. A Novel ID-Based Authentication Framework with Adaptive Privacy Preservation for VANETs. In Proceedings of the 2012 Computing, Communications and Applications Conference; IEEE, 2012; pp. 345–350.
90. Kamat, P.; Baliga, A.; Trappe, W. An Identity-Based Security Framework For VANETs. In Proceedings of the Proceedings of the 3rd international workshop on Vehicular ad hoc networks; Association for Computing Machinery: New York, NY, USA, September 29 2006; pp. 94–95.
91. Zhang, L. OTIBAAGKA: A New Security Tool for Cryptographic Mix-Zone Establishment in Vehicular Ad Hoc Networks. *IEEE Transactions on Information Forensics and Security* 2017, *12*, 2998–3010, doi:10.1109/TIFS.2017.2730479.
92. Tzeng, S.-F.; Horng, S.-J.; Li, T.; Wang, X.; Huang, P.-H.; Khan, M.K. Enhancing Security and Privacy for Identity-Based Batch Verification Scheme in VANETs. *IEEE Transactions on Vehicular Technology* 2017, *66*, 3235–3248, doi:10.1109/TVT.2015.2406877.
93. Karati, A.; Islam, S.H.; Biswas, G.P.; Bhuiyan, M.Z.A.; Vijayakumar, P.; Karuppiah, M. Provably Secure Identity-Based Signcryption Scheme for Crowdsourced Industrial Internet of Things Environments. *IEEE Internet of Things Journal* 2018, *5*, 2904–2914, doi:10.1109/JIOT.2017.2741580.
94. Tangade, S.; Manvi, S.S.; Lorenz, P. Trust Management Scheme Based on Hybrid Cryptography for Secure Communications in VANETs. *IEEE Transactions on Vehicular Technology* 2020, *69*, 5232–5243, doi:10.1109/TVT.2020.2981127.
95. Wang, Y.; Zhong, H.; Xu, Y.; Cui, J.; Wu, G. Enhanced Security Identity-Based Privacy-Preserving Authentication Scheme Supporting Revocation for VANETs. *IEEE Systems Journal* 2020, *14*, 5373–5383, doi:10.1109/JSYST.2020.2977670.
96. Chaymae, T.; Elkhatir, H.; Otman, A. Recent Advances in Machine Learning and Deep Learning in Vehicular Ad-Hoc Networks: A Comparative Study. In Proceedings of the The Proceedings of the International Conference on Electrical Systems & Automation; Bendaoud, M., Wolfgang, B., Chikh, K., Eds.; Springer: Singapore, 2022; pp. 1–14.

97. Sharma, A.; Jaekel, A. Machine Learning Based Misbehaviour Detection in VANET Using Consecutive BSM Approach. *IEEE Open Journal of Vehicular Technology* 2022, *3*, 1–14, doi:10.1109/OJVT.2021.3138354.
98. Li, W.; Joshi, A.; Finin, T. SVM-CASE: An SVM-Based Context Aware Security Framework for Vehicular Ad-Hoc Networks. In Proceedings of the 2015 IEEE 82nd Vehicular Technology Conference (VTC2015-Fall); September 2015; pp. 1–5.
99. Doss, S.; Nayyar, A.; Suseendran, G.; Tanwar, S.; Khanna, A.; Thong, P.H. APD-JFAD: Accurate Prevention and Detection of Jelly Fish Attack in MANET. *Ieee Access* 2018, *6*, 56954–56965.
100. Yu, Y.; Guo, L.; Liu, Y.; Zheng, J.; Zong, Y. An Efficient SDN-Based DDoS Attack Detection and Rapid Response Platform in Vehicular Networks. *IEEE Access* 2018, *6*, 44570–44579, doi:10.1109/ACCESS.2018.2854567.
101. Adhikary, K.; Bhushan, S.; Kumar, S.; Dutta, K. Hybrid Algorithm to Detect DDoS Attacks in VANETs. *Wireless Pers Commun* 2020, *114*, 3613–3634, doi:10.1007/s11277-020-07549-y.
102. Aloqaily, M.; Otoum, S.; Ridhawi, I.A.; Jararweh, Y. An Intrusion Detection System for Connected Vehicles in Smart Cities. *Ad Hoc Networks* 2019, *90*, 101842, doi:10.1016/j.adhoc.2019.02.001.
103. Grover, J.; Prajapati, N.K.; Laxmi, V.; Gaur, M.S. Machine Learning Approach for Multiple Misbehavior Detection in VANET. In Proceedings of the Advances in Computing and Communications; Abraham, A., Mauri, J.L., Buford, J.F., Suzuki, J., Thampi, S.M., Eds.; Springer: Berlin, Heidelberg, 2011; pp. 644–653.
104. Kosmanos, D.; Pappas, A.; Maglaras, L.; Moschoyiannis, S.; Aparicio-Navarro, F.J.; Argyriou, A.; Janicke, H. A Novel Intrusion Detection System against Spoofing Attacks in Connected Electric Vehicles. *Array* 2020, *5*, 100013, doi:10.1016/j.array.2019.100013.
