Security of V2X communications in 3GPP - 5G cellular networks

Hadi Yakan

université PARIS-SACLAY

Security of V2X communications in 3GPP - 5G cellular networks

Sécurité des communications V2X dans les réseaux cellulaires 5G - 3GPP

Thèse de doctorat de l'Université Paris-Saclay

École doctorale n° 580: Sciences et Technologies de l'Information et de la Communication (STIC)
Spécialité de doctorat: Sciences des réseaux, de l'information et de la communication
Graduate School: Informatique et sciences du numérique
Référent: Université de Versailles-Saint-Quentin-en-Yvelines

Thèse préparée dans l'unité de recherche **DAVID** (Université Paris-Saclay, UVSQ) sous la direction de Nadjib AITSAADI, Professeur des universités, le co-encadrement de Ilhem FAJJARI, Chef de projet de recherche

Thèse soutenue à Versailles, le 24 novembre 2023, par

Hadi YAKAN

Composition du jury

Membres du jury avec voix délibérative

Nathalie MITTON Directrice de recherche, Inria Lille-Nord Europe	Présidente
Hassine MOUNGLA Professeur des universités, Université Paris-Descartes	Rapporteur & Examinateur
Fen ZHOU Maître de conférences - HDR, Avignon Université	Rapporteur & Examinateur
Rémi BADONNEL Professeur des universités, TELECOM Nancy - Université de Lorraine	Examinateur
Pierre MERDRIGNAC Responsable projets connectivité, Vedecom	Examinateur

THESE DE DOCTORAT

NNT : 2023UPASG077

université PARIS-SACLAY

ÉCOLE DOCTORALE
Sciences et technologies de l'information et de la communication (STIC)

Titre : Sécurité des communications V2X dans les réseaux cellulaires 5G - 3GPP
Mots clés : C-ITS, V2X, 5G, Securité, Misbehavior Detection, IA

Résumé :

Avec les avancées technologiques apportées par les réseaux 5G, une nouvelle ère de communications de Vehicle-to-Everything (V2X) est apparue, offrant des applications nouvelles et avancées en matière de sécurité, d'efficacité et d'autres expériences de conduite dans les systèmes de transport intelligents (ITS). Cependant, les nouvelles fonctionnalités s'accompagnent de nouveaux défis en matière de sécurité, en particulier dans le domaine des communications Vehicle-to-Network (V2N).

Cette thèse se concentre sur l'application des systèmes de détection de comportements anormaux dans les communications V2X au sein des réseaux 5G. Tout d'abord, nous présentons un nouveau système de détection de comportements anormaux, intégré au réseau central 5G pour détecter et prévenir les attaques V2X. Ensuite, nous proposons un schéma de collaboration entre les nœuds de détection afin d'améliorer les résultats de la détection dans les réseaux 5G edge. Enfin, nous proposons d'utiliser le Federated Learning pour permettre un entraînement distribué et nous évaluons les performances sur une grande variété d'attaques V2X.

Title: Security of V2X communications in 3GPP - 5G cellular networks
Keywords: C-ITS, V2X, 5G, Security, Misbehavior Detection, AI

Abstract:

The introduction of 5G networks has brought significant technical improvements; a new era of Vehicle-to-Everything (V2X) communications has emerged, offering new and advanced safety, efficiency, and other driving experience applications in the Intelligent Transport Systems (ITS). However, with new features come new security challenges, especially in the realm of Vehicle-to-Network (V2N) communications.

This thesis focuses on the application of misbehavior detection in V2X communications within 5G networks. First, we introduce a novel misbehavior detection system integrated with 5G core (5GC) network to detect and prevent V2X attacks. Then, we propose a collaboration scheme between detection nodes to improve detection results in 5G edge networks. Last, we leverage Federated Learning to enable distributed training, and we assess the performance on a wide variety of V2X attacks.

Résumé

Introduction : Chaque année, environ 1,35 million de personnes perdent la vie dans des accidents de la route dans le monde, l'OMS estime qu'une personne perd la vie toutes les 24 secondes. Le nombre de fatalités routières reste élevé malgré les efforts juridiques et législatifs déployés pour améliorer la sécurité routière. Depuis 1999, des technologies, des protocoles et des applications ITS ont été développés dans le but de renforcer la sécurité routière et de réduire le nombre d'accidents. Cette thèse se concentre sur la dernière génération (NR-V2X) et l'amélioration de la sécurité des communications véhiculaires dans les réseaux 5G. Elle aborde les défis posés par les attaques de sécurité et propose des solutions innovantes pour renforcer la fiabilité et la sécurité des communications V2N.

Chapitre 2 : Étude de la littérature de la sécurité des réseaux véhiculaires : Ce chapitre examine en détail les protocoles et les spécifications ITS et V2X de l'ETSI et du 3GPP. Il présente également une revue complète de la littérature existante sur la sécurité V2X et les systèmes de détection de comportements anormaux. Il identifie les lacunes dans la recherche actuelle, établissant ainsi le contexte pour les développements proposés dans les chapitres suivants. En examinant les recherches antérieures, le chapitre identifie les lacunes et les défis non résolus, établissant ainsi un cadre de référence pour les innovations proposées dans les chapitres suivants de la thèse. Cette revue permet de comprendre les tendances actuelles, les meilleures pratiques, et les limites des solutions existantes en matière de sécurité V2X.

Chapitre 3 : Une nouvelle fonction application de la sécurité du réseau central 5G pour la couche Facilities de C-ITS basée sur l'IA : Ce chapitre détaille le développement d'un système novateur de détection de comportement anormal, spécifiquement conçu pour identifier et prévenir les attaques de falsification de position dans les communications V2X, et protéger les serveurs V2X dans le contexte des réseaux 5G. Le système est soigneusement aligné avec les spécifications de l'architecture 5G-V2X de la 3GPP, garantissant ainsi une intégration et une compatibilité optimales. Le système proposé s'appuie sur des techniques de machine learning, et offre une détection précise et efficace des menaces, en proposant un nouveau contrôle de plausibilité (ORPC) pour améliorer les performances de détection.

Chapitre 4 : Détection des comportements V2X anormaux en tant que fonction de réseau 5G central/edge basée sur l'IA/ML : Dans ce chapitre, l'accent est mis sur l'exploration d'un système collaboratif de détection de comportement anormal V2X, qui représente une avancée par rapport au système introduit dans le chapitre précédent. Ce système tire parti de la collaboration entre les nœuds des réseaux edge pour améliorer la détection des comportements anormaux, contribuant ainsi à renforcer la sécurité des serveurs d'applications V2X dans les réseaux 5G edge. Le chapitre présente en détail l'architecture du système, les mécanismes de collaboration en comparant cette approche collaborative avec les systèmes de détection centralisés. Les résultats des simulations démontrent une amélioration dans la capacité du système à identifier et à répondre aux menaces de sécurité, soulignant l'efficacité de la collaboration dans le contexte des communications V2X sécurisées dans les réseaux 5G.

Chapitre 5 : Federated Learning pour la détection de comportement anormal V2X dans les réseaux 5G : Ce chapitre aborde en profondeur la faisabilité et l'efficacité de l'utilisation de l'apprentissage fédéré pour la détection de comportements anormaux dans les réseaux V2X au sein des réseaux 5G edge. Il compare la performance de détection des modèles d'apprentissage fédéré avec celle des modèles centralisés et autonomes, mettant en évidence les forces et les limitations potentielles de chaque approche. À travers des évaluations détaillées, il démontre que l'apprentissage fédéré préserve non seulement la confidentialité des données, mais maintient également une précision de détection comparable, voire supérieure. Cette investigation de l'apprentissage fédéré ouvre de nouvelles voies pour son application dans les réseaux 5G edge, démontrant son potentiel en tant que solution évolutive et efficace pour renforcer la sécurité des communications V2X.

Conclusion : La thèse résume les découvertes clés, soulignant leur importance pour la sécurité des communications V2X et des réseaux 5G. Elle propose également des pistes pour des recherches futures, ouvrant la voie à des développements supplémentaires dans ce domaine en pleine expansion.

Acknowledgments

I would like to express my deepest gratitude to my thesis supervisor, **Professor Nadjib AITSAADI**, for their unwavering support, invaluable guidance, and insightful feedback throughout the course of this research. Their expertise and mentorship have been fundamental in shaping both the direction and success of this work.

Special thanks are due to my thesis advisor, **Doctor Ilhem FAJJARI**, whose expertise and constructive criticism have greatly contributed to refining my writing and research.

I am also grateful to **Doctor Cédric ADJIH**, whose collaborations and insightful discussions on artificial intelligence have enriched this research.

Finally, I would like to express my gratitude to DAVID Lab and Université de Versailles Saint-Quentin-en-Yvelines (UVSQ) for providing the environment and infrastructure required for my research. I also acknowledge the SARWS Project and Université Paris-Est Créteil (UPEC) for the opportunities they provided me during my research. I am also thankful to my colleagues, whose support and exchange of ideas have been a source of motivation and inspiration.

Dedication

I would like to dedicate this work to my parents, my grandmother, my sisters, my brother-in-law, my nephews, all my uncles, aunts, cousins, and friends, and to the souls of my late grandmother and late grandfathers.

Contents

List of Figures

List of Tables

1 - INTRODUCTION

Contents

1.1 . Background

Every year, approximately 1.35 million people lose their lives due to road accidents worldwide, the World Health Organization (WHO) estimates the loss of one life every 24 seconds [1]. The number of road fatalities remains unacceptably high despite legal and legislative efforts to improve road safety. Since 1999, Intelligent Transport Systems (ITS) technologies, protocols, and applications have been developed in an effort to increase traffic safety and lessen accidents.

However, the first generation ITS technology, Dedicated Short-Range Communications (DSRC), faced several challenges that prevented its wide adoption. The main challenges were the slow pace and the high deployment cost of the required infrastructure.

Additionally, there was a market fragmentation in the automotive industry between companies adopting DSRC and others preferring to wait for the next-generation ITS technology, namely Cellular Vehicle-to-Everything (C-V2X). Another significant challenge was the recent reallocation of the spectrum [2],

initially reserved for DSRC, to C-V2X and Wi-Fi by the Federal Communication Commission (FCC).

In 2023, C-V2X seems to address most of these challenges, by leveraging the widely available cellular network infrastructure. It offers increased bandwidth, lower latency, and wider network coverage which can empower advanced network-hosted ITS services.

In addition to **road safety** applications, ITS can offer many other benefits, such as:

- **Traffic efficiency:** like traffic management and parking solutions.
- **Support for Connected Autonomous Vehicles (CAV) and cooperative driving:** by providing drivers and autonomous driving systems enhanced perception, and the ability to perform coordinated movement and maneuvers.
- **Environmental benefits:** traffic management applications can contribute to emissions reduction and improved fuel efficiency, providing environmental advantages.
- **Enhanced road user experience:** this includes real-time information about road conditions, traffic, and weather, as well as features like automatic toll payments, electric vehicle charging reservations, and in-vehicle entertainment services.

1.1.1 . Vehicle-to-Everything (V2X)

ITS refers to the initiatives of utilizing the latest technologies in electronics, information, and telecommunication to enhance safety, efficiency, sustainability, and comfort of transportation systems; Cooperative Intelligent Transport Systems (C-ITS) are ITS which harness the power of communication between two or more ITS stations (vehicle, roadside, central, mobile device) to provide advanced ITS services, like: Vulnerable Road User (VRU) warning, dangerous situation warning, cooperative overtaking, and platooning.

While C-ITS defines the systems and applications that interact directly with road users, Vehicle-to-Everything (V2X) refers to the underlying communication protocol stacks and access technologies which enable C-ITS services.

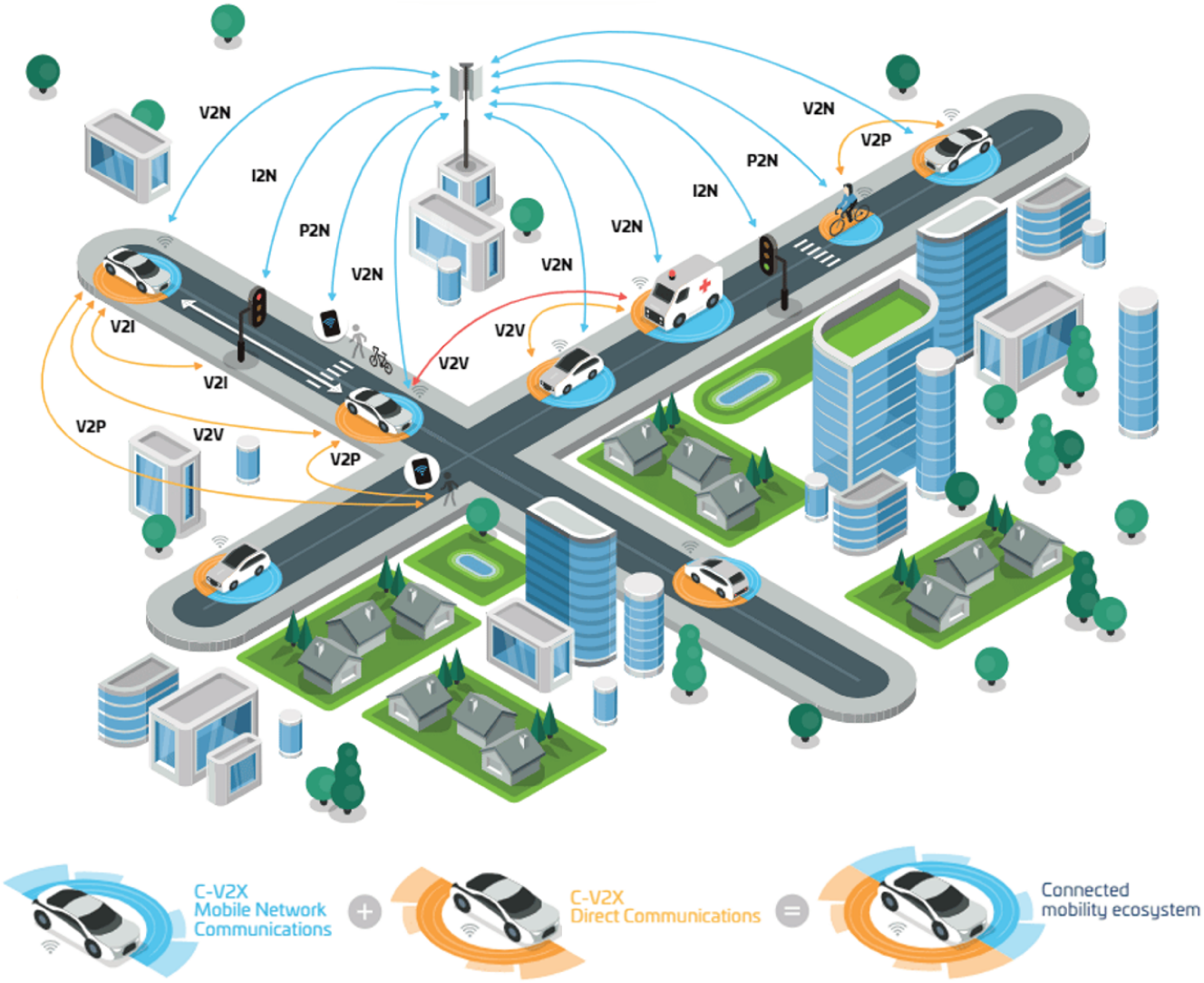

Figure 1.1: V2X communications (Source: 5GAA)

V2X communications can take multiple forms, as depicted in Figure 1.1:

- Vehicle-to-Vehicle (V2V) allows ad-hoc communications between vehicles for safety-related use cases.
- Vehicle-to-Pedestrian (V2P) takes place between vehicles and pedestrians' smartphones to enable VRU use cases.

- Vehicle-to-Infrastructure (V2I) enables communication between an On-Board Unit (OBU) of a vehicle and a Road-Side Unit (RSU) of road infrastructure such as V2X-based traffic lights.
- Vehicle-to-Network (V2N) equips vehicles with wide area network capabilities to communicate with ITS application servers and other services hosted in the network (Internet, Cloud, or Edge servers) to enable a variety of network-based ITS services.

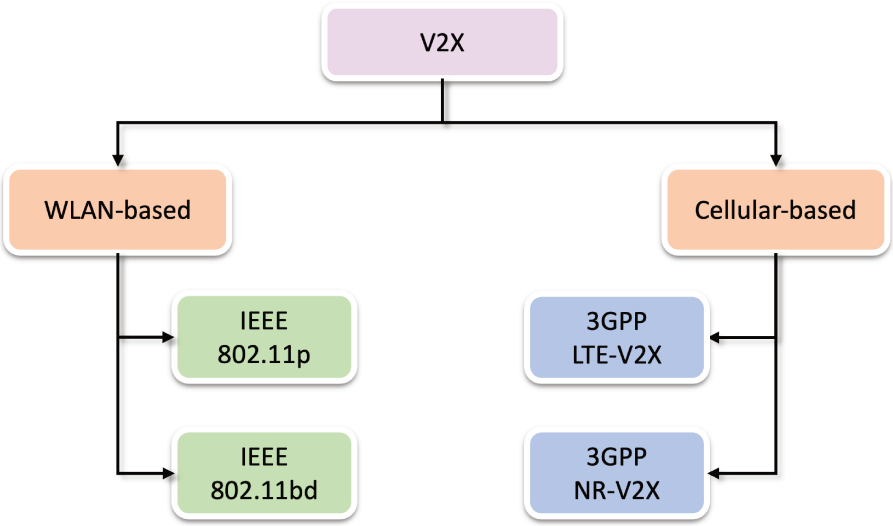

Figure 1.2: V2X Access Technology Standards

While many access technologies were considered to be utilized for V2X communications, the main ones can be divided into two categories, as depicted in Figure 1.2.

- Wireless Local Area Network (WLAN)-based: The first generation of V2X, developed by the Institute of Electrical and Electronics Engineers (IEEE), is based on IEEE 802.11p [3] and IEEE Wireless Access in Vehicular Environments (WAVE) 1609.x standards [4] [5] [6], both of which fall under DSRC, a wireless communication technology designed for short to medium range vehicular communication. A newer standard, which is called IEEE 802.11bd, is under development [7] [8]. A variation of these standards is defined in Europe by the European Telecommunications

Standards Institute (ETSI), namely ETSI ITS-G5 [9] and ETSI ITS standards [10].

- Cellular-based: The next-generation of V2X, developed by the The 3rd Generation Partnership Project (3GPP), is coined C-V2X. It re-utilizes IEEE 1609.x standard for the upper layers, while replacing IEEE 802.11p with cellular-based radio access technologies: Fourth-generation of mobile telecommunications technology (4G) Long Term Evolution based Vehicle-to-Everything (LTE-V2X) and Fifth-generation of mobile telecommunications technology (5G) New Radio based Vehicle-to-Everything (NR-V2X), which are detailed in Chapter 2.

1.1.2 . V2X Challenges

To fully unlock the potential of V2X, several significant challenges must be addressed. Among them the interoperability. Specifically, different manufacturers may employ distinct communication standards, resulting in potential compatibility challenges. Therefore, it is imperative to establish a universally accepted standard to facilitate seamless communication. From a regulatory and legal perspectives, standardization stands out as a primary concern. Regulatory bodies must establish standardized protocols for V2X communications to guarantee consistency and interoperability across all vehicles, devices, and regions.

Another technical concern involves minimizing latency and efficiently allocating resources in a highly mobile network environment. In scenarios where rapid decision-making is crucial, particularly in high-velocity situations, it is essential to minimize communication delays. Ensuring optimal latency is critical for the effectiveness of V2X communications.

Another challenge in this category pertains to the complexity of establishing accountability, especially in the context of accidents or malfunctions. This complexity is further amplified when multiple entities, including vehicles, infrastructure, and pedestrians, are involved in the communication process.

Besides, the scalability of V2X security solutions represents another challenge especially in the near future, when the number of connected vehicles starts to rapidly increase. It is highly important to ensure that these solutions can keep up with the high amount of traffic they might need to process.

On the economic front, the substantial costs associated with V2X pose a significant barrier. The initial investment required for V2X infrastructure, both in vehicles and on roads, can be substantial. Furthermore, identifying feasible and sustainable business models for V2X is challenging, especially when considering that the technology is still in its early development phase.

Additionally, there are social challenges to consider. For V2X to be successful, it is essential for the general public to trust and embrace the technology. Addressing concerns, particularly those related to the handling of location data, is crucial. Moreover, there is a need to assess the potential impact of technology and automation on job redundancy, especially within the transportation sector, which is a subject of concern.

1.1.3 . V2X Security Challenges

While V2X communications bring significant advantages and benefits, they also introduce a variety of unprecedented security challenges, due the dynamic and complex nature of V2X networks. These challenges must be taken seriously to ensure the safety of road users and the reliability of ITS services.

1.1.3.1 . Authentication and Authorization

Authentication is the first layer of protection in all systems. In the context of V2X, authentication mitigates the risk of unauthorized devices or vehicles injecting false data into the system. Also, it is important to ensure that special V2X services, like emergency vehicle warning, can only be accessed by authorized vehicles.

1.1.3.2 . Data Integrity and Misbehavior Detection

As V2X applications heavily rely on the accuracy and timeliness of the data they receive, any compromise in data integrity, whether due to interference, malfunction, or transmission errors, can lead to vehicles or applications making decisions based on incorrect information. This situation endangers the safety of the vehicle's occupants and other road users.

It is essential to enforce secure transmissions through implementing error-checking and validation protocols, coupled with detection of anomalies or inconsistencies in the data and reporting them for review. It is crucial to ensure that compromised authenticated devices are quickly identified and disconnected from the network. The implementation of effective detection and re-

vocation systems, capable of promptly responding to any threats, is vital.

1.1.3.3 . Data Privacy

V2X systems can produce a high volume of data, collecting an extensive information such as the location, speed, direction of vehicles, and even certain aspects of driver behavior. The persistent flow of data gives rise to substantial concerns regarding data privacy and storage. The collection, storage, and processing of this type of data need to be under strict control and monitoring.

Furthermore, the potential sharing of V2X data with external entities introduces an additional level of complexity. While there might be legitimate reasons for data sharing, such as traffic management or targeted marketing, it is important to acknowledge the potential risks associated with accidental data leak resulting from system breaches or illegal access.

Anonymization of data is often suggested as a solution to address privacy-related concerns. However, even anonymized data isn't entirely secure. Indeed, advanced correlation techniques can sometimes de-anonymize this data, leading to potential privacy breaches and exposing sensitive user information.

1.1.3.4 . Cybersecurity Concerns

As vehicles evolve into moving computers, they become attractive targets for hackers. A successful attack, where a malicious actor gains control of a vehicle's systems, can have disastrous consequences.

The risks aren't limited to just the vehicles. The infrastructure that supports V2X communications, such as traffic lights, sensors, communication towers, and V2X application servers, is equally vulnerable. A compromised component of infrastructure can cause interruptions to traffic flow, accidents, and potential loss of life.

The threats also encompasses malware and ransomware attacks. For instance, a malware that feeds false data to the driver or other vehicles, causing confusion and accidents. Alternatively, a vehicle is rendered inoperable and held hostage until a ransom is paid.

Also, man-in-the-middle attacks are noteworthy threats, where malicious actors might intercept and manipulate communications between vehicles, infrastructure, or network. Replay and Sybil attacks, where the attacker resends captured V2X messages or flood the network with false information

pose a further complicated challenge.

The rapidly evolving landscape of cyber-threats requires the implementation of a comprehensive cyber-security management program that enables all the major stakeholders (i.e., car manufacturers, vehicular infrastructure providers, telecommunication providers, and authorities) to respond to new attacks. This program has to include Over-The-Air (OTA) updates to participating vehicles and devices to ensure the timely patching of newly discovered vulnerabilities.

1.1.3.5 . Legal and Regulatory Challenges

The regulatory environment related to V2X technology is is continually evolving. Data ownership is a significant topic that requires attention. The identification of data ownership involves various stakeholders including vehicle owners, manufacturers, infrastructure providers, and other entities, and it is a complex subject involving some legal consequences.

Furthermore, with the increasing global adoption of V2X technology, complying with regulatory requirements in multiple countries becomes more challenging. Different countries and regions have distinct laws and regulations related to the protection of data privacy, such as the General Data Protection Regulation (GDPR) implemented in Europe. For manufacturers and service providers operating on a worldwide scale, the responsibility of maintaining compliance with various regulatory frameworks can pose significant challenges.

1.2 . Thesis Problematic

Most of the proposed V2X security solutions are based on cryptography [11], and the majority consists of creating a vehicular Public Key Infrastructure (PKI) system to distribute and verify signed certificates for eligible vehicles. While a PKI system is essential to protect against external threats, different approaches should be considered to mitigate attacks launched by **malicious insiders**, who are authenticated and already part of the system. The most effective solution is to implement a misbehavior detection system that monitors and analyzes the data sent by authenticated vehicles and reports potential unusual behaviors. Researchers addressed the implementation of V2X mis-

behavior detection system while assuming V2V communications [12]. Their proposed solutions are critical; yet they are not suitable for V2N due to the potential possibility of a vehicle exhibiting normal behavior on V2V while misbehaving on V2N. To the best of our knowledge, none tackled the **protection of V2X application servers in a 5G V2N environment** against large-scale data manipulation attacks launched by authenticated misbehaving vehicles. These attacks can result in significant consequences.

Among the numerous threats that we can encounter, position data integrity is especially crucial due to its central role in various ITS applications, including real-time traffic information exchange, advanced driver assistance systems, and autonomous driving. The accuracy and integrity of position data are of paramount importance. During a position falsification attack, the attacker manipulates transmitted position data, creating a deceptive representation of the vehicle's actual location. This can lead to a range of issues, from minor disruptions in traffic flow to significant road accidents. For instance, a vehicle that inaccurately declares its presence in a designated lane or at a specific geographical position may prompt unnecessary evasive maneuvers by other vehicles, potentially resulting in traffic disturbances or even collisions.

Therefore, the detection and prevention of such attacks are critical to ensure the safety and efficiency of V2X communications. However, the detection of position falsification attacks is a complex task that requires sophisticated techniques and systems. Traditional security measures, such as cryptography techniques, are not sufficient to detect these attacks as they can only verify the authentication of the sender but not the trustworthiness of the message content.

1.2.1 . Research Motivation

Considerable academic research has been dedicated to the analysis of misbehavior detection in V2V communications. These studies made notable contributions to improve the security of Vehicular Ad-Hoc Network (VANET) through the detection and elimination of misbehaving nodes. However, these solutions do not monitor V2N traffic, therefore a significant gap exists within misbehavior detection for V2N communications, specifically within the domain of 5G networks.

V2N communications play a vital role in the V2X communications ecosys-

tem by enabling the transmission of information between vehicles and network components. The transmission of information plays a crucial role in the operation of intelligent transportation systems, facilitating functionalities such as centralized traffic management, early hazard warnings, intelligent path selection, and other services based on cloud computing.

The objective of this thesis is to address this gap by studying misbehavior detection in V2N within the context of 5G networks. This work aims to leverage Artificial Intelligence (AI) technologies, specifically machine learning, deep learning, and federated learning, to propose and implement innovative systems for detecting and mitigating position falsification attacks in V2N communications.

1.2.2 . Research Objectives

The main objective of this work is to investigate innovative solutions for enhancing the security of V2X communications within 5G networks. This thesis focuses specifically on the development and evaluation of misbehavior detection systems capable of detecting and mitigating position falsification attacks in 5G V2N communications. The objectives of the investigation can be further specified as follows:

O1: Design and implement a novel misbehavior detection system, coupled with the 5G network, capable of detecting and preventing position falsification attacks in 5G V2N communications. This system will take profit from machine learning techniques to identify anomalies in position data and detect possible attacks. Compatibility with 3GPP 5G V2X architecture specifications will ensure the system's applicability in real-world scenarios.

O2: Investigate the feasibility of a collaborative approach to V2X misbehavior detection. This study will investigate how cooperation between edge network nodes can improve the performances of the malicious behavior detection system. The collaborative system will be implemented to protect V2X application servers in the 5G edge network, thereby enhancing the security of V2X communications.

O3: Investigate federated learning for V2N misbehavior detection in 5G edge networks. Federated learning is an approach to machine learning that

enables the collaborative training of models across multiple decentralised devices or servers containing local data samples without sharing the data. This research will investigate how federated learning can be implemented in 5G edge networks in order to improve the scalability of the misbehavior detection system.

1.3 . Contributions

To address the aforementioned problematic, we shared our three contributions in the conference papers below:

C1: H. Yakan, I. Fajjari, N. Aitsaadi, C. Adjih, "A Novel AI Security Application Function of 5G Core Network for V2X C-ITS Facilities Layer", in IEEE International Conference on Communications (ICC) 2023. (Accepted)

C2: H. Yakan, I. Fajjari, N. Aitsaadi, C. Adjih, "5G V2X Misbehavior Detection as Edge Core Network Function based on AI/ML", in IEEE Global Communications Conference (GLOBECOM) 2023. (Accepted)

C3: H. Yakan, I. Fajjari, N. Aitsaadi, C. Adjih, "Federated Learning for V2X Misbehavior Detection System in 5G Edge Networks", in ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM) 2023. (Accepted as Full Paper)

Furthermore, we had the opportunity to contribute to the European project SARWS [13]: real-time location-aware road weather services composed from multi-modal data. We led Task 3.4 of the project, titled "Security of V2X communications and applications". It was a collaborative effort between multiple academic and industry partners to generate Deliverable D3.4, which is a technical report on the security aspect of the project.

1.4 . Thesis Outline

This thesis is divided into six chapters, organized as follows:

Chapter 2: Literature Study. It gives insights into the existing literature on V2X security and misbehavior detection systems. Specifically, it provides an analysis of the current state of research in the field and identifies research deficiencies that will be addressed by this thesis.

Chapter 3: Development of a Novel Misbehavior Detection System in 5G V2N. This chapter presents an novel system for detecting and preventing position falsification attacks in V2X communications. The system leverages machine learning techniques and conforms to the 3GPP 5G V2X architecture requirements. In this chapter, the system's architecture, machine learning model design, implementation details, and evaluation results are detailed.

Chapter 4: Exploration of a Collaborative V2X Misbehavior Detection System. This chapter investigates a collaborative machine learning model for V2X misbehavior detection, expanding on the concept introduced in Chapter 3. It explains how collaboration between edge detection nodes can improve the detection results of the misbehavior detection system and protect V2X application servers in the 5G edge network.

Chapter 5: Investigation of Federated Learning for V2X Misbehavior Detection. This chapter examines the application of federated learning for detecting V2X misbehaviour in 5G edge networks. It compares the detection performances of distributed and centralized V2X misbehavior detection system in 5G edge networks.

Chapter 6: Summary - The concluding chapter of the thesis provides a summary of the main research findings. The implications of these findings and the future work for V2X communications and cellular network security are discussed.

2 - LITERATURE STUDY

Contents

2.1 . ETSI ITS Architecture

To better understand our solution architecture detailed in Chapters 3, 4, and 5, it is necessary to give insights into the ITS station reference architecture, the ITS security architecture defined by ETSI, and the 5G V2X architecture proposed by 3GPP. We will also discuss in this chapter the state-of-the-art of V2X misbehavior detection.

2.1.1 . ITS-S Reference Architecture

Several standard organizations are working on the development of ITS standards. While their published standards are generally similar due to harmonization efforts, they might have minor differences to comply with certain legal or technical requirements. Within the framework of our work, we will focus on the standards developed by ETSI and 3GPP.

An ITS Station (ITS-S) is a communication device that participates in an ITS network. ITS-S can be part of an OBU on a vehicle or a pedestrian mobile equipment, an RSU, or an ITS application server.

The ITS-S protocol stack, depicted in Figure 2.1, encompasses six layers, four horizontal layers: ITS Applications Layer, Facilities Layer, Networking and

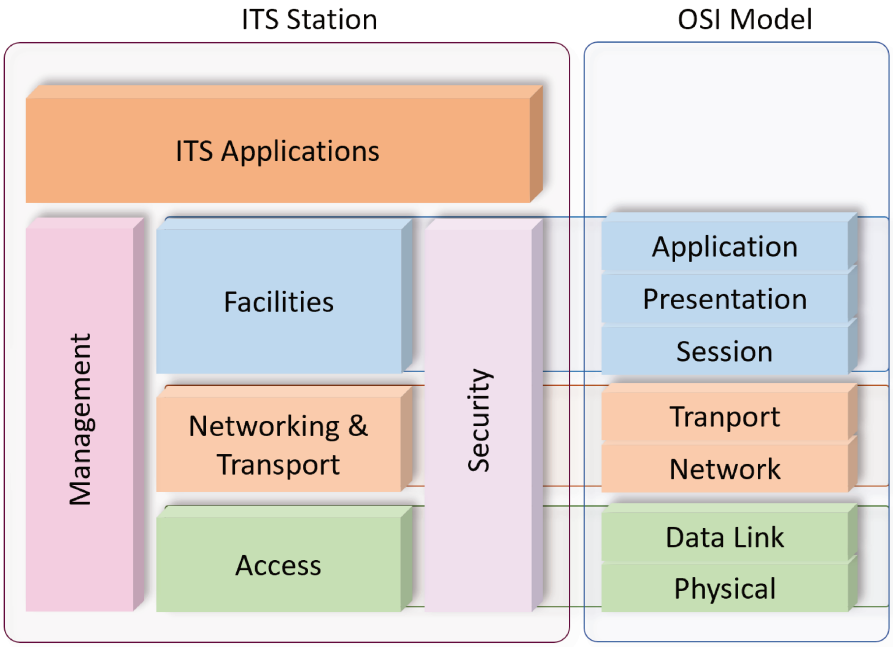

Figure 2.1: ETSI ITS Station Reference Architecture [10]

Transport Layer, Access Layer, and two vertical layers: Management Layer, and Security Layer.

2.1.1.1 . ITS Applications Layer

The ITS Applications Layer [14] is considered as an extension to the Application layer of the Open Systems Interconnection (OSI) [15] reference model. ITS applications are responsible for processing information to deliver ITS services. ETSI divides them into three main classes: i) Road Safety, ii) Traffic Efficiency, and iii) other Applications [16]. An ITS Application must be certified for the actions it performs, and must register with the ITS-S management entity and provide its communication requirements.

2.1.1.2 . Facilities Layer

The Facilities Layer [14] supports the ITS Applications Layer by providing a set of essential functionalities and services that can be simultaneously utilized by multiple ITS applications. These services include positioning, event triggering, timestamping, geostamping, and access to Local Dynamic Map (LDM) [17] [18], which holds important information of surrounding objects. The Facilities

Layer facilitates the collection of information from the surrounding environment and sharing them with multiple ITS applications that might need them.

Besides controlling session management, the Facilities Layer also specifies the messages format and their sending frequency, e.g. ETSI's Cooperative Awareness Messages (CAM) [19] and Decentralized Environmental Notification Messages (DENM) [20] are two essential basic messages used for road safety defined at the Facilities Layer.

CAM messages are periodically sent by the ITS-S to share its position, velocity, and heading at different frequencies ranging between 10 and 100 milliseconds, depending on multiple predefined factors.

DENM messages are used to broadcast and notify about certain road events, like accidents, upcoming hazard, slippery road, or traffic jam, etc.

The Facilities layer provides other application-support services such as time-stamping and geo-stamping of V2X messages, which are essential for ensuring the integrity and relevance of the messages. Additionally, this layer manages the publish/subscribe mechanism for known data objects, enabling ITS applications on higher layers to process LDM data.

2.1.1.3 . Networking and Transport Layer

The Networking and Transport layer is positioned immediately below the Facilities Layer. As the name implies, this layer combines both Transport and Network layers of the OSI reference model.

Multiple communication protocols can be utilized within this layer. Notably, Basic Transport Protocol (BTP) [21] and GeoNetworking [22] are non-IP protocols specifically designed for ITS, serving safety applications and time-critical local broadcasts. Additionally, standard protocols such as TCP [23] and UDP [24] over IPv6 [25] are employed for non-time-critical end-to-end communications.

It's worth noting that standard groups selected non-IP solutions for safety applications, even though IP communications could have been equally effective.

2.1.1.4 . Access Layer

The Access Layer encompasses both the Data Link and Physical Layers in the OSI reference model. This layer focuses on the wireless communication aspects, including channels, Quality of Service (QoS), access mechanisms, and

data transmission. Operating on this layer, ITS-G5, defined by ETSI and based on IEEE 802.11p, is initially proposed as the primary access protocol for vehicular networks in Europe, however, ETSI added support to LTE-V2X [26] and is currently working to integrate NR-V2X as well. C-V2X technologies operate on two interfaces: the uplink/downlink interface known as the radio interface between UTRAN and the User Equipment (Uu), and the sidelink interface named Proximity-based Communication 5 (PC5). All the aforementioned protocols and their respective layers are depicted in Figure 2.2.

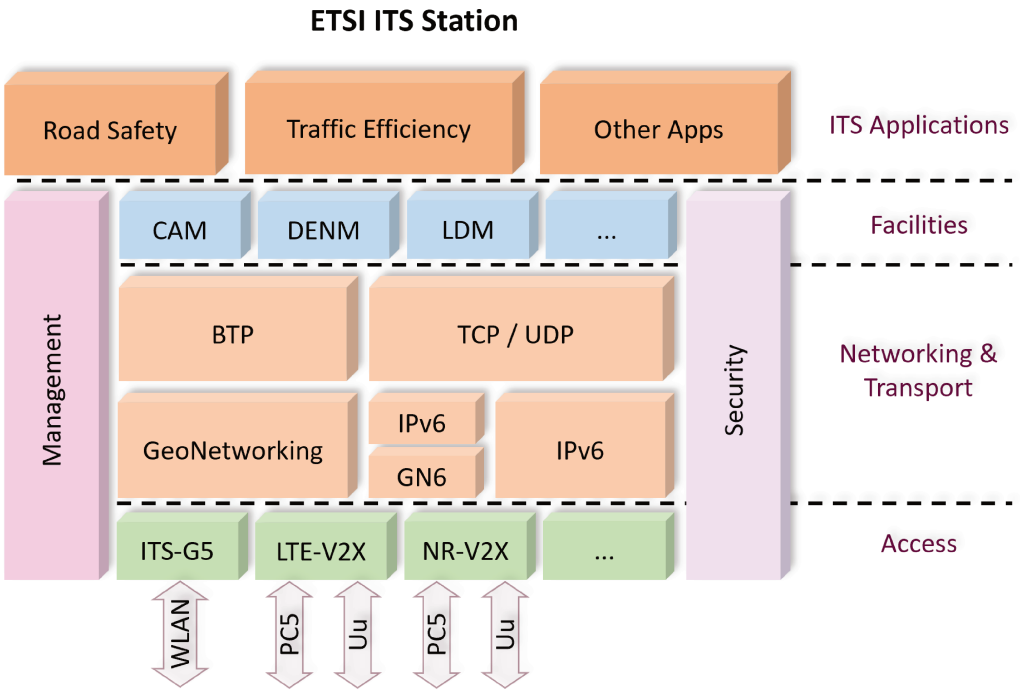

Figure 2.2: ETSI ITS Station Protocol Stack

2.1.1.5 . Management Layer

Maintaining multiple networking, transport, access protocols, and interfaces requires an entity to manage the cross-layer communication profiles, flows, and paths across all the horizontal layers. The Management Layer, the first vertical layer, collects communication requirements from ITS applications, monitors the interfaces' status, and maps the traffic flows to their best available communication path. It also encompasses other functions that are dedicated to regulatory, station, and application management of the ITS-S.

2.1.1.6 . Security Layer

The second vertical layer is the Security Layer. It is in charge of enforcing trust and privacy, through certificates management, encryption keys, and synchronizing the change of pseudonyms and identifiers across the horizontal layers.

For instance, when a vehicle decides to change its pseudonym certificate, the station ID, IP and MAC addresses have to be changed at the same time to eliminate simple correlations and inference attacks.

To standardize secure session establishment, International Organization for Standardization (ISO) created ISO 21177 standard [27], which defines the specifications and procedures for secure ITS stations communications and access control; it is also compatible with both ETSI vehicular certificates [28] and IEEE 1609.2 certificates [29], as well as the Internet Engineering Task Force (IETF) RFC 8902 [30] which adds support of vehicular certificates to Transport Layer Security (TLS) version 1.3 protocol. The services that can be provided by this layer are detailed in Section 2.1.2.4.

2.1.2 . ITS Communications Security Architecture

2.1.2.1 . Vehicular PKI System

As per ETSI specifications, ETSI TS 102 941 [32], a vehicular PKI system shall be in place to secure V2X communications. It provides access control, trust and privacy management, and confidentiality when required. As depicted in Figure 2.3, this system consists of:

1. Enrollment Authority (EA): Its main role is to authenticate the ITS stations and provide them with the necessary permissions for ITS communications. Every ITS station has a unique ID and cryptography keys established during the initialization process. Afterward, the enrollment phase begins, during which the ITS-S authenticates itself with the EA to obtain enrollment credentials. These credentials grant the ITS-S granular permissions for specific ITS applications and services. Only the EA has access to the real identity of an ITS-S.

2. Authorization Authority (AA): Its main role is issuing multiple Authorization Tickets to an ITS station after validating its Enrollment Credentials with the Enrollment Authority. Authorization Tickets are pseudonymous certificates, which are swapped often to keep the real ITS station

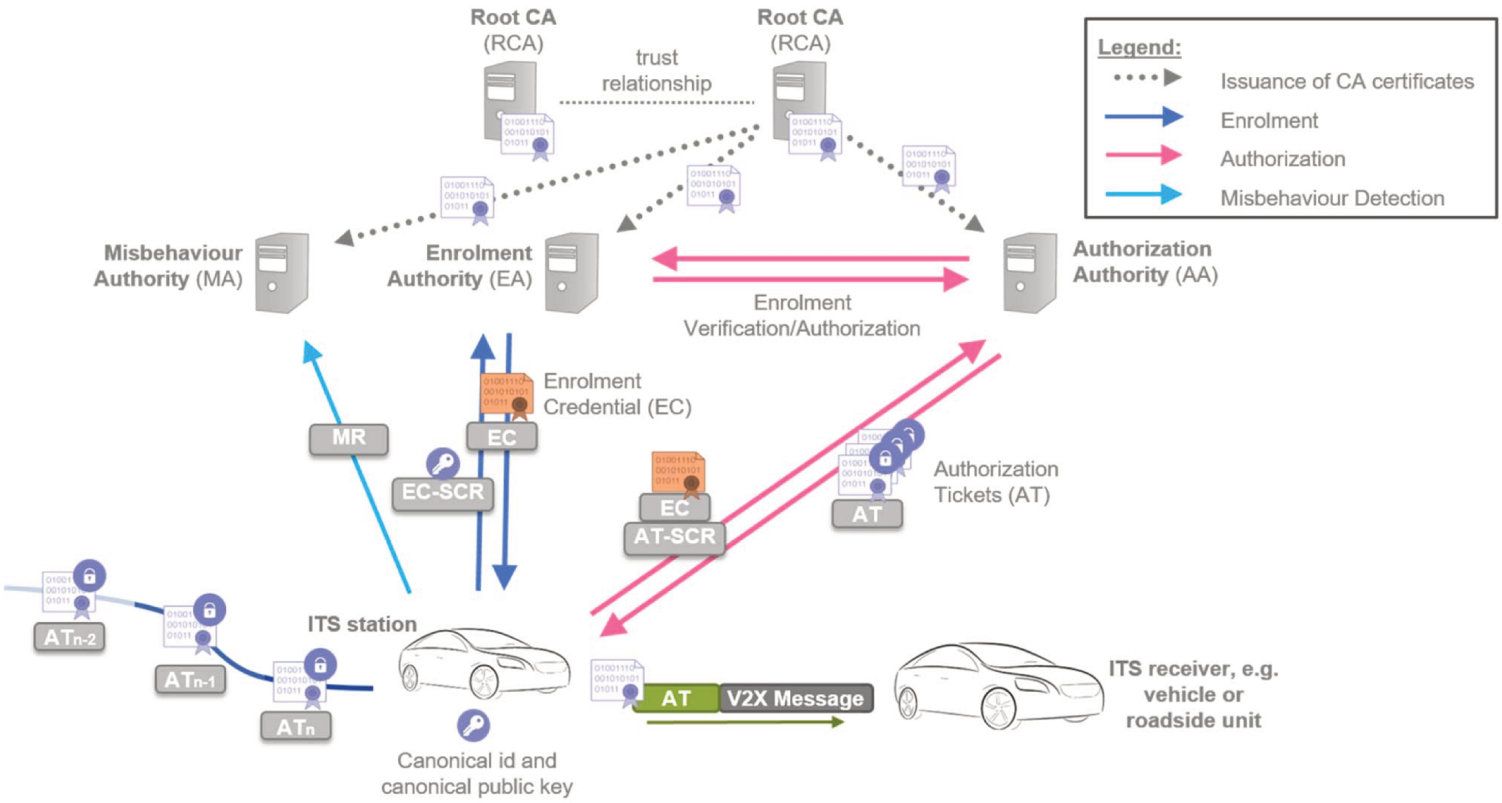

Figure 2.3: ETSI ITS Communications Security Architecture [31]

identity private, and used by the ITS-S to confirm that it has the necessary permissions to access specific ITS services and encrypt/sign secured messages.

3. Misbehavior Authority (MA): Recently added in Release 2 of the standard, its main role is to receive and analyze misbehavior reports sent by ITS stations, and decide whether to revoke the certificate of reported vehicles.