105. Ghaleb, F.A.; Zainal, A.; Rassam, M.A.; Mohammed, F. An Effective Misbehavior Detection Model Using Artificial Neural Network for Vehicular Ad Hoc Network Applications. In Proceedings of the 2017 IEEE Conference on Application, Information and Network Security (AINS); November 2017; pp. 13–18.
106. Maria, A.; Rajasekaran, A.S.; Al-Turjman, F.; Altrjman, C.; Mostarda, L. Baiv: An Efficient Blockchain-Based Anonymous Authentication and Integrity Preservation Scheme for Secure Communication in VANETs. *Electronics* 2022, *11*, 488.
107. Jiang, Y.; Shen, X.; Zheng, S. An Effective Data Sharing Scheme Based on Blockchain in Vehicular Social Networks. *Electronics* 2021, *10*, 114.
108. Miller, V.S. Use of Elliptic Curves in Cryptography. In Proceedings of the Advances in Cryptology — CRYPTO '85 Proceedings; Williams, H.C., Ed.; Springer: Berlin, Heidelberg, 1986; pp. 417–426.
109. Hess, F. Efficient Identity Based Signature Schemes Based on Pairings. In Proceedings of the Selected Areas in Cryptography: 9th Annual International Workshop, SAC 2002 St. John's, Newfoundland, Canada, August 15–16, 2002 Revised Papers 9; Springer, 2003; pp. 310–324.
110. Zaimi, I.; Boushaba, A.; Squalli Houssaini, Z.; Oumsis, M. A Fuzzy Geographical Routing Approach to Support Real-Time Multimedia Transmission for Vehicular Ad Hoc Networks. *Wireless Netw* 2019, *25*, 1289–1311, doi:10.1007/s11276-018-1729-9.
111. Al-Ani, A.D.; Seitz, J. QoS-Aware Routing in Multi-Rate Ad Hoc Networks Based on Ant Colony Optimization. *Netw. Protoc. Algorithms* 2015, *7*, 1–25.
112. Ye, B.; Jayasumana, A.P.; Piratla, N.M. On Monitoring of End-to-End Packet Reordering over the Internet. In Proceedings of the International conference on Networking and Services (ICNS'06); July 2006; pp. 3–3.
113. Jayasumana, A.; Piratla, N.; Banka, T.; Bare, A.; Whitner, R. *Improved Packet Reordering Metrics*; 2008;
114. Zhang, D.; Gong, C.; Jiang, K.; Zhang, X.; Zhang, T. A Kind of New Method of Intelligent Trust Engineering Metrics (ITEM) for Application of Mobile Ad Hoc Network. *Engineering Computations* 2020.
115. The Network Simulator - Ns-2 Available online: https://www.isi.edu/nsnam/ns/ (accessed on 3 April 2022).
116. nsnam Ns-3 Available online: https://www.nsnam.org/ (accessed on 8 September 2022).

117. OPNET Network Simulator. *Opnet Projects*.
118. Aljabry, I.A.; Al-Suhail, G.A. A Survey on Network Simulators for Vehicular Ad-Hoc Networks (VANETS). *Int. J. Comput. Appl* 2021, *174*, 1–9.
119. Hussain, S.A.; Saeed, A. An Analysis of Simulators for Vehicular Ad Hoc Networks. *World Applied Sciences Journal* 2013, *23*, 1044–1048.
120. Kumar, P.; Shukla, S. A Survey of Simulation Tools for VANET. *Journal of Analysis and Computation (JAC)* 2019, 1–2.
121. OMNeT++ Discrete Event Simulator Available online: https://omnetpp.org/ (accessed on 16 April 2022).
122. Eclipse SUMO - Simulation of Urban MObility Available online: https://www.eclipse.org/sumo/ (accessed on 21 June 2022).
123. Azzaoui, N.; Korichi, A.; Brik, B.; Amirat, H. A Survey on Data Dissemination in Internet of Vehicles Networks. *Journal of Location Based Services* 2022, 1–44.
124. Härri, J.; Filali, F.; Bonnet, C.; Fiore, M. VanetMobiSim: Generating Realistic Mobility Patterns for VANETs. In Proceedings of the Proceedings of the 3rd international workshop on Vehicular ad hoc networks; 2006; pp. 96–97.
125. Martinez, F.J.; Toh, C.K.; Cano, J.-C.; Calafate, C.T.; Manzoni, P. A Survey and Comparative Study of Simulators for Vehicular Ad Hoc Networks (VANETs). *Wireless Communications and Mobile Computing* 2011, *11*, 813–828.
126. Veins Available online: https://veins.car2x.org/download/ (accessed on 28 May 2022).
127. Amoozadeh, M.; Ching, B.; Chuah, C.-N.; Ghosal, D.; Zhang, H.M. VENTOS: Vehicular Network Open Simulator with Hardware-in-the-Loop Support. *Procedia Computer Science* 2019, *151*, 61–68.