4. Sending, Relaying, and Receiving ITS Stations

2.1.2.2 . Trust and Privacy Management

Trust is enforced through the provisioning of certificates which allows an ITS station to assert their permission to use the ITS system, as well as their permission to use specific ITS services. Privacy is ensured through the use of regularly and frequently changed provisioned pseudonyms. The synchro-

nization of identifier changes including pseudonym certificate, the station ID, network ID and MAC address is enforced to avoid simple correlations.

2.1.2.3 . Confidentiality

Confidentiality is not always required in ITS communications. Most co-operative awareness and hazard warning messages are sent in clear text. However, when a V2X service requires confidentiality, ITS-S shall rely on IEEE 1609.2 [29] mechanisms which utilize vehicular certificates to encrypt the payload before transmission. In C-V2X, starting with 3GPP release 16, confidentiality can also be enabled on the access layer for unicast communications.

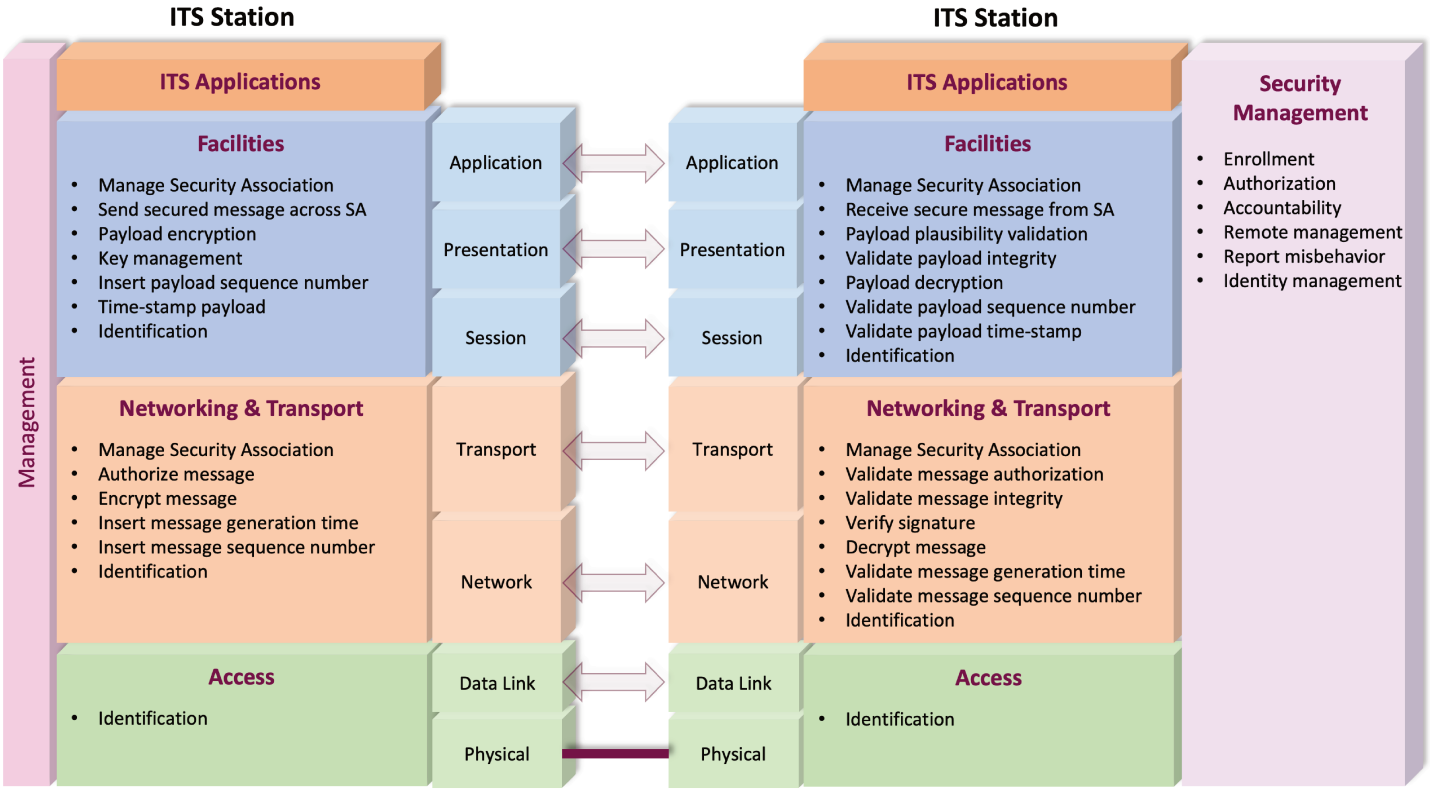

Figure 2.4: ETSI ITS-S Security Services [31]

2.1.2.4 . ITS-S Security Services

The security layer of the ITS protocol stack is responsible for ensuring trust and privacy, through encryption keys and certificates management. It ensures as well the coordination of synchronous pseudonyms and identifiers changes across the horizontal layers.

The security services provided by the Security Layer are spread across all the horizontal layers, their placement is depicted in Figure 2.4, and they can be categorized as follows:

- Enrollment services: manage the enrollment credentials with the EA.
- Authorization services: manage the authorization tickets with the AA.
- Accountability services: record incoming and outgoing messages for accountability purposes.
- Identity management services: supports simultaneous change of communication IDs (pseudonym certificate, station ID, network ID, MAC address).
- Security Association management services: establish secure communications between ITS stations.
- Integrity services: calculate, insert, and validate checksum values.
- Replay protection services: verify the consistency of messages by including and verifying their timestamps and sequence numbers.
- Payload plausibility validation service: determines the reliability of information derived from an incoming communication.
- Reporting service: reports suspicious activities to ITS infrastructure.
- Remote management services: allow ITS infrastructure to remotely manage the transmission capabilities of a misbehaving ITS station.

2.2 . V2X Communications in 3GPP

Local V2X messages in C-V2X are primarily transmitted over the sidelink, also called the PC5 interface, which utilizes short-range direct communication in the 5.9 GHz [33] frequency band. This ensures low latency and high reliability, making it ideal for safety-critical applications like collision avoidance.

Similarly, the Uu interface can be used to transmit V2X messages through the network, which can provide vehicles with a wider range of communication compared to PC5. However, as depicted in Figure 2.5, this requires the support of a network-hosted V2X Application Server, and might introduce delay.

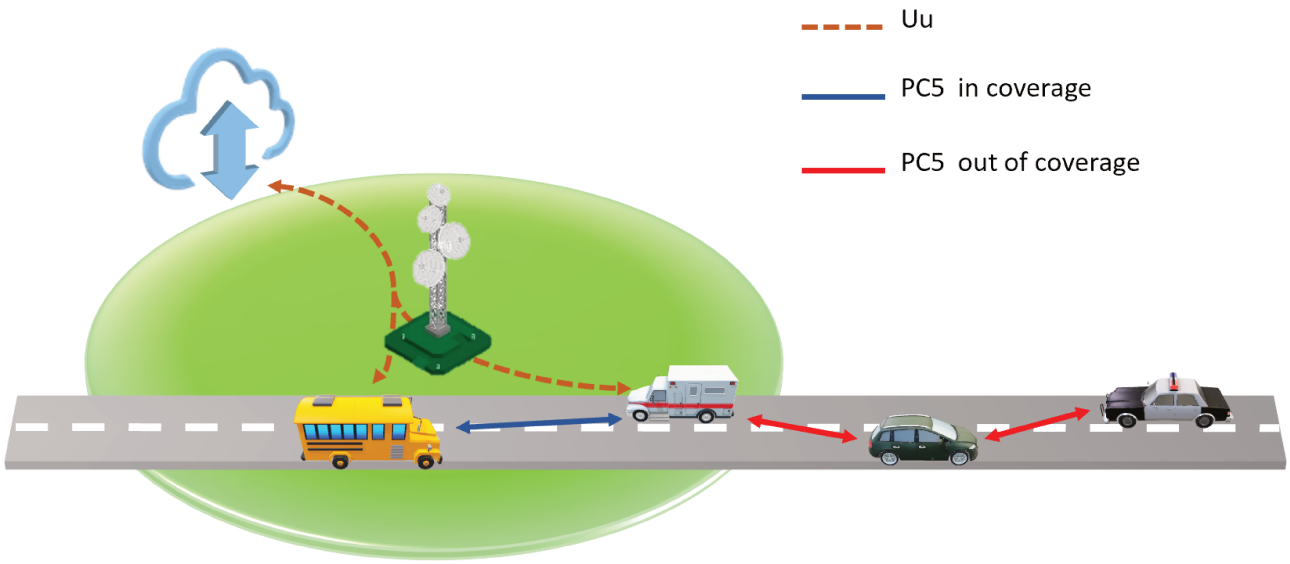

Figure 2.5: C-V2X communications over PC5 and Uu interfaces

C-V2X communications over PC5 have two modes [34]:

- **In coverage**: V2X communications take place when vehicles or V2X-enabled devices are located under the coverage area of a cellular network. This mode of communication relies on the existing cellular base stations which facilitates scheduling and efficient resource allocation, hence improving communication efficiency and minimizing disruptions. As a result, users are provided with a better organized setting that offers enhanced QoS.
- **Out of coverage**: The cellular network can pre-authorize out of coverage communications. Thus, when V2X vehicles are no longer within the coverage area of a cellular network, they can still leverage the sidelink, without depending on any underlying network infrastructure. This mode is crucial in environments when the cellular service is saturated or completely unavailable. In the absence of a centralized network, the responsibility lies with individual devices to autonomously manage and minimize instances of interference.

In summary, in coverage V2X communications offer enhanced services supported by centralized supervision, while out-of-coverage guarantees uninterrupted communication in case of network limitations.

Figure 2.6 illustrates the integration between the C-V2X access layer protocol stack with the protocol stacks of ITS stations. The top layers, as specified by IEEE, ETSI, and ISO, can be effectively re-used in the context of C-V2X. While the access layer protocols are defined by 3GPP to enable cellular V2X communications. 3GPP access layer protocol stack can be summarized as follows:

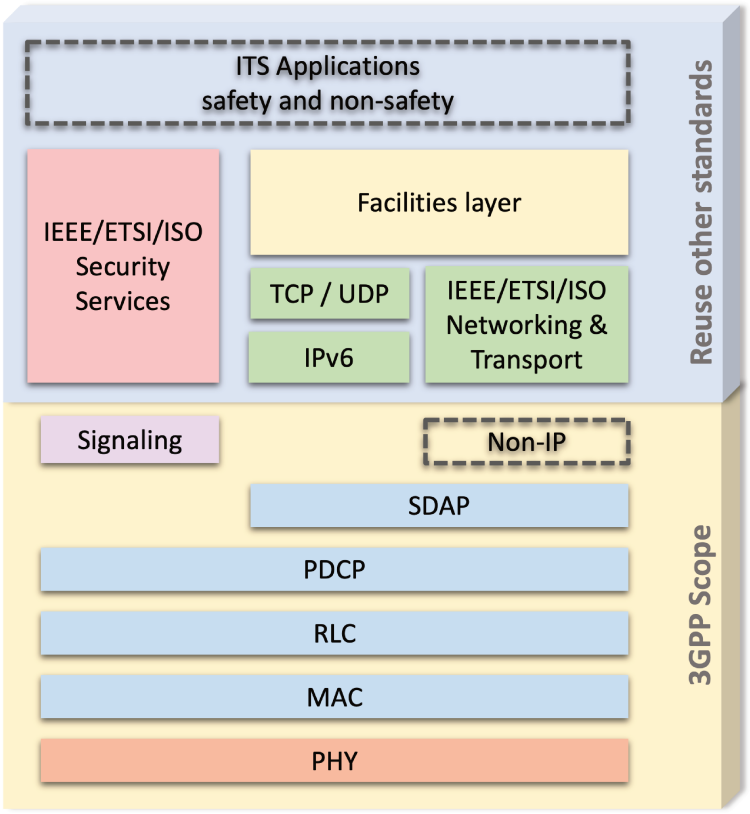

Figure 2.6: Cellular-based V2X ITS Station Protocol Stack

- Service Data Adaptation Protocol (SDAP) [35] new sublayer is introduced in 5G New Radio (NR). As a user-plane, it is responsible for mapping data flows to their corresponding QoS headers.
- Packet Data Convergence Protocol layer (PDCP) [36] sublayer exists in both control-plane and user-plane protocol stacks. It provides functionalities such as header compression, encryption, and integrity.
- Radio Link Control (RLC) [37] sublayer facilitates the processes of segmentation, reassembly, error detection, and recovery mechanisms.

- Medium Access Control (MAC) [38] sublayer is responsible for multiplexing, scheduling information reporting, and error correction.
- Physical layer (PHY), defined in [39][40][41][42][43][44][45], is positioned at the lowest level of the protocol stack. It receives the control information from Radio Resource Control (RRC) to perform coding, data modulation, resource mapping, and antenna mapping for sending or receiving the data on the physical medium.

On the control-plane, signaling protocols may include the following:

- Non-Access Stratum (NAS): It is responsible for the establishment, management, and release of end-to-end connections, including mobility management, security procedures, and user data transmission between the UE and the core network.
- Radio Resource Control (RRC): It is responsible for the configuration, management, and release of radio resources between the UE and the network. This includes processes such as broadcast of system information, connection establishment, handovers, and re-configurations.
- PC5-Signaling (PC5-S): The protocol used for the control plane signaling over the PC5 reference point for the secure layer-2 link.

2.2.1 . V2X Architecture in 3GPP

2.2.1.1 . 3GPP Release 14

3GPP introduced, in Release 14 [46], the first generation of cellular V2X communications based on LTE. A new architecture is defined for V2X and it is integrated as part of Evolved Packet System (EPS) and the existing LTE-based radio communications.

LTE-V2X [47] is based on the development of Device-to-Device (D2D) communications, also called Proximity Services (ProSe) [48], delivered as part of 3GPP Release 12 [49]. The essential functionality of D2D is to enable direct communication between mobile devices through the creation of a new communication interface, the sidelink or PC5, enabling traffic to be directly transferred from a device to another without transiting over a base station. The main use-cases of D2D are:

1. Providing proximity communications during disasters, like earthquakes, where the availability of cellular network infrastructure might be compromised.

2. Replacing legacy public safety communication used by public authorities, like police, ambulance and firefighters.

3. Offloading traffic from cellular network in a crowded environment like a stadium or concert.

As direct communication between vehicles is also required in V2X, D2D and the sidelink fulfill many of the requirements for the basic use cases of V2X communications which are focused on road safety and traffic management.

To improve reliability, the vehicles are pre-authorized by the service provider to utilize the PC5 interface to communicate with each others, even when they are "out of coverage". Which means that the vehicles can still communicate and exchange V2X messages despite the absence of the cellular network and the base stations.

2.2.1.2 . 3GPP Release 15

In 3GPP Release 15 [50], the first phase of 5G [51] [52] [53] is introduced. The new 5G Core (5GC) network and NR radio access technology specifications including Millimeter Wave (mmWave) [54] were developed and published. However, the V2X architecture in this release is still based on LTE (4G).

The work on V2X in this release focused on defining new advanced V2X use cases, coined enhanced Vehicle-to-Everything (eV2X), like platooning, autonomous driving, remote driving, and extended sensors, which have strict latency, reliability and throughput requirements. These requirements cannot be achieved using LTE-V2X. 3GPP Release 15 uses the same V2X EPS architecture introduced in 3GPP Release 14. Also, it utilizes LTE-V2X on the PC5 interface with improvements to radio specifications like 64-Quadrature Amplitude Modulation (QAM), numerology, and link aggregation.

2.2.1.3 . 3GPP Release 16

In order to meet the strict delay, reliability, and throughput requirements of eV2X use cases, 3GPP needed to develop NR-V2X. It is considered the sec-

ond generation C-V2X technology and it was released as part of 3GPP Release 16 [55].

A new V2X architecture based on 5GC is introduced, also, the NR-V2X [56] is defined to be simultaneously used on the sidelink with LTE-V2X. The intention is to use LTE-V2X for the basic road safety use cases, and dedicate NR-V2X to serve the advanced use cases of eV2X, due to its better performance and higher throughput. Also, one of the important features introduced with NR-V2X is the support of unicast, groupcast and broadcast communications, compared to the broadcast-only LTE-V2X. Furthermore, the Uu interface is based on NR instead of LTE providing better uplink and downlink performance for V2N services.

The new 5G V2X architecture brings numerous performance improvements, including:

1. Software-Defined Network (SDN) architecture segregates Control Plane (CP) and User Plane (UP), which can minimize the end-to-end communication delay by bringing the services closer to users.

2. The use of micro-services and Network Function Virtualization (NFV) architectures have the potential benefit of improving performance, reliability, capacity, and availability of the control functions.

3. Network slicing enables the creation dedicated slices for specific V2X services to optimize their performance and isolate them from the remaining services hosted by the telecommunications service provider.

4. 5GC is a Service-Based Architecture (SBA), where all functions can communicate with each other's using 3GPP-defined Application Programming Interface (API) over HTTP/2 protocol.

a) 5G Control-Plane

The 5GC network contains all of the control-plane functions:

- Access and Mobility Management Function (AMF) [57]: manages registration procedures, connection mobility, and User Equipment (UE) authentication. It interacts with the Radio Access Network (RAN) to facilitate the establishment of connections between user devices and the network, hence enabling smooth transitions across various access technologies or geographical areas.

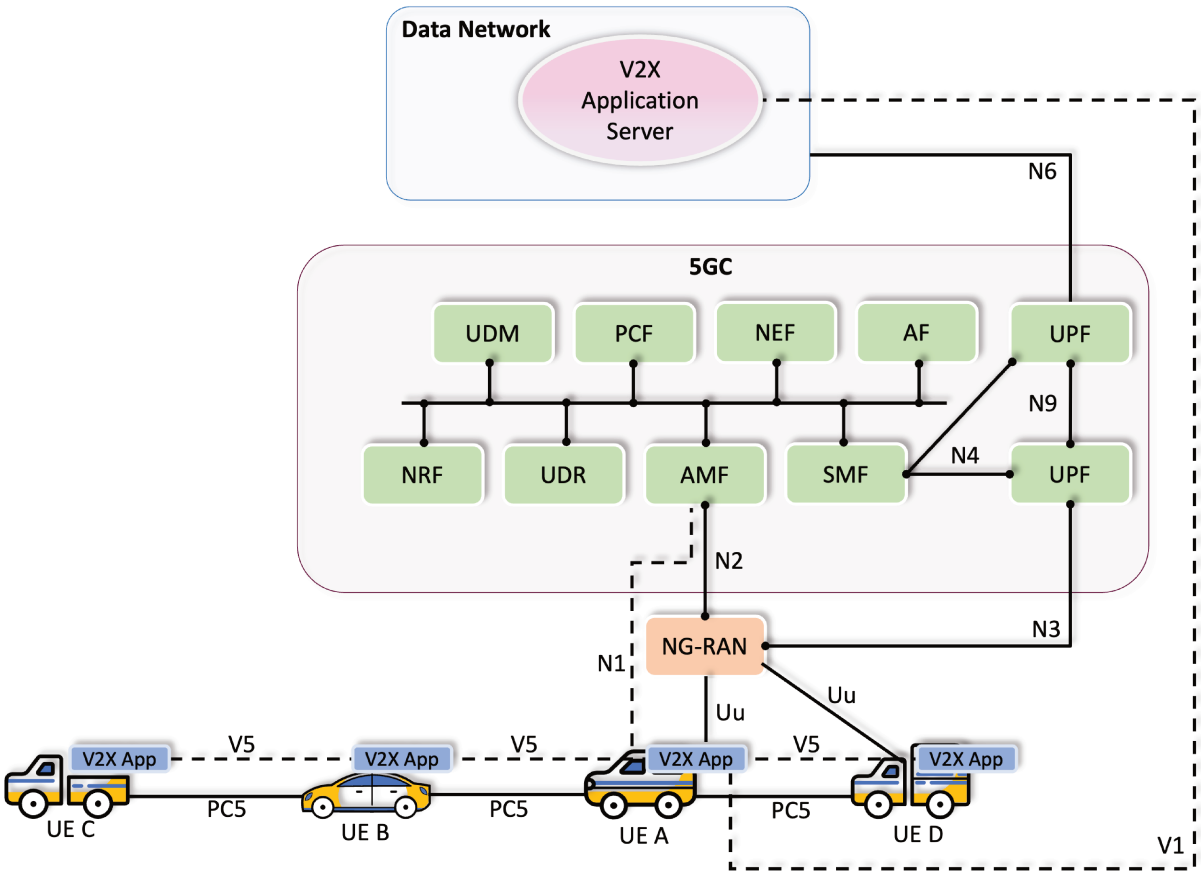

Figure 2.7: 5G System architecture for V2X communication over PC5 and Uu (non-roaming) [56]

- Session Management Function (SMF) [58]: manages user sessions and responsible for IP address allocation, QoS, and routing on the user plane.
- Policy Control Function (PCF) [59]: provides policy rules to control plane functions to ensure that the network resources are used as intended. Decisions about the treatment of sessions and traffic are made by considering subscription information, service requirements, and various other factors.
- Authentication Server Function (AUSF) [60]: responsible for facilitating the authentication process between the UE and the 5G network.
- Unified Data Management (UDM) [61]: manages subscription data and user identities and generates access credentials.

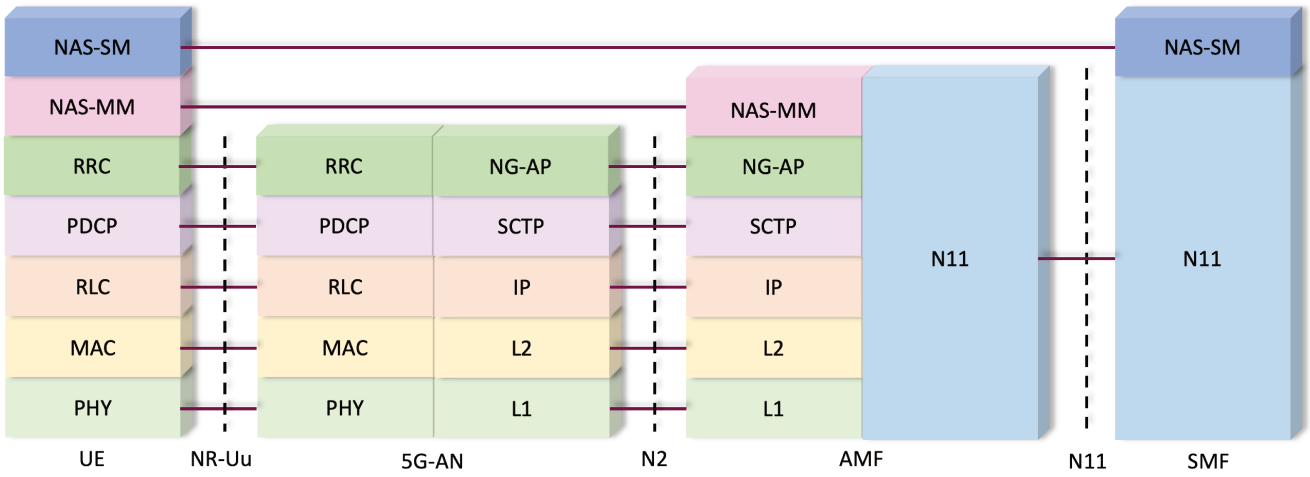

Figure 2.8: 5G Control-Plane Protocol Stack

- Unified Data Repository (UDR) [62]: a centralized storage for structured data, serving other Network Functions like.
- Network Data Analytics Function (NWDAF) [63]: it collects and processes data to assist other NFs in decision-making.
- Network Slice Selection Function (NSSF) [64]: considering the support of network slicing in 5G, which involves the creation of multiple virtual networks and multiple network functions on a shared infrastructure, the NSSF directs user sessions to their relevant slice.
- Network Repository Function (NRF) [65]: handles service discovery between 5GC network function by maintaining information about these services and their instances.
- Location Management Function (LMF) [66]: has the capability to determine the location and velocity of a UE to coordinate and schedule resources.
- Application Function (AF): interacts with the PCF, to influence traffic routing, QoS, and other policies based on the needs of the third-party application.

In the context of V2X services, 5G network functions play crucial roles to ensure its efficient functioning.

The AMF obtains subscription information for V2X from the UDM. It collaborates with the PCF to provide necessary V2X service-related parameters for both the UE and the Next-Generation Radio Access Network (NG-RAN). It also aids in creating or updating user context specifically for the V2X service, hence enabling seamless service delivery.

The UDM manages subscription information for V2X communication, which are stored on the UDR.

The PCF is responsibile for the provision of the V2X policy and parameters to specific UEs based on their PC5 capabilities. It provisions UEs with authorization and policy parameters for V2X communication over PC5 and Uu interfaces. It also supplies the AMF with PC5 QoS parameters for NG-RAN.

The NRF assists in identifying the appropriate PCF that supports the V2X service. This function helps route V2X service-related requests to the correct PCF.

Lastly, V2X Application Server, which is considered an AF, handles both uplink and downlink data between UEs. It can request QoS sustainability analytics from the NWDAF through the Network Exposure Function (NEF) [67][68] for potential QoS changes in specific geographic areas. It's responsible for provisioning the 5GC and UEs with V2X communication parameters for both PC5 and Uu interfaces. The NEF further contributes by allowing the external V2X Application Server to update information related to the V2X service within the 5G network.

Together, these network components create a robust framework to support the V2X service within the 5G ecosystem.

b) 5G User-Plane

The 5G user-plane consists of:

- UE: a mobile phone, vehicle, or V2X infrastructure. The UE performs various tasks, including reporting V2X capabilities and PC5 capabilities to the 5G Core (5GC) over the N1 interface. It can communicate its need for V2X policy provisioning and receive V2X parameters from the 5GC. Additionally, the UE manages procedures for V2X communication over the PC5 interface, configuring parameters related to communication and mapping V2X service types to frequencies. These parameters can be pre-configured or updated through signaling from the PCF or the V2X

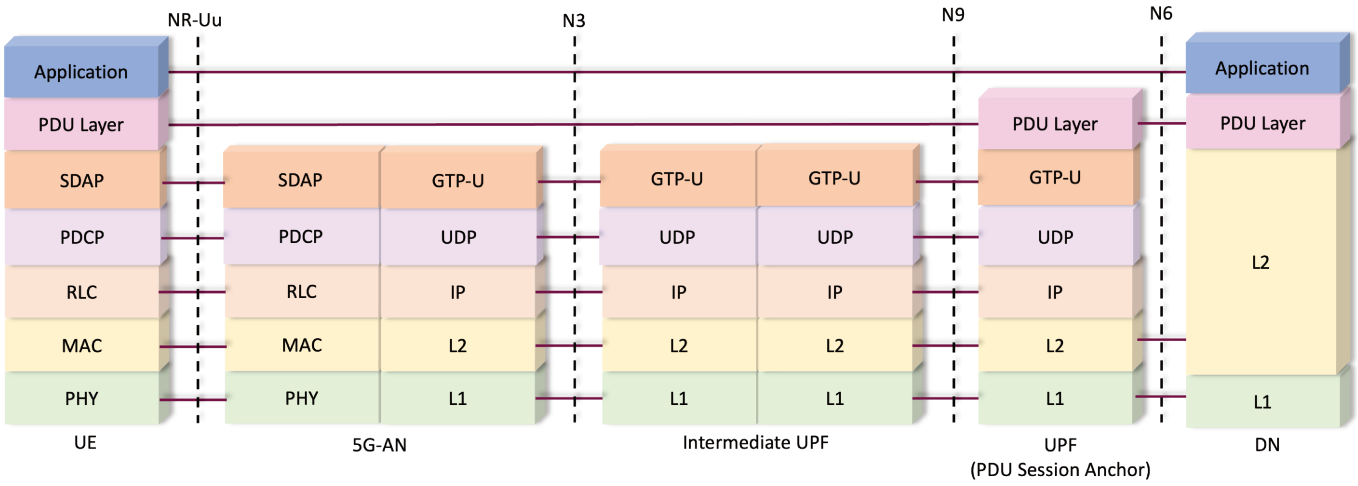

Figure 2.9: 5G User-Plane Protocol Stack

Application Server.

- NG-RAN: gNodeB - 5G base station (gNB)
- User Plane Function (UPF) [69] and Intermediate User Plane Function (I-UPF): controlled by the SMF to ensure reliable transmission of user traffic; it performs tunneling, packet routing and forwarding, QoS, and application detection.
- Data Network (DN): external data network connected to the 5G network. It can be the internet, cloud network, or other third-party service like the V2X Application Server (V2X AS). Local Area Data Network (LADN) is the external edge network, located closer to users to support applications requiring ultra-low latency communications.

Table 2.1 lists the most important reference points that are used in NR-V2X.

2.2.2 . V2X Communications Security in 3GPP

2.2.2.1 . 3GPP Release 14 and Release 15

LTE-V2X Sidelink security (PC5 interface): In LTE-V2X, access-layer level encryption is not supported for V2X communications over the PC5 interface. Instead, V2X side-link communications security relies on application-layer pro-

Reference Point	Position	Protocol
N1	Between UE and AMF	NAS
N2	Between AMF and gNB	Next-Generation Application Protocol (NGAP)
N3	Between gNB and UPF	GPRS Tunnelling Protocol-User plane (GTP-U)
N4	Between SMF and UPF	Packet Forwarding Control Protocol (PFCP)
N5	Between AF and PCF	HTTP/2
N6	Connects UPF to external IP networks	TCP/IP
N9	Between different UPFs	GTP-U
N11	Between AMF and SMF	HTTP/2
V1	Between UE V2X App and V2X AS	-
V5	Between UEs V2X Apps	-
Uu	Between UE and gNB	-
PC5	Between UEs	-

Table 2.1: Summary of 5G System Reference Points

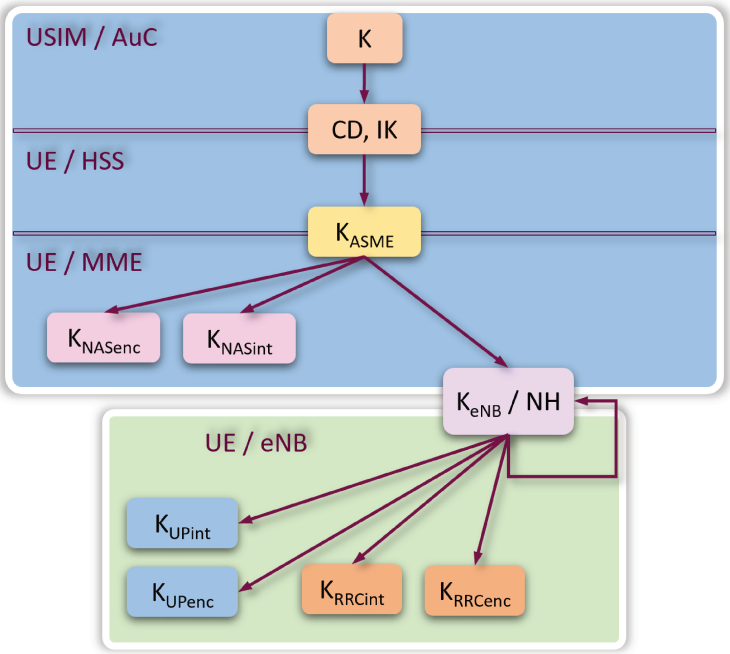

Figure 2.10: Key hierarchy in 4G [70]

tection mechanisms defined in IEEE 1609.2 to provide the optional V2X application-level data confidentiality, when needed, during transmission over the PC5 interface [71].

Uplink/Downlink security (LTE-Uu interface): Confidentiality and integrity protection for V2X communications over LTE-Uu interface can be enabled similarly to the encryption and protection of standard mobile communication. In the EPS, LTE leverages EPS Encryption Algorithm (EEA) and EPS Integrity Algorithm (EIA) to ensure the confidentiality and integrity protection respectively. Both algorithms support the highly secure Advanced Encryption Standard (AES) and Zu Chongzhi (ZUC) algorithms, however, they also support weaker algorithms, like SNOW 3G [72]. Also, in LTE, integrity protection on LTE-Uu interface is only supported for signaling but not for user-plane traffic.

The EPS authentication protocol, Evolved Packet System Authentication and Key Agreement (EPS-AKA), which provides users' authentication, is vulnerable to identity theft attacks, as it occasionally allows sending the user's permanent identifier, International Mobile Subscriber Identity (IMSI), in clear text. The key used to encrypt user traffic on the LTE-Uu interface is K_{UPenc} and the key hierarchy is depicted in Figure 2.10.

2.2.2.2 . 3GPP Release 16

NR-V2X Sidelink security (NR-PC5 interface): Starting with 3GPP release 16, access layer confidentiality and integrity protection are supported on the side-link, enabling optional encryption for V2V, V2I, and V2P unicast communications. However, encryption of groupcast and broadcast communications is not defined in release 16, because these communication types are mostly used for road safety purposes where confidentiality is not required.

The key used to protect local V2X unicast communications is called K_{NRP} *(New Radio PC5)*, and the key hierarchy is depicted in Figure 2.12. The procedures to establish secure unicast direct communication and encrypt user plane data on the access layer between two NR-V2X-enabled vehicles on the PC5 interface are depicted in Figure 2.13.

Uplink/Downlink security (NR-Uu interface): The main advantage of 5G Authentication and Key Management (5G-AKA) over EPS-AKA is that the user's permanent identifier Subscription Permanent Identifier (SUPI) is sent encrypted and never in its clear text form. In 5G System (5GS), cipher algo-

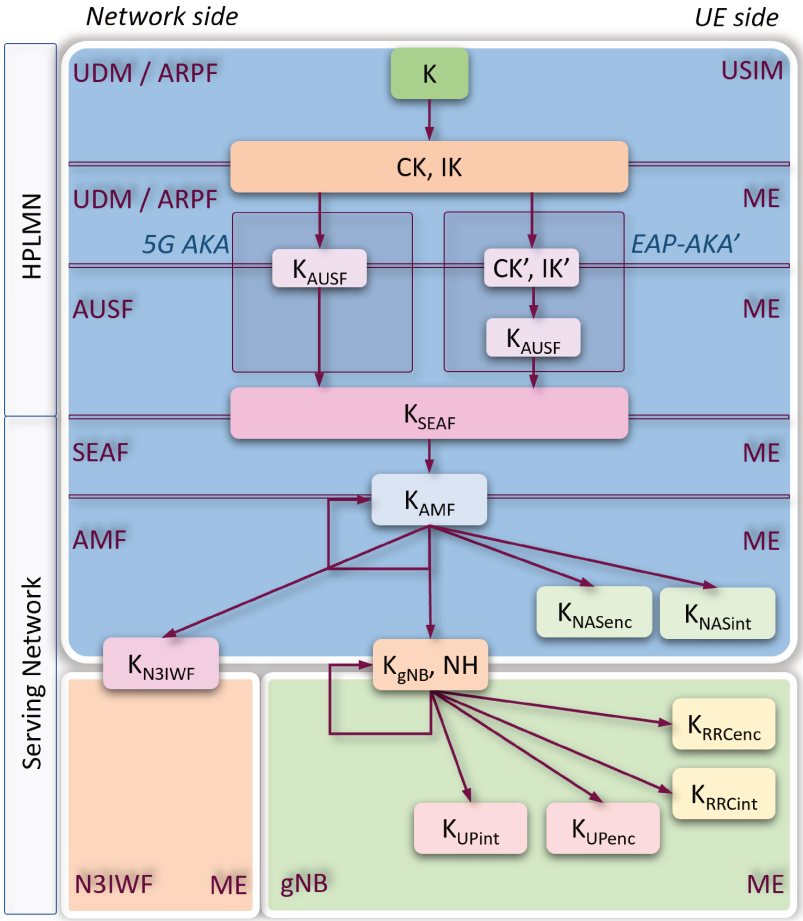

Figure 2.11: Key hierarchy in 5G [73]

rithms NR Encryption Algorithm (NEA) and NR Integrity Algorithm (NIA) are imported from EPS's EEA and EIA without changes, and they are used equally for confidentiality and integrity protection in 5G. Also, in release 16, both signaling traffic and user-plane traffic have both confidentiality and integrity protection fully supported under 5G systems. Two keys are used to encrypt and protect the integrity of user traffic on the NR-Uu interface. They are named K_{UPenc} and K_{UPint} respectively, and their key hierarchy is depicted in Figure 2.11.

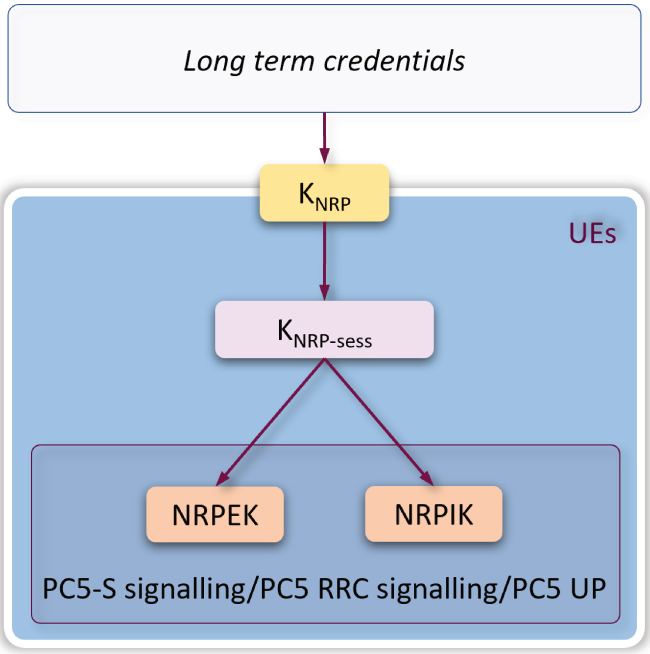

Figure 2.12: Key hierarchy for NR-V2X PC5 unicast link [74]

2.3 . Related Work on V2X Security and Misbehavior Detection

Ghosal et al. [75] provided an overview of the V2X architecture and applications, along with an analysis of security challenges and requirements for both IEEE 802.11p and cellular-based V2X. The authors also proposed a classification scheme for different types of V2X attacks. To mitigate these attacks, the research community has proposed solutions in three major domains: i) symmetric key cryptography, ii) message authentication, and iii) privacy preservation using PKI-based, identity-based, or group-based solutions. The article serves as a important reference for understanding the security challenges and corresponding solutions within the V2X domain.

Lu et al. [76] provide a comprehensive overview of the challenges and strategies to secure V2X services in a 5G network environment. The challenges are categorized into three groups: trust, security, and privacy. The authors also provide a summary of the key research papers addressing these challenges. The proposed solutions are classified based on their location in the 5G architecture: Data Network/Internet, 5G Core Networks, Network Edge, or V2X Communications layer. This classification helps readers to better under-

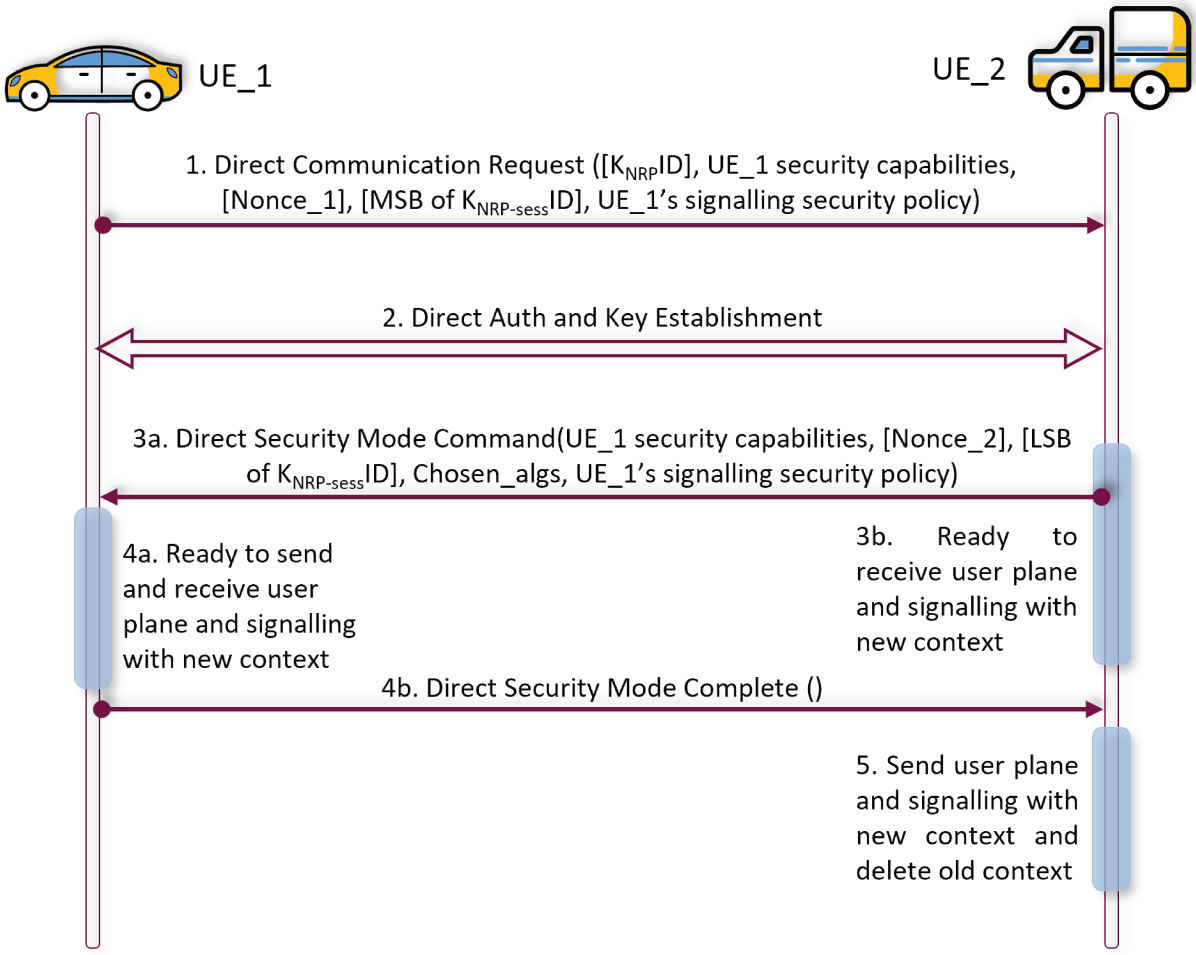

Figure 2.13: Security establishment at connection set-up for NR-V2X PC5 unicast [74]

stand the applicability and effectiveness of the proposed solutions in different parts of the 5G network.

In [77], Sharma et al. provide a detailed review of 3GPP architecture and specifications for 5G V2X security, and they compare it to previous LTE-based V2X. The authors also present security issues and challenges specific to cellular-based V2X, and a comprehensive list of V2X attack types with the affected entities across 5G, 4G and DSRC architectures. Lastly, the authors propose a novel conceptual security function, Security Reflex Function (SRF) which is an edge-based authentication function created to support the rapid changes in

the network and achieve faster authentication/re-authentication for vehicles.

Based on the LuST scenario [78], van der Heijden et al. created VeReMi [79], a publicly accessible dataset of V2X attacks. The dataset was generated using the vehicular network simulators SUMO [80] and Veins [81]. The dataset contains the tracks of both normal and misbehaving vehicles, the latter engage in five distinct position falsification attacks in V2V. The dataset was utilized by the authors to compare various plausibility checks.

In [82], Kamel et al. proposed Framework For Misbehavior Detection (F2MD) which is a framework based on Veins simulator that can be leveraged to develop, test and compare different misbehavior detection algorithms in V2V environment. The prevention mechanism proposed depends on revoking the certificate of misbehaving vehicles.

In [83], So et al. made used of a machine learning-based method for detecting V2V misbehavior. They developed, taking profit of MATLAB, a machine learning version of the VeReMi dataset before proposing a model that makes use of six input features, including two plausibility checks and four numerical measurements of the vehicle's movement. The authors compared the performance of standalone plausibility checks, K-Nearest Neighbors (KNN) and Support Vector Machine (SVM) algorithms. The authors showed that accuracy was improved by 6 to 7 percent, and the precision improved by around 20 percent, while maintaining the recall within 5 percent margin. In [84], Sharma et al. used Scikit-learn [85] to evaluate more machine learning algorithms, including Naïve Bayes (NB), Random Forest, and Ensemble Boosting and Voting. They proved that the Random Forest and Ensemble models outperformed the related strategies.

In [86], Bißmeyer et al. proposed a centralized misbehavior detection system for VANETs. This system receives and analyzes misbehavior reports sent by network nodes upon the detection of an incident. Based on the plausibility of the data received in these reports, the centralized system makes the final decision whether the reported vehicle is considered as behaving correctly or not. The proposed system employs a Bayesian network to calculate the probability of misbehavior based on the reported data, and it is shown to be effective in detecting various types of misbehavior attacks in simulations.

In [87], Gyawali et al. proposed a misbehavior detection model that runs

locally on vehicles, and consists of two main components: a misbehavior detection system for position falsification attacks, and a false alert verification scheme which protects against false alert attacks. They used VeReMi dataset for position falsification, however, they generated their own private dataset to simulate false alert attacks.

In [88], Kim et al. compared KNN, SVM, RF, Extreme Gradiant Boosting (XGB), and Multi-Layer Perceptron (MLP) on the VeReMi-ML dataset. They further introduce a new extended set of differential features that allow the checking of mobility constraints and inconsistencies, which, as per the reported results, proved to improve the machine learning models' performance. When the extended features are used, MLP performs the best, followed by XGB and RF. The authors also proposed Zero-Day Attack Detection based on Auto-encoder architecture. The auto-encoder is a Deep Neural Network (DNN) encompassing an encoder and a decoder with internal hidden layers. The encoder projects the input data to a lower dimensional space, and the decoder restores the lower dimensional representation to the output in the input dimensional space. The auto-encoder is then trained to minimize the reconstruction error while using only normal behavior data. After that, the model can then be leveraged to classify normal and abnormal data based on a pre-defined error rate threshold.

In [89], Lv et al. proposed a misbehavior detection system for VANET using privacy-preserving federated learning and blockchain technology. Their approach allows vehicles to request receiving a shared training model from a RSU, train this model using its local data, and upload the updated model's parameters without sharing local training data with the RSU. The RSU receives models from multiple vehicles and aggregates them based on accuracy scores to produce a new averaged model, which is stored on the blockchain.

In [90], Kamel et al. created an extended version of the VeReMi dataset. This expansion comprises 19 attack types, including a large set of sophisticated V2X attacks like Denial of Service (DoS), Data Replay, and Sybil, at various vehicles densities. The study also uses basic misbehavior detection mechanism based on Long Short-Term Memory (LSTM) and DNN. The results are considered as a baseline for further research. This large dataset can be leveraged to compare and enhance their detection mechanisms and develop new ones.

Alladi et al. proposed in [91] a misbehavior categorization scheme based on deep learning for intrusion detection in V2V. They propose centralized training in the cloud, while the detection nodes are located on Edge servers close to RSUs. They exclude Delayed Messages and Eventual Stop attacks, and study the seventeen remaining attacks in the VeReMi-extension dataset. Single-stage and multi-stage classifiers are proposed with similar performances, and several deep learning model architectures are compared. They consist of several layers of Convolutional Neural Network (CNN) and/or LSTM. The authors found that the models with 2-CNN/1-LSTM layers and 4-LSTM layers performed slightly better than other models.

Sedar et al. proposed in [92] a V2X misbehavior detection system based on LSTM and reinforcement learning. To distinguish between genuine and false data, the proposed model analyzes V2X data from vehicles as time-series at an Edge or Cloud server. Using VeReMi extension dataset, they assessed their approach against various attack types and reported promising results.

2.4 . Conclusion

This chapter delved into the ETSI ITS and 3GPP V2X specifications. It detailed the ITS station protocol stack, ITS communication security architecture, ITS station security services, C-V2X communications interfaces and protocol stack, 5G V2X network architecture in 5G, 5G control-plane and user-plane protocol stacks, and C-V2X communications security.

Next, the chapter presented the state-of-the-art in V2X misbehavior detection techniques proposed by fellow researchers which are focused on V2V. This highlights the the lack of solutions and standardization of misbehavior detection for V2N communications, which we address in the next chapter.

Table 2.2: Summary of recent proposals for V2X misbehavior detection

Paper	Dataset	Approach	Simulator/Tools	Solution Environment
[79]	VeReMi	Local Plausibility Checks	Veins, OMNeT++, SUMO	V2V, DSRC
[83]	VeReMi	Local Machine Learning	MATLAB	V2V
[84]	VeReMi	Local Machine Learning	scikit-learn	V2V
[87]	VeReMi + private dataset for false alerts	Local Machine Learning	scikit-learn, Veins, OMNeT++, SUMO	V2V, DSRC
[88]	VeReMi + modified VeReMi attacks	Local Machine Learning, MLP, and Autoencoder	Not provided	V2V
[89]	VeReMi	Local MLP, Federated Learning on Edge, Blockchain	PyTorch	V2V, V2I, Edge, Backend
[90]	VeReMi Extension	Local Plausibility Checks, Misbehavior Report	F^2MD, Veins, OMNeT++, SUMO	V2V, V2I, DSRC, LTE-V2X, Backend
[82]	VeReMi Extension	Local Plausibility Checks, Machine Learning, MLP, LSTM, Misbehavior Report, and Global Analysis	F^2MD, Veins, OMNeT++, SUMO, scikit-learn, TensorFlow	V2V, V2I, DSRC, LTE-V2X, Backend
[91]	VeReMi Extension excluding two attacks	Edge CNN, LSTM	Not provided	V2V, V2I, Edge
[92]	VeReMi Extension	Centralized Reinforcement Learning	Not provided	V2V, V2I, Edge/Cloud

3 - A NOVEL AI SECURITY APPLICATION FUNCTION OF 5G CORE NETWORK FOR V2X C-ITS FACILITIES LAYER

Contents

3.1 . Introduction

In the near future, vehicles will leverage 5G to communicate with each others and to access C-ITS applications hosted in the cloud. The V2X market is expected to rapidly grow, and the number of C-ITS service providers and services will increase. Most service providers will leverage V2N communications to provide various services to vehicles, like centralized road hazard notifications, traffic efficiency, weather alerts and forecasts, pollution meters, and even entertainment services. Securing these V2N-based services is an additional challenge which has not been deeply addressed yet.

In this chapter, we propose and implement a novel misbehavior detection system, compliant with 3GPP 5G V2X architecture specifications. It uses

machine learning techniques, to detect and prevent position falsification attacks that might occur during V2N communications, protecting, hence, V2X application servers. When an abnormal position is detected, fast and reactive countermeasures against the sending vehicle are taken in collaboration with the 5G core network, to stop the attack.

3.2 . Problem Statement

While a vehicular PKI system is essential to protect against external threats, other approaches should be considered to mitigate attacks launched by malicious insiders, whom are authenticated and already part of the system. The most effective solution is to implement a misbehavior detection system. It monitors and analyzes the data sent by authenticated vehicles and report potential unusual behaviors. Various papers addressed the implementation of V2X misbehavior detection system adapted for V2V communications. However, none of the existing studies have addressed the issue of protecting V2X application servers in a 5G V2N environment from data manipulation attacks initiated by authenticated misbehaving vehicles.

For instance, an ITS service related to weather and pollution will highly rely on the accuracy of the data, measurements, and their respective positions reported by vehicles. A misbehaving vehicle, whether intentionally or unintentionally, sending incorrect positions will contaminate the database used in calculations and forecasting. Consequently, the ITS service provider's business might be negatively impacted. In severe cases, the amount of contaminated data might affect the functionality of the ITS service to an extent where it becomes inaccurate, irrelevant, or even unusable. In such a scenario, the loss of users' trust in the service will occur, and the data manipulation attack might be seen as a form of DoS attack. In addition, it is important to note that these types of attacks have the potential to target safety-related services, potentially resulting in significant consequences.

In this chapter, we address the challenge of detecting malicious vehicles that behave normally in V2V but manipulate positions during V2N communications in a 5G network environment to attack V2X application servers. Note that the position manipulation attacks addressed in this chapter correspond to the five attack types described in VeReMi dataset [79], as depicted in Fig-

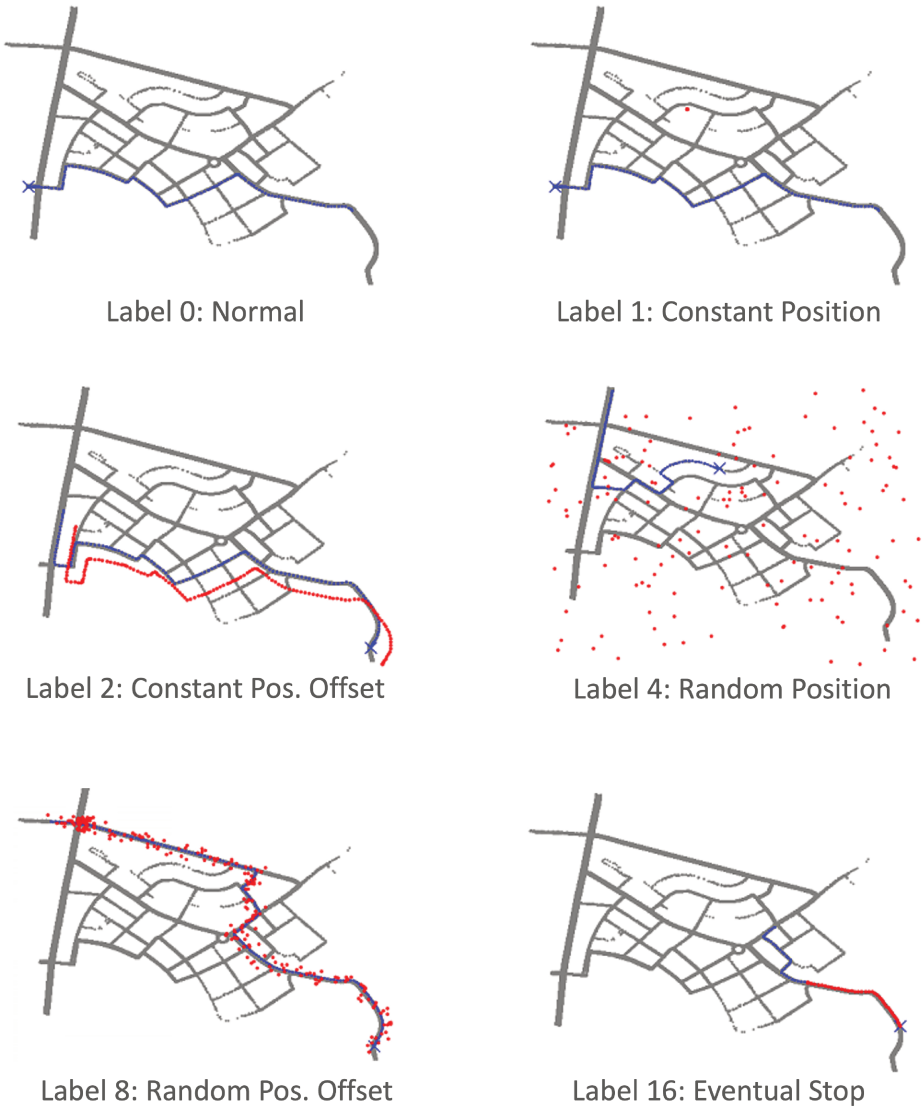

Figure 3.1: VeReMi Dataset Classes

ure 3.1.

- **Attack type 1: constant position**, where the attacker sends the same fixed position despite its movement.
- **Attack type 2: constant position offset**, where the attacker always adds a fixed value to its real position.
- **Attack type 4: random position**, where the attacker generates a new

and random position every time it sends a message.

- **Attack type 8: random position offset**, where the attacker adds a random value to its real position.
- **Attack type 16: eventual stop**, where the attacker starts by sending its accurate position, and after a while, it sends a fixed position similar to attack type 1.

3.3 . Proposed Architecture

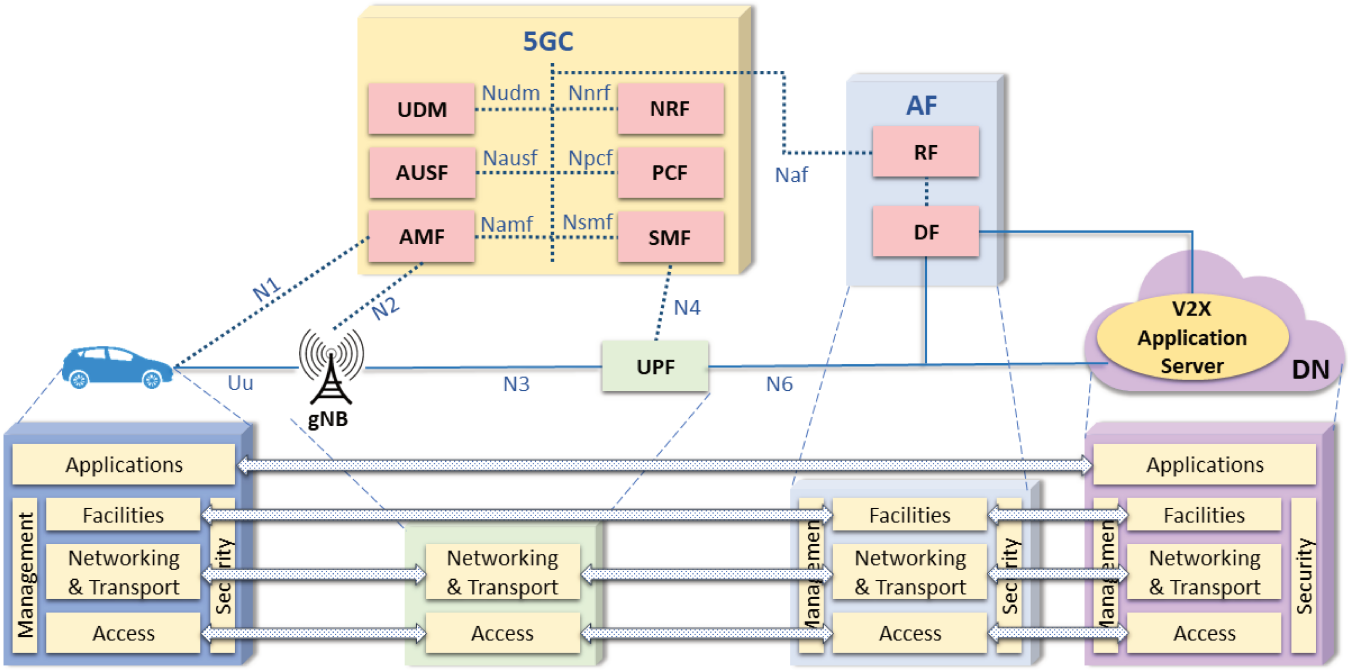

Figure 3.2: Proposed Architecture

To protect V2X application servers from position manipulation attacks and misbehaviors, we propose a V2X Security Application Function (AF) integrated with the 5G core network, in compliance with 3GPP specifications. As depicted in Figure 3.2, our proposed security application function is integrated with the reference architecture of 5G V2X. It consists of two main components:

- Detection Function (DF) which is responsible for real-time analysis and monitoring of V2X traffic packets on the user-plane. Its functionality is detailed in depth in Section 3.4
- Reporting Function (RF) integrated within the 5G core network control-plane to allow telecommunications service providers and legal authorities to revoke the access of reported malicious vehicles and stop the attack.

The data flow in an end-to-end V2N scenario in a 5G environment, where multiple vehicles are sending position information to a V2X application server hosted in the public domain is as follow:

1. The traffic generated by the vehicles is transmitted to the 5G base station using NR radio over the Uu interface.
2. The gNB encapsulates the data using the GPRS Tunnelling Protocol (GTP) protocol, and tunnels it toward the UPF over the N3 interface. To handle internal user-plane routing, telecommunications service providers by adding GTP-U headers to all traffic crossing the N3 interface.
3. Finally, the PDU Session Anchor-UPF (PSA-UPF), which terminates a PDU session and has a direct connection to the public/private domains, removes the GTP-U header before forwarding the traffic to the V2X Application Server in a normal TCP/IP encapsulation over the N6 interface.

To implement our security application, the UPF is configured to forward the V2X traffic specific to the protected V2X service to the DF. The latter will analyze the traffic in real-time, before forwarding it to the V2X AS. When an attack is detected, the DF will notify the RF, which can initiate the countermeasure procedure to stop the attack. Alternatively, the UPF can be configured to duplicate the data it sends to V2X AS and send a copy of it to the DF.

In Figure 3.3, we present a high-level end-to-end workflow of the countermeasure that our security application initiates if an attack is detected. The DF is continuously analyzing V2X traffic as it flows from vehicles toward the V2X AS.

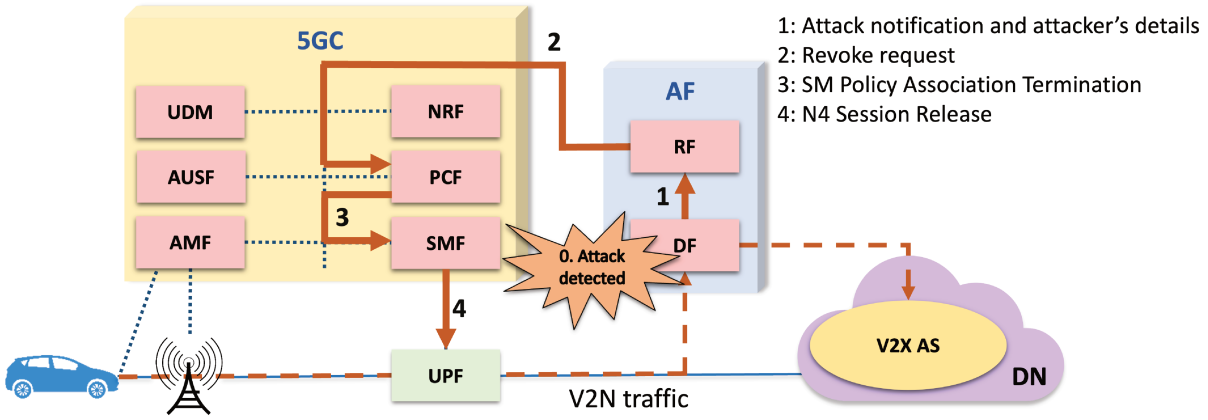

Figure 3.3: Countermeasure Workflow

When an attack is detected:

- Step 1: the DF will notify the RF and provide the necessary details related to the attacker.
- Step 2: the RF changes the policy on the PCF to revoke the attacker's V2X service, which is similar to the service termination procedure when users bypass their spending limit [93].
- Step 3: the PCF may invoke SM Policy Association Termination procedure to trigger a "network requested PDU session release" procedure, as described in [52].
- Step 4: N4 Session Release is the first step taken in the PDU session release procedure.

For sake of simplicity, we omit its remaining steps. After the countermeasure is completed, the attacking vehicle will no more have access to the V2N service that it tried to attack.

3.4 . Proposed AI-based Detection

Misbehavior detection can be achieved through different techniques. In [12], the authors classified these techniques into two categories, node-centric and data-centric. Essentially, node-centric misbehavior detection places a higher importance on evaluating the trustworthiness of the nodes inside the network. In contrast, data-centric misbehavior detection focuses on guaranteeing the integrity and authenticity of the data, which leverage the use of plausibility checks. They are simple and quick verification methods used to determine whether data or values are reasonable and likely to be accurate. They do not perform extensive validation, but rather play the role of initial filters to identify errors or anomalies. In [83], the authors proved the benefit of utilizing Machine Learning combined with plausibility checks to improve misbehavior detection results.

In this section, we will first summarize the general concept of Machine Learning and its different applications, before presenting the details of our proposed detection model.

3.4.1 . Machine Learning

Machine learning is a branch of Artificial Intelligence (AI) that is centered on utilizing data and algorithms to replicate the learning process seen in humans, with the aim of progressively enhancing its accuracy.

Machine Learning has received significant attention and has had a significant impact in several businesses and academic domains. It focuses on the advancement of algorithms to acquire knowledge and make decisions or forecasts by leveraging data, rather than depending on explicit programming. Essentially, instead of being coded with explicit instructions, these algorithms undergo training using extensive datasets, enabling them to independently generate predictions or make classifications when given new data. In light of the growing field of big data, machine learning presents an opportunity for recognizing significant patterns and gaining valuable insights from this extensive data store that is generally difficult for the human brain to detect. Its potential applications span a wide range of fields, including health diagnostics and financial predictions, among many others.

The main machine learning algorithms can be categorized as follows:

- **Supervised learning** is the most common form of machine learning, it is mainly used for **classification** and **regression** applications. It involves the training of algorithms using labeled data, where the desired output is already known. The main objective of the algorithms is to acquire knowledge of a systematic relationship between input data and corresponding output data. Once the algorithms are trained, they possess the capability to make predictions for new inputs that they have not previously encountered. Supervised learning algorithms include a variety of methods, some of which:
  - Linear Regression
  - Logistic Regression
  - Decision Trees and Random Forests
  - Support Vector Machines (SVM)
  - k-Nearest Neighbors (k-NN)
  - Neural Networks
- **Unsupervised learning** applications encompass **clustering**, **dimensionality reduction**, and **anomaly detection**. Unsupervised learning algorithms differ from supervised learning algorithms in that they are designed to handle datasets without explicit labels. However, the primary objective of these algorithms is to identify underlying patterns within the dataset, such as clusters or groups. The algorithms falling under this particular category are:
  - K-Means Clustering
  - Hierarchical Clustering
  - Principal Component Analysis (PCA)
- **Reinforcement learning** is characterized by the agent's iterative engagement with its surrounding environment in order to acquire knowledge and improve performance. The agent is provided with feedback based on its behaviors, which might be in the form of incentives for

correct decisions or penalties for unfavorable actions. The agent's behavior is refined over time via an iterative process that involves taking action and receiving feedback, with the aim of maximizing the cumulative reward. Reinforcement learning encompasses a range of essential algorithms and techniques, including:

- Deep Q-Networks (DQN)
- Policy Gradient Methods
- Monte Carlo Tree Search

The process of training a supervised machine learning model involves utilizing a dataset that contains predetermined input-output pairings in order to train the model on extracting certain relationships between the two. The first step involves the collection and preparation of data, whereby unprocessed data is subjected to cleaning, transformation, and division into separate sets for training and testing. The selection of an appropriate algorithm is thereafter determined according to the specific job at hand, such as classification or regression. During the training process, the model utilizes the training data to generate predictions. The difference between these predictions and the true labels is calculated using a loss function. The internal parameters of the model are modified in order to minimize the loss function, usually using optimization methods such as gradient descent. After training, the model's performance is assessed on the testing dataset. Modifications, including the tuning of hyperparameters, may be conducted in light of this assessment, prior to considering the model suitable for deployment.

3.4.2 . Proposed Model

Many plausibility checks for V2X are proposed in the related work. For instance, Location Plausibility Check (LPC) utilizes i) current vehicle speed and position, ii) average acceleration, and iii) Gaussian distribution for Confidence Intervals (CIs) to predict the next vehicle position on both X and Y directions. If the new position sent by the vehicle falls under the 95% CI on both directions, the score of the new position is set to 0, meaning that the position is plausible. However, if the new position is determined to be outside 95% CI but within 99% CI, then the score is increased by 1 per direction. Lastly, if the new position is outside the 99% CI, the score is increased by 2 per direction.

Therefore, the score range of location plausibility check is between 0 and 4, where 4 means that the position received is unlikely to be plausible.

Movement Plausibility Check (MPC) compares the displacement with velocity. If the calculated displacement is 0 while the average velocity is not 0, the score is set to 1, which means that the vehicle is not reporting a change in position despite its movement. Otherwise, the score is set to 0.

Plausibility checks can be used on their own to detect misbehaviors. However, combining plausibility checks results with quantitative information about the vehicle's movement and behavior, to use them as input features for machine learning can enhance the detection results.

As we are following a data-centric approach, we depend on the data itself and its semantic to perform the detection. In other words, we aim to evaluate if the received data is relevant and thus possible/plausible.

We propose a detection algorithm which enhances the algorithm proposed in [83], by adding a new plausibility check called On-Road Plausbility Check (ORPC), in order to improve detection performance. We recall that in [83] the authors used supervised machine learning with input features LPC, MPC, and the 4 quantitative values to classify the labeled data in VeReMi dataset. Our algorithm falls under the same supervised category. However, we use an unsupervised learning method to compute the score of our proposed plausibility check.

ORPC verifies whether the new received position is on the road or not. To achieve this, it leverages historical location data sent by vehicles which passed by the covered area earlier. Historical location data consist of a set of latitudes and longitudes recorded without any vehicle identification or labels. They designate normal positions where vehicles are expected to be. Therefore, when a new position is received, we can calculate how close it is to the nearest normal position. If it's relatively close, the new position is considered plausible. If not, it is considered implausible.

As the historical data is not labeled, we cannot leverage it using supervised machine learning. Instead, we propose to use an unsupervised machine learning approach, namely, anomaly detection (also called outlier detection). Anomaly detection is the identification of data points that do not conform to the expected pattern of a given group. One of the anomaly detection tech-

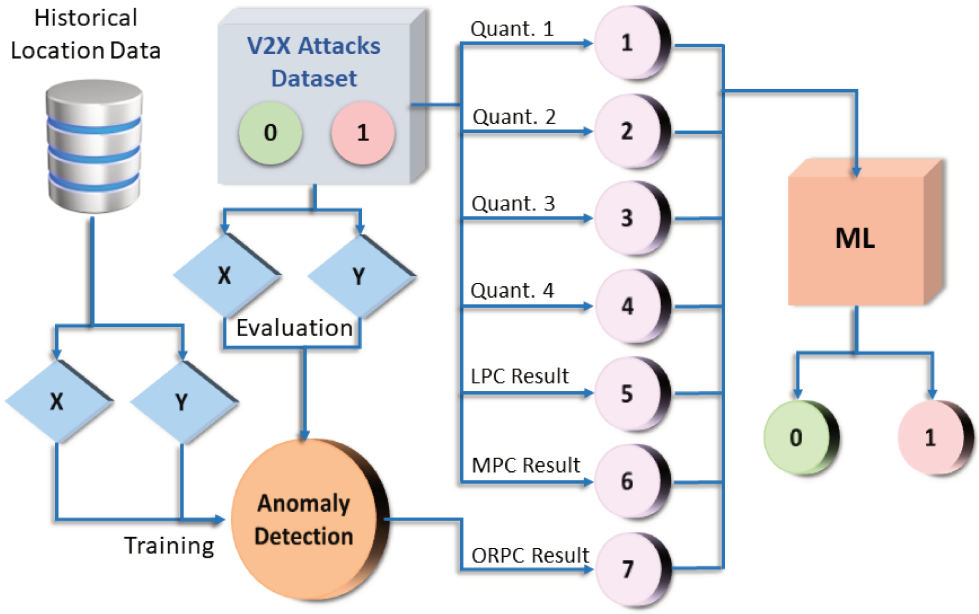

Figure 3.4: Proposed Machine Learning Model

niques consist of leveraging KNN and its functions to perform the evaluation and detection.

We utilize KNN functions in a non-traditional approach in order to perform anomaly detection using the following technique:

1. Fitting the historical data to a KNN model.
2. Visualizing the distances' values between neighboring positions.
3. Estimating a cut-off distance value, if bypassed, the new position will be considered abnormal.
4. Calculating the distance between the vehicle's new position and the nearest normal historical position (nearest neighbor).
5. Comparing the calculated distance to the cut-off value.

If the calculated distance is less than the cut-off value, the new position is considered on-road, and the ORPC result is set to 0. If the distance is beyond the cut-off value, the position is considered off-road and therefore probably implausible and the ORPC result is set to 1.

Our algorithm, depicted in Figure 3.4, uses seven input features. Four features use quantitative information which are the first, second, third, and fourth input features. Three plausibility check results are the fifth, sixth and seventh input features:

1. Difference between calculated average velocity based on displacement and time and the predicted average velocity based on reported velocity and time in the X direction.
2. Same as (1) but in the Y direction.
3. The magnitude of features (1) and (2).
4. Displacement based on calculated distance vs. predicted displacement based on average velocity.
5. Location Plausibility Check (LPC) result.
6. Movement Plausibility Check (MPC) result.
7. On-Road Plausibility Check (ORPC) result.

3.5 . Performance Evaluation

3.5.1 . 5G network environment and dataset

In order to evaluate the performance of our proposed solution, we emulate a 5G Standalone (SA) network, then implement our security application, and finally integrate it with the 5G architecture. We utilize VeReMi-ML dataset to evaluate the proposal.

The dataset is split into two sub-datasets, one for training and one for testing, using four-to-one (4:1) size ratio. Our scheme runs in four phases:

1. Feature extraction phase
2. Training phase
3. Validation phase
4. Testing phase

During the first phase, all features including plausibility checks' scores and quantitative information are calculated. In the second phase, the machine learning model is trained using the calculated values in the previous step as input. In the third phase, a performance baseline is set by using the trained model to evaluate an unused portion of the training dataset. In the fourth phase, the real performance is measured by evaluating the results produced by the trained model using the test dataset that was never used during previous phases.

For 5G emulation, we use EstiNet [94] to build our evaluation environment. EstiNet added a 5G version to its simulator which includes UE and gNB simulation, as well as 5G core emulation imported from free5GC [95]. EstiNet includes all the major functions of 5GC CP such as AMF, SMF, PCF, AUSF, NSSF, UDM, UDR, and NRF. Each function runs on its own docker instance, making it compliant with the microservices architecture. EstiNet also includes the UP components. First, the UE and RAN simulation, running on the same node, both registering with the core network. Second, the emulated UPF function, controlled by the SMF, connects the UEs to the DN where the V2X application server is hosted. EstiNet also supports integrating custom-built applications through docker images, which is essential to integrate our proposed solution.

Using Estinet, we built a topology corresponding to the proposed architecture in Figure 3.2. Our security application has two components, the misbehavior detection function (DF) and the reporting function (RF). We developed the detection function using python and scikit-learn [85], while the reporting function was developed in C. Then, we transformed our application into a docker image, which runs as an Application Function (AF) connected to the 5G core network.

As Free5GC does not implement all 3GPP procedures, we could not implement the countermeasure workflow as planned. Instead, we use a workaround to disconnect the attacker's PDU session using the *upCnxState* deactivation procedure described in [58]. While this workaround should not be used in actual 5G implementations, it produces a similar effect of disconnecting the PDU sessions of reported vehicles during the simulation.

We use the recorded traces of both normal and false positions in VeReMi-ML dataset to simulate traffic sent by vehicles to the V2X application server.

As the dataset was originally created for VANET and V2V scenarios, we had to perform some modifications to fit our V2N environment. The two main customization are:

- Arrange by sender: The original dataset is arranged based on sender/receiver pair. The first modification we made is to merge all the messages of the dataset, arrange them based on sender ID and sending time, and then filter and remove duplicate messages. In doing so, we create lists of all the messages sent by each sender across the map. We also assume full 5G coverage area and message visibility.

- Last message assessment: The last two messages received from the same sending vehicle are directly evaluated against the machine learning model. Note that in the original algorithm, the evaluation is vehicle-based. It means that it occurs only when the receiving vehicle stops receiving messages from the sending vehicle. This change helps making real-time predictions while limiting the number of cached positions to only one, preserving the vehicle's path privacy.

3.5.2 . Performance metrics

The performance of the models is determined by the number of *True Positive (TP)*, *True Negative (TN)*, *False Positive (FP)*, and *False Negative (FN)* produced by the models. Which can also be represented using the four formulas below:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{3.1}$$

Accuracy is the ratio of correctly classified instances to the total number of instances.

$$Precision = \frac{TP}{TP + FP} \tag{3.2}$$

Precision indicates the ratio of correctly predicted attacks to the overall detection predicted by the model.

$$Recall = \frac{TP}{TP + FN} \tag{3.3}$$

Recall, also called detection rate, represents the ratio of correctly predicted attacks to the overall actual attacks.

We recall that a high precision implies a small number of false positives. And high recall suggests fewer false negatives, which also translates to an increased detection rate. Consequently, a performing model is characterized by high precision and high recall.

$$F1 - score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{3.4}$$

The F1-score is not a distinct metric; rather, it combines precision and recall into a single value.

3.5.3 . Evaluation and Results

We evaluate our proposed solution on two stages:

- Offline analysis: using Python and Scikit-learn to numerically evaluate the performance of different ML classification models directly on the dataset. During the offline analysis stage, we compare the performances of multiple machine learning algorithm on the dataset, including: Logistic Regression, Decision Tree, Random Forest, SVM, KNN, Naïve Bayes, Ensemble: Voting, Bagging, Boosting, and Stacking, and finally Neural Network.

- Real-time scenario: using the 5G network emulator running in a simulated scenario to analyze the vehicles' V2N traffic in real-time while it is transmitted over the 5G network.

3.5.3.1 . Offline Analysis

In the offline evaluation, the dataset is split into two main sets. An 80 percent dedicated for training and cross-validation, and 20 percent assigned as the test set. We run 5-fold cross-validation, where the training set is split into five sub-sets, four for training and one for validation. The process runs five times alternating the selection of the validation set between the sub-sets. With each run, a different sub-set is considered as a validation set and the rest are used for training. The results of training and cross-validation assist us fine-tuning the models', and establishing a baseline for the expected performance during the test phase. After tuning the machine learning models, they are

trained on all of the 80 percent training/cross-validation set (all five sub-sets). Then, it is used to evaluate the test set which forms 20 percent of the original dataset.

Table 3.1 compares the performances of several ML algorithms, with and without using ORPC, per attack type and when all of the attacks are combined.

Six different machine learning models, per algorithm, are trained using the training dataset. Five models trained uniquely with one specific type of attacks. And the sixth model is trained using all the five types of attacks. The first five models are used for analysis purposes only, because in a real-world scenario, the model needs to be trained on all of the attacks, like the sixth model. After the models training using the training dataset, the models are used to locally evaluate (offline) the messages recorded in the test dataset. The results are generated by comparing each message prediction to its respective label in the dataset, and calculating the metrics described in the previous section.

Analyzing the results in Table 3.1, starting with attack type 1, the Random Forest model achieves the highest precision, while KNN outperforms it in recall and the overall F1-score.

Concerning attack type 2 results, all of the algorithms performed poorly before adding ORPC. This is due to a high bias problem, where the input features are not enough to predict the correct classification of this kind of attacks. Therefore, after adding ORPC as an additional input feature, the detection results are considerably enhanced. All of the algorithms perform well with a slight edge to SVM in precision, Logistic Regression, Ensemble Stacking and Neural Network in recall and F1-scores.

The nature of attacks type 4 and type 8 includes randomization, therefore, they are considered as the easiest attacks to detect. All the machine learning algorithms were able to detect all of the attacks.

Due to the nature of attack type 16, the labels used in the dataset for messages are not consistent; they do not reflect the actual state of the sent message, instead, the label 16 is assigned to all messages sent by an attacking vehicle whether the vehicle is performing the attack or behaving normally. Therefore, the detection performance on attack type 16 seems to be the lowest. Under these circumstances, the best performing algorithms on precision

are Random Forest and Ensemble Bagging. Naïve Bayes scores slightly better than the rest on recall, and on the overall F1-score, which also shows a good performance from SVM.

In summary, we notice similar performances between most of the machine learning algorithms, with very slight advantage to Random Forest and Ensemble learning algorithms. Also, adding ORPC shows it effectiveness especially in detection of attack type 2, which affects the overall performance, improving the precision by 3 percent, the recall by 15 percent, and F1-score by 11 percent.

Table 3.1: Offline Results: ORPC Impact on Performance

Metric	Algorithm	Attack 1		Attack 2		Attack 4		Attack 8		Attack 16		All Attacks	
		with ORPC	*without ORPC*	*with ORPC*	*without ORPC*	*with ORPC*	*without ORPC*	*with ORPC*	*without ORPC*	*with ORPC*	*without ORPC*	*with ORPC*	*without ORPC*
Precision	*Logistic Reg.*	0.9897	0.9897	0.9898	0.0000	1.0000	1.0000	1.0000	0.9990	0.6595	0.6436	0.9388	0.8995
	Decision Tree	0.9943	0.9955	0.9730	0.2723	1.0000	1.0000	1.0000	1.0000	0.5296	0.5381	0.9252	0.8343
	Random Forest	**0.9977**	**0.9977**	0.9898	0.2624	1.0000	1.0000	1.0000	1.0000	0.7074	**0.7066**	**0.9609**	0.9111
	KNN	0.9760	0.9741	0.9730	0.2912	1.0000	1.0000	1.0000	1.0000	0.5440	0.5416	0.9304	0.8450
	SVM	0.9888	0.9876	**0.9906**	0.0000	1.0000	1.0000	1.0000	1.0000	0.6585	0.6575	0.9383	0.9134
	Naive Bayes	0.8500	0.9233	0.9897	0.0000	1.0000	0.9980	0.9892	0.9863	0.6474	0.6001	0.9108	0.8800
	Ens. Voting	0.9943	0.9955	0.9888	0.2820	1.0000	1.0000	1.0000	1.0000	0.5296	0.6492	0.9507	0.8776
	Ens. Bagging	0.9955	0.9955	0.9889	0.2878	1.0000	1.0000	1.0000	1.0000	**0.7080**	0.7032	0.9596	0.9084
	Ens. Boosting	0.9966	0.9966	0.9888	**0.5000**	1.0000	1.0000	1.0000	1.0000	0.6552	0.6529	0.9348	**0.9270**
	Ens. Stacking	0.9943	0.9955	0.9898	0.0000	1.0000	1.0000	1.0000	1.0000	0.6606	0.6584	0.95202	0.9111
	Neural Network	0.9888	0.9899	0.9898	0.0000	1.0000	1.0000	1.0000	1.0000	0.6608	0.6590	0.9376	0.9205
Recall	*Logistic Reg.*	0.7962	0.7962	**0.9656**	0.0000	1.0000	1.0000	1.0000	0.9960	0.6309	0.6294	0.8758	0.7066
	Decision Tree	0.7935	0.7953	0.9466	0.2009	1.0000	1.0000	1.0000	1.0000	0.4901	0.5107	0.8440	0.7012
	Random Forest	0.7953	0.7944	0.9647	0.0480	1.0000	1.0000	1.0000	1.0000	0.5152	0.5205	0.8623	0.6923
	KNN	**0.8116**	**0.8161**	0.9457	**0.2271**	1.0000	1.0000	0.9990	0.9990	0.5411	0.5403	0.8687	**0.7375**
	SVM	0.7962	0.7962	0.9538	0.0000	1.0000	1.0000	1.0000	0.9960	0.6370	0.6370	0.8758	0.7015
	Naive Bayes	0.7962	0.7962	0.9611	0.0000	1.0000	1.0000	1.0000	1.0000	**0.6507**	**0.6613**	**0.8909**	0.7093
	Ens. Voting	0.7917	0.7935	0.9629	0.0878	1.0000	1.0000	1.0000	1.0000	0.5548	0.5563	0.8603	0.7021
	Ens. Bagging	0.7944	0.7944	0.9638	0.0534	1.0000	1.0000	1.0000	1.0000	0.5297	0.5320	0.8618	0.6935
	Ens. Boosting	0.7944	0.7944	0.9629	0.0009	1.0000	1.0000	1.0000	1.0000	0.6218	0.6126	0.8820	0.6980
	Ens. Stacking	0.7935	0.7953	**0.9656**	0.0000	1.0000	1.0000	1.0000	1.0000	0.6309	0.6294	0.8611	0.7072
	Neural Network	0.7962	0.7962	**0.9656**	0.0000	1.0000	1.0000	1.0000	0.9990	0.6317	0.6355	0.8763	0.7010
F1-score	*Logistic Reg.*	0.8825	0.8825	**0.9776**	0.0000	1.0000	1.0000	1.0000	0.9975	0.6449	0.6364	0.9062	0.7915
	Decision Tree	0.8826	0.8842	0.9596	0.2313	1.0000	1.0000	1.0000	1.0000	0.5091	0.5240	0.8827	0.7620
	Random Forest	0.8851	0.8845	0.9771	0.0811	1.0000	1.0000	1.0000	1.0000	0.5962	0.5995	**0.9089**	0.7868
	KNN	**0.8863**	**0.8881**	0.9592	**0.2552**	1.0000	1.0000	0.9995	0.9995	0.5425	0.5410	0.8985	0.7876
	SVM	0.8821	0.8816	0.9719	0.0000	1.0000	1.0000	1.0000	0.9980	0.6476	**0.6471**	0.9060	0.7936
	Naive Bayes	0.8223	0.8551	0.9752	0.0000	1.0000	0.9990	0.9946	0.9931	**0.6491**	0.6293	0.9007	0.7855
	Ens. Voting	0.8815	0.8831	0.9757	0.1339	1.0000	1.0000	1.0000	1.0000	0.5419	0.5992	0.9033	0.7801
	Ens. Bagging	0.8836	0.8836	0.9763	0.0901	1.0000	1.0000	1.0000	1.0000	0.6060	0.6057	0.9081	0.7865
	Ens. Boosting	0.8841	0.8841	0.9757	0.0018	1.0000	1.0000	1.0000	1.0000	0.6380	0.6321	0.9077	**0.7964**
	Ens. Stacking	0.8826	0.8842	**0.9776**	0.0000	1.0000	1.0000	1.0000	1.0000	0.6454	0.6436	0.9043	0.7963
	Neural Network	0.8821	0.8825	**0.9776**	0.0000	1.0000	1.0000	1.0000	0.9995	0.6459	0.6470	0.9059	0.7959

3.5.3.2 . Real-Time Evaluation

After the offline analysis, we evaluate the online scenario using the 5G emulator. We define a custom scenario with 171 simulated V2X vehicles, 32 benign and 139 misbehaving, sending more than 7500 messages as V2N traffic which are distributed as shown in Table 3.2. As Random Forest model performed related well during the offline stage, we use it to analyze the live traffic during the online scenario.

The online results are collected in two different configurations of the security application:

- Detection mode: the attackers are neither reported nor disconnected, and the objective is to expose the security application to the maximum amount of messages and record the detection performance on the message-level.
- Prevention mode: in order to protect the V2X application server, the security application will report and request the disconnection of a detected attacker once it reaches a pre-defined attack threshold, and the results are recorded in terms of number of disconnected vehicles.

Also, we run each mode three times, using three models:

1. Without ORPC
2. With ORPC
3. With ORPC while excluding attack type 16 from the training

As depicted in Figure 3.5, simulated vehicles are assigned vehicle IDs from the test set. They register with the 5G network before they start sending their V2N messages with the exact timing, frequency, and position recorded in the dataset. When the message reaches the security application, the vehicle ID is checked to determine if it is previously known. If the vehicle ID is new, time and position information included in the first message are cached. When the second message arrives from the same vehicle ID, the previous message is retrieved from the cache and combined with the new message. The two messages are processed to create the plausibility checks scores and calculate the quantitative values. Then, they are submitted as input to the ML classification model. The model returns 0 if it predicts a normal position, and returns 1 if

Table 3.2: Online Scenario: Number of Vehicles and Messages

Vehicle Type	Vehicles count	Messages count	Prediction count
Normal	32	1483	1451
Attack 1	35	1308	1273
Attack 2	30	1249	1219
Attack 4	29	1369	1340
Attack 8	22	790	768
Attack 16	23	1470	1447
Total	171	7669	7498

it estimates an abnormal/manipulated position. If an attack is predicted, the attack counter of that vehicle ID is incremented. In prevention mode, when the attack counter reaches a previously set reporting threshold, the vehicle ID and its related information are reported to the core network to revoke the attacking vehicle's connection to the V2N service. The reporting threshold helps tuning the sensitivity of the security application.

3.5.3.2.1 Detection Mode

The results of six simulations are depicted in Figure 3.6. The three tables on the left represent the results of binary classification. While the three confusion matrices on the right are used for analysis purposes only. They are obtained using multi-class classification models.

In models (i), where ORPC is not used, there a is high number of false negatives. Around 25 percent of the messages labeled as malicious are predicted as normal, the recall score is 0.7528. The multi-class confusion matrix in (i) reveals that around half of attacks type 2 and type 16 messages are misclassified. The misclassification of attack type 2 is due to the lack of features as discussed earlier in the offline results section, which we address by adding ORPC. Concerning attack type 16, its lower performance is due to the inconsistency of labels discussed earlier. Also, we notice the model's confusion in classification of attack types 1 and 16, due to the similarities between the two

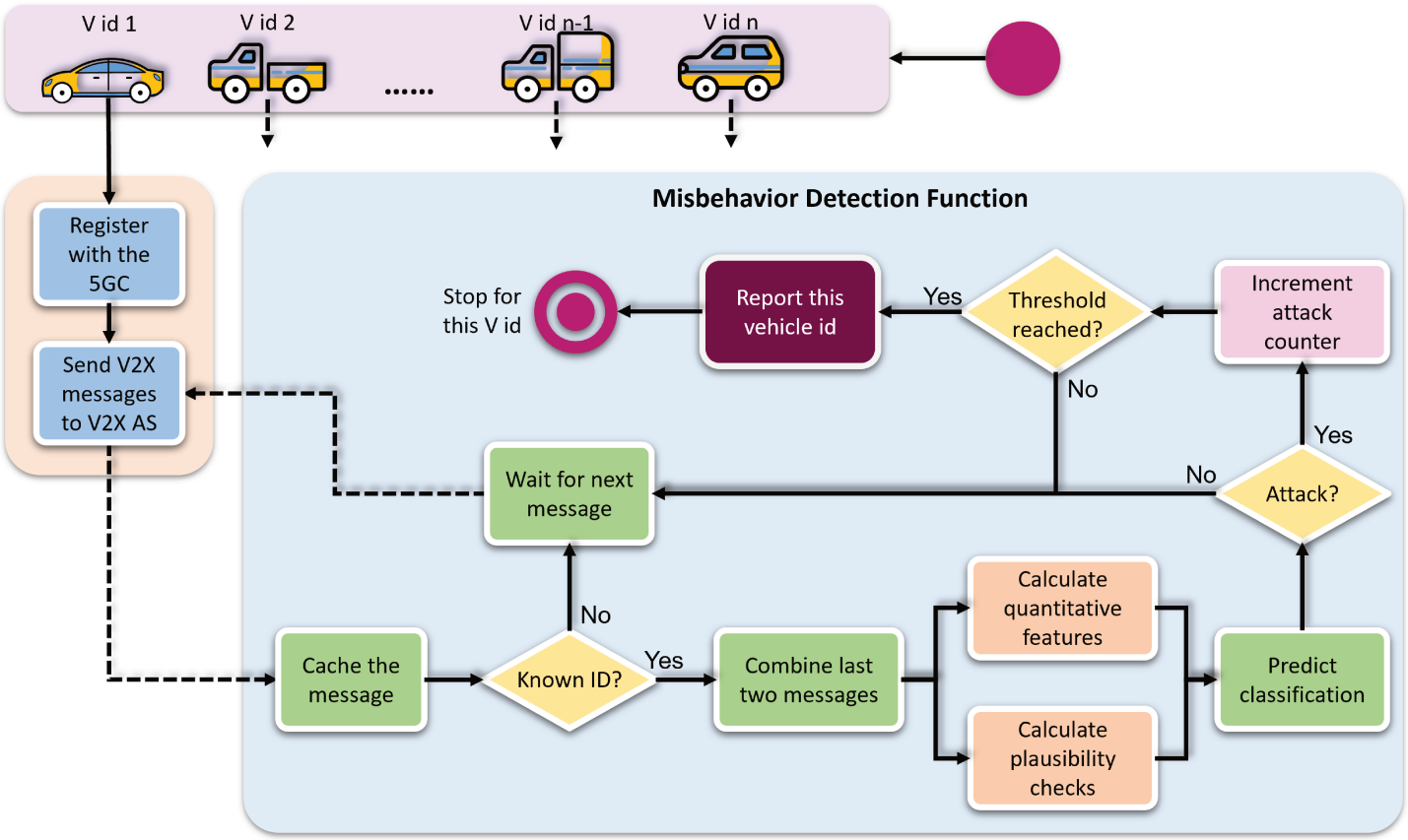

Figure 3.5: The workflow of the detection function

attacks. They actually perform the same attack type, the only difference between the two is that vehicles using attack type 1 are not always attacking.

After adding ORPC as an input feature, we notice the impact on the overall detection in (ii), where recall increased from 0.7528 to 0.8444. The main factor of this increase is the dramatic improvement in attack type 2 detection, which is highlighted in the second confusion matrix.

Due to the similarity between attack type 1 and attack type 16, we exclude the attack type 16 from the machine learning training phase in (iii). We notice less false negatives than (ii), also, the recall score increases from 0.8444 to 0.8632. We notice that most attack type 16 messages are now classified as type 1 in the third confusion matrix.

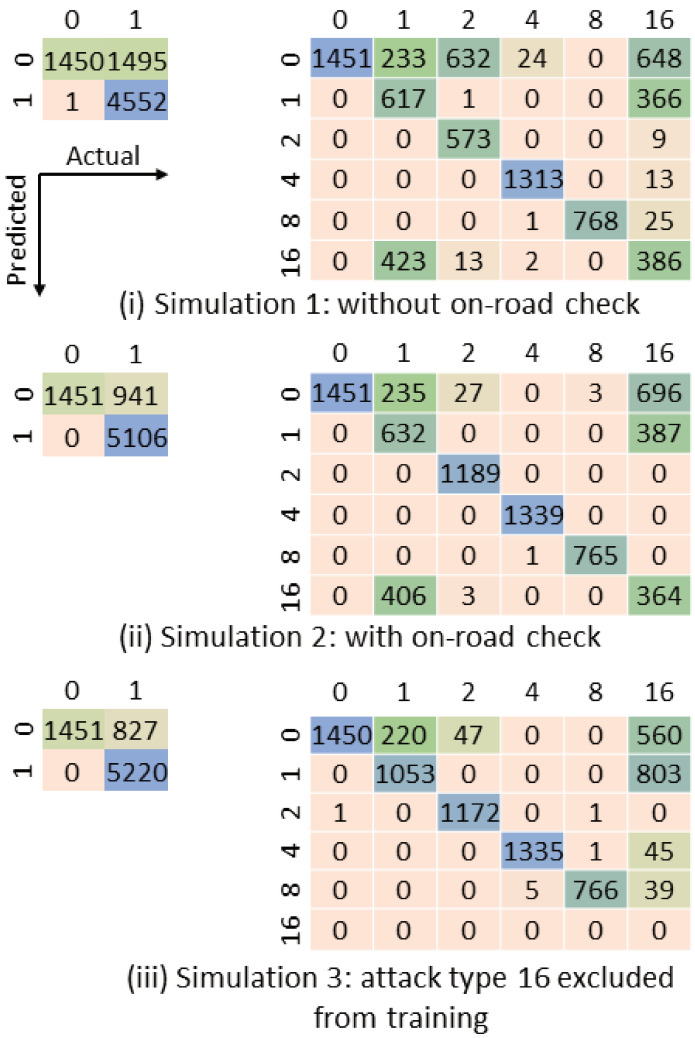

Figure 3.6: Online Results: Predictions and Confusion Matrices

3.5.3.2.2 Prevention Mode

In prevention mode, we also compare the performance of the three previous models. However, the attacking vehicles are reported and disconnected after they reach a reporting threshold. The performance is based on the number of reported vehicles instead of the number of detected messages.

Models (i) and (iii) misclassified one normal vehicle, producing a single false positive detection when the reporting threshold is aggressively set to 1.

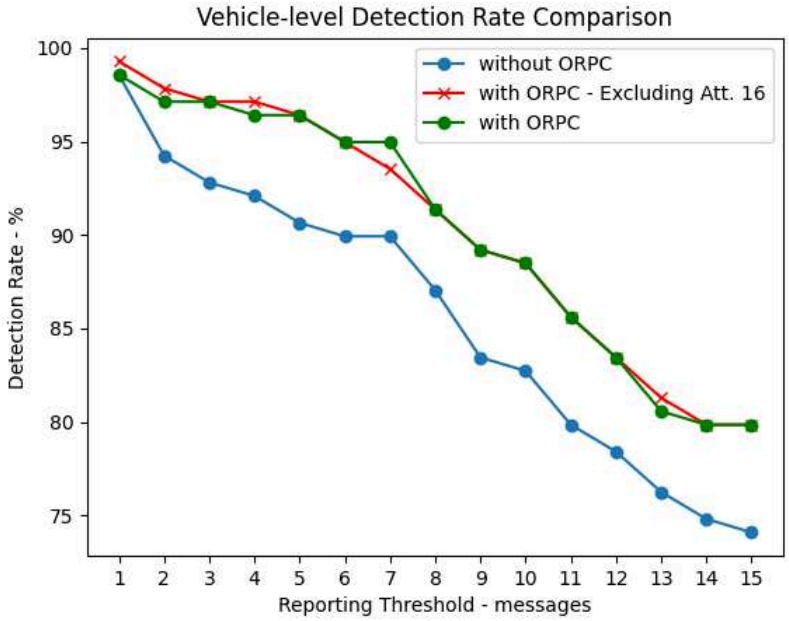

Figure 3.7: Online Results: Vehicle-level Detection Rate Comparison

However, when reporting threshold is increased to 2 and beyond, both models do not produce any false positives, achieving a perfect precision. Also, model (ii) did not produce any false positives at all reporting thresholds.

As there is no remarkable difference in precision between the models, we compare their recall performance only at different reporting thresholds. As depicted in Figure 3.7, using the lowest reporting threshold value of 1, models (i) and (ii) scored a detection rate of 98.56 percent, while model (iii) scores slightly better at 99.28 percent.

The importance of using ORPC is demonstrated starting with reporting threshold value of 2, where both models that uses the additional plausibility check (models ii and iii) achieve approximately 5 percent higher than the model that does not use it (model i). In conclusion, using ORPC not only improves the message-level detection but also the vehicle-level detection rate, especially when the detection system's threshold and sensitivity level are tuned, hence, improving the system's flexibility and reliability.

3.6 . Conclusion

In this chapter, we propose a framework to protect V2X application servers in 5G networks. Our proposed solution integrates a V2X misbehavior detection system, as an application function, to the 5G core network. In order to detect position manipulation attacks, we propose a detection mechanism based on AI, which leverages historical data to calculate an on-road plausibility check. The message-level results demonstrate a considerable improvement in recall and a slight enhancement in precision. Also, in vehicle-level detection results, our algorithm scored 5 percent higher regarding the detection rate.

4 - 5G V2X MISBEHAVIOR DETECTION AS EDGE CORE NETWORK FUNCTION BASED ON AI/ML

Contents

4.1 . Introduction

Within the complex framework of V2X communications, the integrity and correctness of transmitted data emerge as important concerns, especially in the field of 5G networks. While the previous chapter delved into the foundational aspects of integrating a V2X misbehavior detection for V2N communications within the 5G system architecture, this chapter focuses on bringing the solution to the edge network to improve scalability and meet the low-latency requirements of some V2X applications while exploiting the potential of a collaborative approach between edge detection nodes.

The rationale behind a collaborative approach comes from the concept of leveraging the feedback of a previous edge node on a specific vehicle to improve the likelihood of a correct classification made by the current node.

In this chapter, we present our novel V2X edge misbehavior detection system that utilizes machine learning techniques in compliance with 3GPP 5G V2X architecture specifications. Our proposed system aims to detect and prevent position falsification attacks that may occur during V2N edge communications, thus ensuring the authenticity of data received by V2X application

servers. The misbehavior detection application instances are implemented in the edge network, where they are interconnected and can collaborate to improve detection accuracy. We evaluate the performance of our system to demonstrate its effectiveness in detecting attacks.

4.2 . Problem Statement

Traditional V2V misbehavior detection techniques might eliminate malicious nodes on V2V and V2N when these nodes are misbehaving on both PC5 and Uu interfaces. However, it is possible for a vehicle to behave normally on V2V while acting maliciously on V2N. This could potentially result in V2V misbehavior detection systems failing to detect and mitigate all attacks, thereby necessitating the implementation of additional solution.

As 5G V2X networks enable the deployment of low-latency V2N services in the edge network, it is important to ensure that misbehaving vehicles cannot compromise the edge-hosted services. While standalone detection nodes within an edge network can detect and respond to anomalies, their isolated perspectives might limit their detection capabilities. This limitation becomes even more evident when considering sophisticated attacks that may exploit the nature of decentralized V2X communications.

Throughout this chapter, our main objective is to address the following crucial question: How can a collaborative approach in V2X misbehavior detection within the 5G edge network enhance the overall security of the system? While individual nodes possess localized data and insights, collaboration can potentially harness an enhanced view of the network, leading to more accurate and timely detection of misbehaviors.

Note that, in this chapter, we address the five types of position manipulation attacks that were identified in the VeReMi dataset [79], which are the same attack types addressed in Chapter 3, and depicted in Figure 3.1. We propose a V2X misbehavior detection system for the 5G edge network which utilizes two advanced machine learning models to improve detection accuracy.

4.3 . 5G Edge Misbehavior Detection Architecture

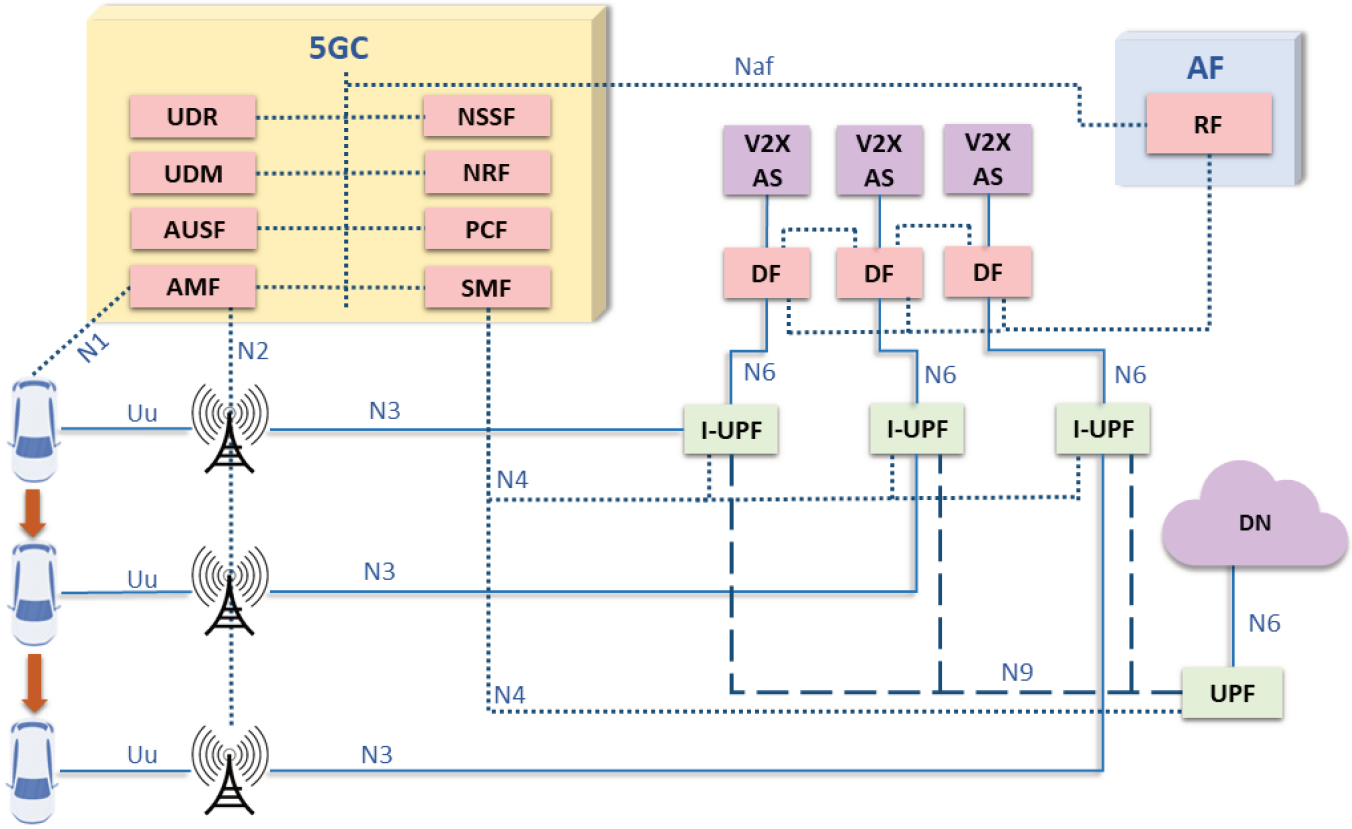

Figure 4.1: Proposed 5G Edge Misbehavior Detection Architecture

V2X Application Servers can gather information from vehicles and also provide them a variety of C-ITS services. V2X ASs can be hosted in the edge network, in the Cloud, or on the Internet. Regardless of their deployment option, V2X ASs need to be protected from position manipulation attacks and other misbehaviors.

To achieve this, a misbehavior detection system needs to be in place. In terms of deployment, it can be whether centralized or distributed with many instances. The deployment option of the misbehavior detection system depends on the architecture of the C-ITS application it is protecting. If the C-ITS application requires low latency, it might be deployed in the edge network, where instances of the application are created on the LADN near to the local

I-UPF. For each C-ITS application instance, a misbehavior detection function (DF) instance is created to protect the application as shown in Figure 4.1.

Our proposed misbehavior detection system is considered as an extension of the proposed system in the previous chapter, and it consists of two main components:

- Several instances of interconnected Detection Functions (DFs): they perform real-time analysis and monitoring of V2X traffic packets on the UP. Each DF instance will analyze the V2X traffic flowing between vehicles and local V2X AS. DF instances can communicate and exchange detection information related to vehicles moving between their respective coverage areas.

- A single instance of Reporting Function (RF): integrated with the 5G core network control-plane to enable telecommunications operators or legal authorities to revoke access of the reported malicious vehicles and stop the attack. It acts as an Application Function (AF), which controls the application's traffic flows by interacting with the 5G core network using 3GPP standard API.

4.4 . Collaborative Proposal: AI-based Detection

We propose two machine learning models: Standalone and Collaborative, which we will refer to as S.A. and Collab., respectively. The S.A. model is the same model used in Chapter 3 which includes seven features, while the Collab. model provides an improvement over the previous model by adding an eighth feature. To predict if the vehicle's behavior is benign or malicious, both models utilize the six input features proposed in [83]. We recall that these feature are the following.

1. The difference between calculated average velocity based on displacement and time and the predicted average velocity based on reported velocity and time in the X direction.

2. Same as feature (1) but in the Y direction.

3. The magnitude of features (1) and (2).

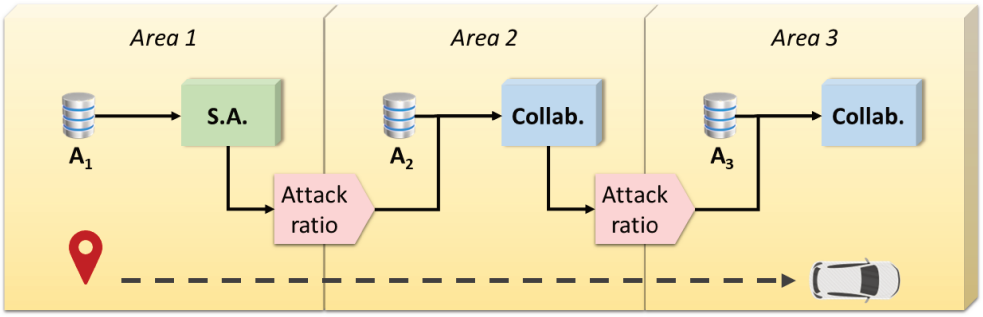

Figure 4.2: Proposed Machine Learning Models

4. The difference between displacement (based on the calculated distance) and predicted displacement (based on the average velocity).
5. Location Plausibility Check (LPC) result.
6. Movement Plausibility Check (MPC) result.
7. On-Road Plausibility Check (ORPC), used in both S.A. and Collab.
8. Attack Ratio from previous area, used in Collab. only.

The first four features consist of quantitative movement values. The following two features are the results of plausibility checks proposed in related work. Lastly, the last two features include one that we introduced in the previous chapter and a new additional feature, the details of which are provided below.

The new feature we propose in this chapter is the "Attack Ratio" of a vehicle. It represents the ratio of the number of attack messages predicted by the detection model in the previous area to the total number of messages sent by the vehicle while passing through that area. It is shared between the detection nodes upon the transition of the vehicle from an area to another. It can be seen as a reputation value of the vehicle, which is exchanged between different areas. This idea is similar to the machine learning concept proposed in [96].

We recall that the LPC, MPC, and ORPC plausibility check are detailed in the previous chapter, section 3.4.2.

An Example is depicted in Figure 4.2, where we can notice that when a new vehicle enters area 1 and it's not previously known by any other neighbor area, the S.A. model is selected to analyze this vehicle's traffic because the previous attack ratio cannot be determined. Once this vehicle moves to area 2, the attack ratio calculated in area 1 will be transmitted and used as an input to the Collab. model instance of area 2. When this car moves from area 2 to area 3, the new attack ratio predicted by the Collab. instance in area 2 will serve as an input feature to the next Collab. instance of area 3, and so on.

4.5 . Performance Evaluation

To evaluate the performance and efficiency of our proposed scheme, we utilize two evaluation approaches:

- Offline analysis which numerically evaluates the VeReMi-ML dataset using Python and Scikit-learn [85], to compare the initial performance of the standalone and collaborative models.
- Real-time scenario using free5GC [95] 5G network emulator, UERANSIM [97], and generating new attacks in real-time, by utilizing a subset of VeReMi-ML dataset to create new traffic sent by vehicles to the V2X application servers.

We leverage Accuracy, Precision, Recall, and F1-score metrics to evaluate the models. The formulas of these metrics are detailed in Section 3.5.2.

Table 4.1: Offline Results: ORPC

ML Model	*Accuracy*	*Precision*	*Recall*	*F1-score*
w/o ORPC	0.8926	0.9131	0.6921	0.7874
w/ ORPC	0.9504	0.9609	0.8626	0.9091

In the previous chapter, we established that RF algorithm is one of the top-performing algorithms on the dataset. Therefore, we employ it for all the evaluation models in this chapter.

4.5.1 . Offline analysis procedures and results

Before presenting the offline results of the comparison between Standalone and Collaborative models, we will explain the process of training and testing both models, which is depicted in Figure 4.3. The offline evaluation dataset is divided into three main subsets, which are built based on vehicle ID. The first subset *X* is reserved to S.A. training, the second subset *Y* is dedicated to Collab. training, while the third subset *Z* is used to test both models and compare their performances.

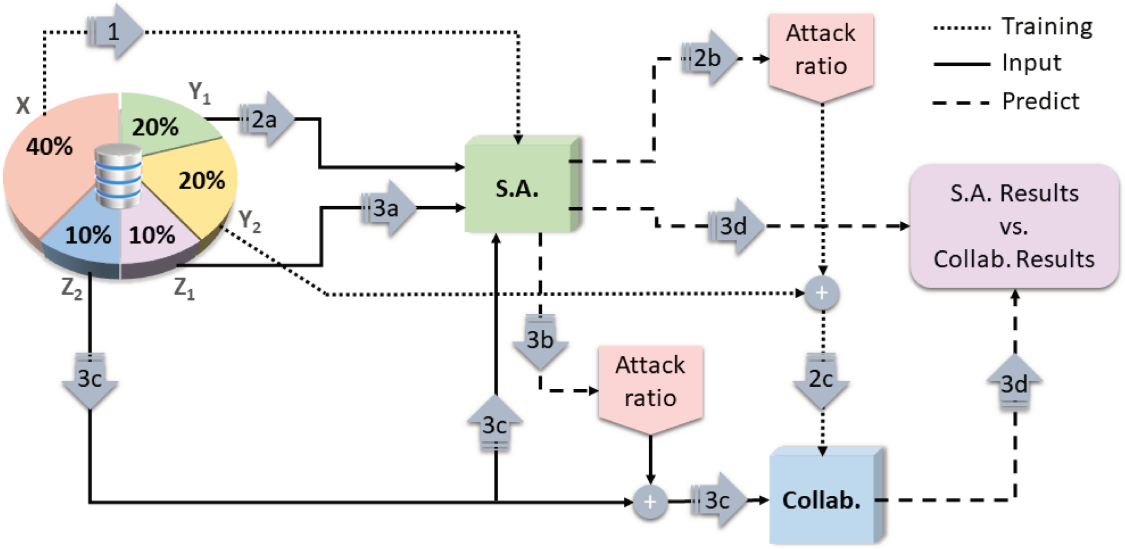

Figure 4.3: Offline Training and Evaluation Process

- **Step 1 - Training S.A. model:** The vehicles' sent messages, recorded in subset *X*, are processed to produce the first seven input features, which are then utilized to train the S.A. model.
- **Step 2a - Input training set to S.A.:** After S.A. model is trained, the second step is to train the Collab. model; subset *Y* is used for this purpose. It is divided into two additional subsets: Y_1 and Y_2. Every vehicle in subset *Y* has its messages split equally between Y_1 and Y_2. Subset Y_1 represents messages of a specific vehicle sent in the first area, and Y_2 represents the remaining messages of that vehicle, assumed to be sent under the coverage of the second area.

- **Step 2b - S.A. prediction and attack ratio calculation:** The S.A. model performs its evaluation of Y_1. The attack ratio of a vehicle is calculated, it is the number of predicted attack messages by S.A. to the total number of predictions.
- **Step 2c - Training Collab. model:** The attack ratio, calculated in the previous step, is combined with the remaining features extracted from Y_2, and all the eight features are utilized to train the Collab. model.
- **Step 3a - Input test set to S.A.:** After S.A. and Collab. models are trained, the third step is to utilize the test subset Z to compare the performances of both models. Similar to subset Y, subset Z is divided into two halves, where the first half Z_1 is evaluated using S.A. model only.
- **Step 3b - S.A. prediction and attack ratio calculation:** S.A. performs message classification for the messages assumed under the first area, Z_1, and the attack ratio is then calculated.
- **Step 3c - Input test set to S.A. and Collab.:** For a fair evaluation, we limit the comparison of the models to subset Z_2 only, because it is the only set evaluated by both models.
- **Step 3d - Performance comparison:** The results of this process are shown in Table 4.2. The scores of the Standalone and the Collaborative models are, respectively: 0.9400 and 0.9721 on accuracy, 0.9335 and 0.9338 on precision, 0.8518 and 0.9716 on recall, 0.8908 and 0.9523 on F1-score. The enhanced performance of the Collaborative model across all metrics during the offline evaluation is a strong indicator of the positive impact derived from incorporating the attack ratio feature.

Table 4.2: Offline Results: Standalone and Collaborative Models

ML Model	*Accuracy*	*Precision*	*Recall*	*F1-score*
S.A.	0.9400	0.9335	0.8518	0.8908
Collab.	0.9721	0.9338	0.9716	0.9523

4.5.2 . Online Evaluation

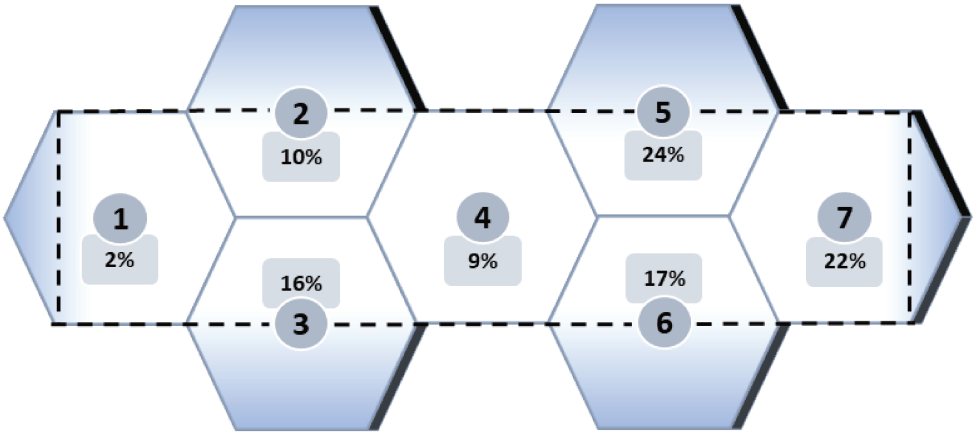

Figure 4.4: Online Scenario: 5G service areas and V2X messages distribution

To validate these findings in a more realistic environment with multiple areas, we conduct an online evaluation. To achieve this objective, we implement an emulation of 5G core network using free5GC, including eight UPFs for the user-plane. Seven of them serve as I-UPFs, also called branching UPFs, providing access to local edge networks, or LADNs. And one PSA-UPF providing access to DN. Next to each branching UPF, two application instances are created in the edge network: an instance of the V2X application, and another instance for the misbehavior detection function.

We use UERANSIM [97] to simulate UEs and the RAN part of the system. UERANSIM exchanges control messages with free5GC to authenticate and register 5G base stations and UEs, which are considered vehicles. When a vehicle is registered and authenticated, the required user-plane GTP tunnel is created. The vehicle, depending on its actual location, can then communicate with its local V2X AS, passing through the gNB/UPF pair serving the vehicle's specific 5G service area.

In the dataset, the range of coordinates sent by vehicles forms a rectangular field. To provide cellular connectivity across the map and maximize the number of transiting vehicles between areas, we divide the map into seven coverage areas as show in Figure 4.4. Each area corresponds to a small cell that consists of a gNB, served by an I-UPF, and instances of both local V2X

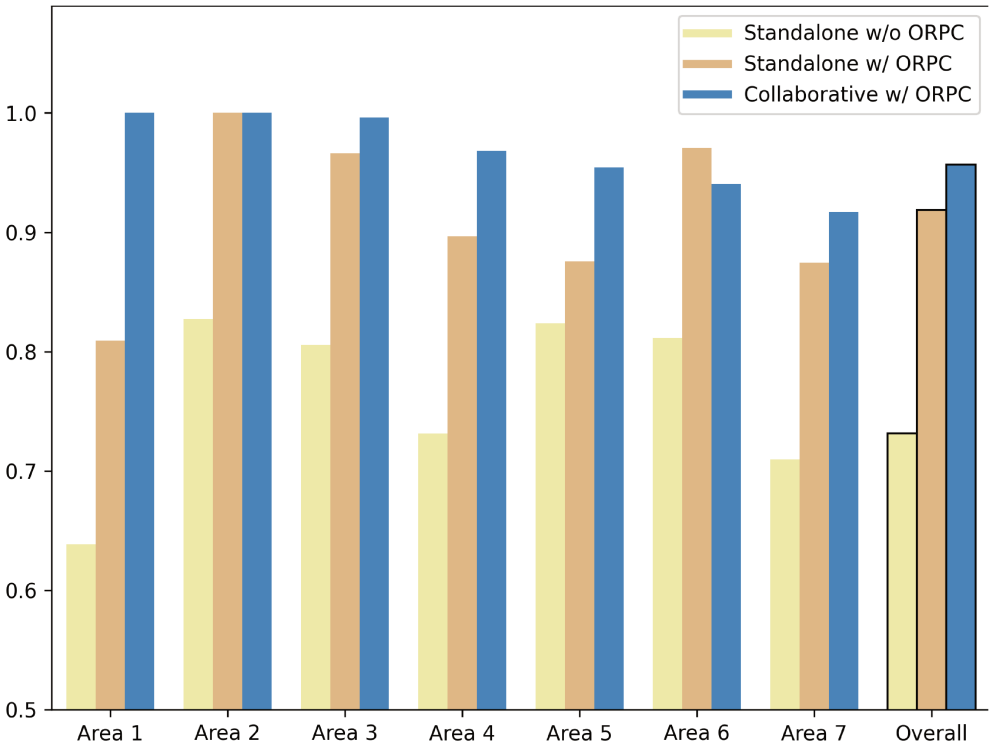

Figure 4.5: Online Results: Accuracy of Standalone and Collaborative Models

AS and DF. Neighboring DF instances are interconnected and therefore can share the attack ratios of vehicles moving between them.

During the experiment, both standalone and collaborative models analyzed in real-time over $28,000$ generated messages sent by vehicles as they moved between two or more areas. The results of the online testing phase are summarized in Figure 4.5 and detailed in Table 4.3. They confirm the advantages of using Collab. model over S.A. model, previously observed in the offline results.

In terms of areas, the collaborative model consistently demonstrates superior performance in six out of the seven areas, with the exception being area 6. In general, the collaborative model has superior performance in terms of accuracy, recall, and F1-score. The respective improvements in performance, compared to the performance of the standalone model, were 3.8 percent, 4.7 percent, and 2.5 percent. Nevertheless, the standalone model has a

Table 4.3: Online Results: Standalone and Collaborative Models

		Attack 1		Attack 2		Attack 4		Attack 8		Attack 16		All Attacks	
Metric	**Area**	*S.A.*	*Collab.*	*S.A.*	*Collab.*	*S.A.*	*Collab.*	*S.A.*	*Collab.*	*S.A.*	*Collab.*	*S.A.*	*Collab.*
	Area 1	0.7182	1.0000	0.9945	1.0000	1.0000	1.0000	1.0000	1.0000	0.7143	1.0000	0.8092	1.0000
	Area 2	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
	Area 3	0.9729	0.9986	0.9533	1.0000	0.9986	1.0000	0.9953	1.0000	0.9729	0.9898	0.9661	0.9962
Accuracy	*Area 4*	0.8555	1.0000	0.9782	1.0000	0.9966	1.0000	0.9897	1.0000	0.8555	0.9048	0.8964	0.9683
	Area 5	0.8252	0.9313	0.9869	1.0000	0.9995	1.0000	0.9896	1.0000	0.8256	0.9313	0.8756	0.9542
	Area 6	0.9797	0.9679	0.9545	0.9196	0.9938	0.9601	0.9857	0.9334	0.9739	0.9197	0.9707	0.9405
	Area 7	0.8209	0.8760	0.9924	1.0000	0.9995	1.0000	0.9886	1.0000	0.8203	0.8745	0.8746	0.9169
	Overall	**0.8912**	**0.9501**	**0.9779**	**0.9863**	**0.9982**	**0.9932**	**0.9908**	**0.9886**	**0.8909**	**0.9315**	**0.9187**	**0.9569**
	Area 1	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
	Area 2	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
	Area 3	0.9972	1.0000	0.9970	1.0000	0.9973	1.0000	0.9973	1.0000	0.9972	1.0000	0.9994	1.0000
Precision	*Area 4*	0.9905	1.0000	0.9929	1.0000	0.9932	1.0000	0.9931	1.0000	0.9905	1.0000	0.9984	1.0000
	Area 5	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
	Area 6	0.9860	0.9365	0.9866	0.9364	0.9877	0.9415	0.9875	0.9383	0.9872	0.9366	0.9974	0.9869
	Area 7	0.9985	1.0000	0.9990	1.0000	0.9991	1.0000	0.9990	1.0000	0.9985	1.0000	0.9998	1.0000
	Overall	**0.9956**	**0.9885**	**0.9965**	**0.9895**	**0.9966**	**0.9896**	**0.9965**	**0.9895**	**0.9957**	**0.9882**	**0.9992**	**0.9978**
	Area 1	0.4333	1.0000	0.9890	1.0000	1.0000	1.0000	1.0000	1.0000	0.4286	1.0000	0.7709	1.0000
	Area 2	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
	Area 3	0.9485	0.9973	0.9093	1.0000	1.0000	1.0000	0.9932	1.0000	0.9485	0.9797	0.9599	0.9954
Recall	*Area 4*	0.7179	1.0000	0.9633	1.0000	1.0000	1.0000	0.9862	1.0000	0.7179	0.8096	0.8771	0.9619
	Area 5	0.6504	0.8627	0.9738	1.0000	0.9991	1.0000	0.9792	1.0000	0.6510	0.8626	0.8507	0.9450
	Area 6	0.9710	1.0000	0.9214	0.9002	1.0000	0.9813	0.9838	0.9278	0.9602	0.9005	0.9672	0.9408
	Area 7	0.6422	0.7517	0.9858	1.0000	1.0000	1.0000	0.9782	1.0000	0.6410	0.7486	0.8497	0.9002
	Overall	**0.7839**	**0.9100**	**0.9593**	**0.9830**	**0.9998**	**0.9968**	**0.9851**	**0.9876**	**0.7850**	**0.8733**	**0.9031**	**0.9503**
	Area 1	0.6047	1.0000	0.9945	1.0000	1.0000	1.0000	1.0000	1.0000	0.6000	1.0000	0.8706	1.0000
	Area 2	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
	Area 3	0.9722	0.9986	0.9512	1.0000	0.9986	1.0000	0.9952	1.0000	0.9722	0.9897	0.9793	0.9977
F1-score	*Area 4*	0.8324	1.0000	0.9779	1.0000	0.9966	1.0000	0.9896	1.0000	0.8324	0.8948	0.9338	0.9806
	Area 5	0.7882	0.9263	0.9867	1.0000	0.9995	1.0000	0.9895	1.0000	0.7886	0.9262	0.9193	0.9717
	Area 6	0.9784	0.9672	0.9529	0.9180	0.9938	0.9610	0.9857	0.9330	0.9735	0.9182	0.9821	0.9633
	Area 7	0.7817	0.8582	0.9924	1.0000	0.9995	1.0000	0.9885	1.0000	0.7807	0.8562	0.9187	0.9475
	Overall	**0.8772**	**0.9476**	**0.9775**	**0.9862**	**0.9982**	**0.9932**	**0.9908**	**0.9886**	**0.8779**	**0.9272**	**0.9487**	**0.9735**

very slight advantage in terms of precision.

The attack types that exhibited the most notable enhancements are attack type 1 and attack type 16. The primary factor contributing to the sub-optimal performance of attack type 1 lies in our real-time prediction approach, which takes into account only the last two messages without considering the complete trajectory of the vehicle. Consequently, when a malicious vehicle remains stationary in reality while transmitting a fake fixed position, the standalone model erroneously classifies this behavior as benign. On the other hand, the collaborative model mitigates this limitation by adding the historical behavior of the vehicle through the attack ratio feature, leading to a notable 13 percent improvement in the detection rate for attack type 1. Finally, there is a notable improvement in the detection of attack type 16, characterized by inconsistent labelling, with an increase of 9 percent.

4.6 . Conclusion

In this chapter, we propose a collaborative V2X misbehavior detection system to protect V2X application servers in the 5G edge network. To detect position manipulation attacks, we propose an improved machine learning model which leverages collaboration between detection nodes to improve performance.

Our work presents a significant step in exploring the advantages of leveraging collaboration between edge network nodes to enhance detection results. More studies are needed to explore different methods of collaboration and address more sophisticated V2X attacks.

5 - FEDERATED LEARNING FOR V2X MISBEHAVIOR DETECTION IN 5G

Contents

5.1 . Introduction

Maintaining the integrity and security of V2X systems requires detecting and preventing malicious activities that might target V2X-enabled devices, including position falsification, Denial of Service (DoS), sybil and replay attacks. Misbehavior detection is a critical component for protecting V2X communications.

Initial proposals to address misbehavior detection in V2X rely on creating simple plausibility checks to detect abnormal behaviors. Later, several papers leverage these plausibility checks along with additional vehicles' movement data to train traditional machine learning models to improve detection results. With the advancement of machine learning and deep learning, new solutions are proposed based on Recurrent Neural Networks (RNN) due to their ability to retain trajectory information.

With the emergence of AI and 5G, the potential to leverage techniques such as Federated Learning to effectively resolve the scalability challenge has emerged as a promising solution.

In this chapter, we propose a distributed misbehavior detection system based on LSTM and Federated Learning, where the nodes of the system are installed in the 5G edge network across the coverage zone, to protect C-ITS application servers hosted on the edge, cloud, or the internet, against a wider variety of V2X attacks.

The scalability of Federated Learning makes it the best solution for implementing a V2X misbehavior detection system in large-scale deployments. Due to the anticipated exponential growth of the number of connected vehicles and low-latency requirements of some V2X applications, traditional centralized approaches may not always be capable of meeting the massive data volume and computational demands. Federated Learning distributes the learning process, enabling edge nodes to provide their local knowledge and participate in training without overwhelming central servers. This distributed computing paradigm enables efficient and scalable V2X misbehavior detection systems, capable of accommodating the expected growth in vehicle connectivity.

Federated Learning provides a convenient solution for detecting V2X misbehavior in 5G networks. It enables V2X systems to detect and prevent position falsification attacks and other malicious behaviors while protecting the privacy of individual vehicles by leveraging the power of collaborative and privacy-preserving learning. Federated Learning paves the way for robust and effective V2X misbehavior detection systems in the era of 5G-enabled connected vehicles via its adaptability, scalability, and ability to integrate real-time data from diverse sources.

5.2 . Problem Statement

While the primary line of defense for V2X networks relies on the vehicular PKI systems, which are crucial for protecting against external threats, additional measures are necessary to mitigate attacks that could be launched by malicious insiders who have already authenticated and are part of the network. To address this issue, a Misbehavior Detection System can be imple-

mented, it acts like an Intrusion Detection/Prevention System (IDS/IPS) in traditional IT networks.

The introduction of advanced machine learning algorithms has emerged as a viable approach to enhance security measures inside the expanded 5G network architecture, in the developing landscape of V2X communications. Federated Learning, characterized by its decentralized training methodology, introduces a new framework that has the potential for improving scalability and data privacy. Nevertheless, the use of Federated Learning into V2X systems for detecting misbehavior, especially inside 5G edge networks, is an area that has not been thoroughly studied.

The primary issue addressed in this chapter is to explore the effective use of Federated Learning for the purpose of V2X misbehavior detection in 5G edge networks. Additionally, it examines whether Federated Learning can maintain the detection performance attained by conventional centralized learning approaches used in a V2X misbehavior detection context.

We take into account all the nineteen V2X misbehavior types included in VeReMi-extension dataset [90], which are depicted in Figure 5.1. In the figure, the blue dots represent the authentic values that the vehicle would have transmitted if it was benign, whereas the red and other colored dots are the values that were actually transmitted by the attacker. The gray dots are an aggregation of normal values transmitted by non-malicious vehicles.

The attack types are the following:

- **Attack type** 1 **(Constant Position):** The misbehaving vehicle reports the same fixed position despite its movement.
- **Attack type** 2 **(Constant Position Offset):** The vehicle adds a predetermined fixed value to its current position, resulting in the generation of a path that is parallel to the true path.
- **Attack type** 3 **(Random Position):** The misbehaving vehicle sends a random position instead of its actual one.
- **Attack type** 4 **(Random Position Offset):** The vehicle adds a random number to its true position, creating a fuzzy path.
- The next four attacks, namely **Attack type** 5 **(Constant Speed)**, **Attack**

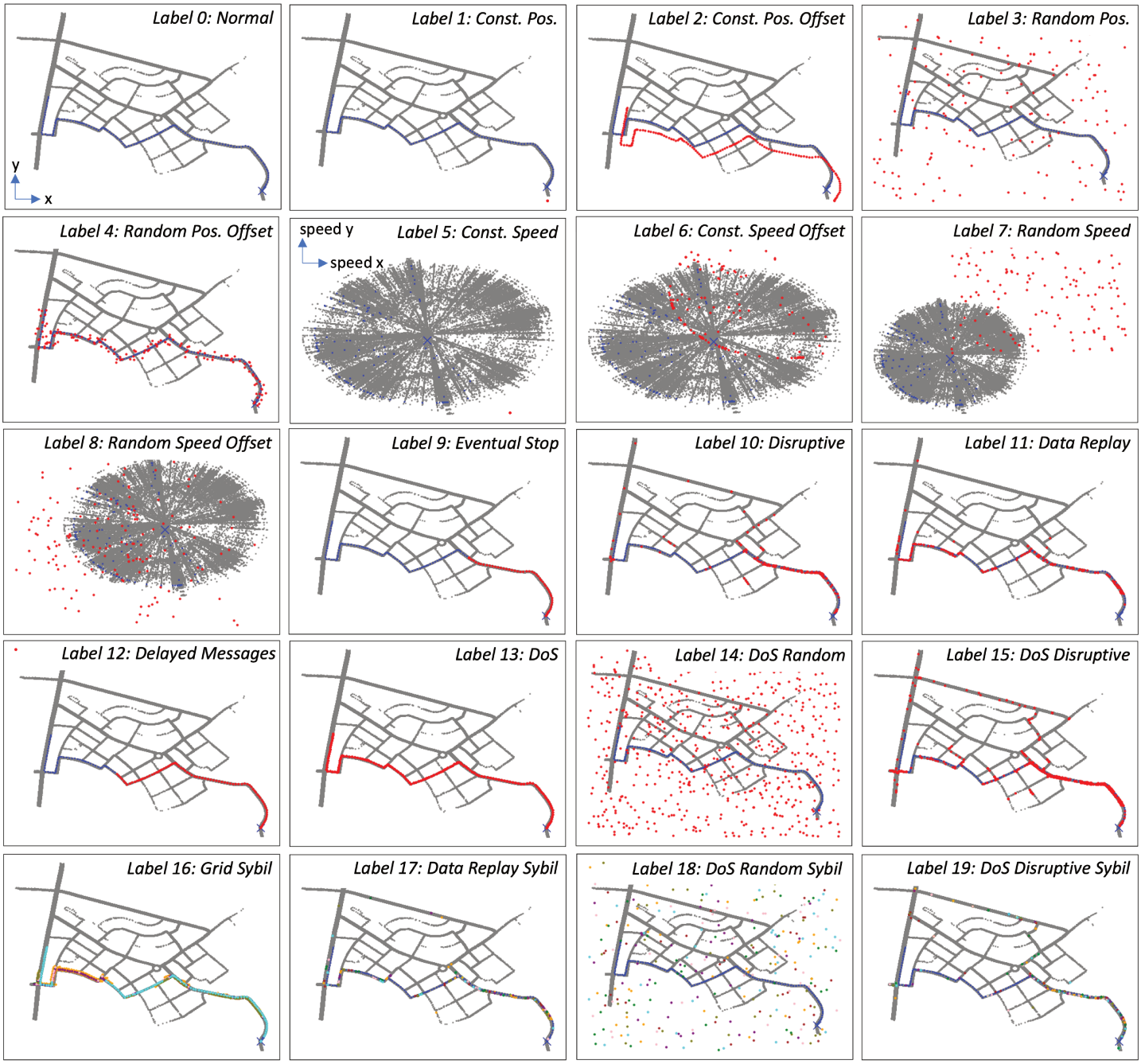

Figure 5.1: VeReMi Extension Dataset Classes

type 6 **(Constant Speed Offset)**, **Attack type** 7 **(Random Speed)**, and **Attack type** 8 **(Random Speed Offset)**, are respectively similar to the first four, the only difference is that the misbehaving vehicle alters its speed value instead of its location. This is also reflected in the four corresponding graphs in Figure 5.1 which depict speed values.

- **Attack type** 9 **(Eventual Stop):** The vehicle sends its precise location and speed information at first, but after a certain duration, it starts reporting a fake fixed position and zero speed.
- **Attack type** 10 **(Disruptive):** The vehicle replays information previously received from random neighbors to overwhelm the network.
- **Attack type** 11 **(Data Replay):** The attacking vehicle replays the information received from a specific target neighbor as its own.
- **Attack type** 12 **(Delayed Messages):** The attacking vehicle sends its old accurate movement information after a pre-defined period of time.
- **Attack type** 13 **(DoS):** The attack consists of sending accurate information at an increased frequency to flood the network.
- **Attack type** 14 **(DoS Random):** The attack is similar to DoS in terms of message frequency, however, all the values included in the messages are random and not accurate.
- **Attack type** 15 **(DoS Disruptive):** This attack combines the increased sending frequency with the flooding of previously received random neighbors' messages.
- **Attack type** 16 **(Grid Sybil):** The intention of the attacker is to create a fake congestion by creating ghost vehicles, where vehicle pseudo IDs are created for nonexistent vehicles in a specific target position, and the attacker maintains a realistic communication profile of the ghost vehicles. Each pseudo ID is represented with a different color in the corresponding graph in Figure 5.1.
- **Attack type** 17 **(Data Replay Sybil):** This attack is a more sophisticated form of Attack type 11. The attacker replays the data of a targeted neighbor using multiple pseudo IDs to masquerade the real attacker's identity.

- **Attack type** 18 **(DoS Random Sybil):** It combines three forms of attacks into one: increased frequency of sent messages, randomness of all values in the messages, and multiple fake pseudo IDs.
- **Attack type** 19 **(DoS Disruptive Sybil):** The attacker replays previously received messages from random neighbors with high frequency using many pseudo IDs.

As 5G enables low-latency C-ITS applications in the Edge networks, it is important to protect them against V2X attacks launched by misbehaving vehicles, which can have significant impact and might disrupt the operations of critical C-ITS services. Therefore, in this chapter, we propose a novel approach to detect and mitigate V2N misbehavior by leveraging LSTM and Federated Learning.

5.3 . 5G Edge Misbehavior Detection System Architecture

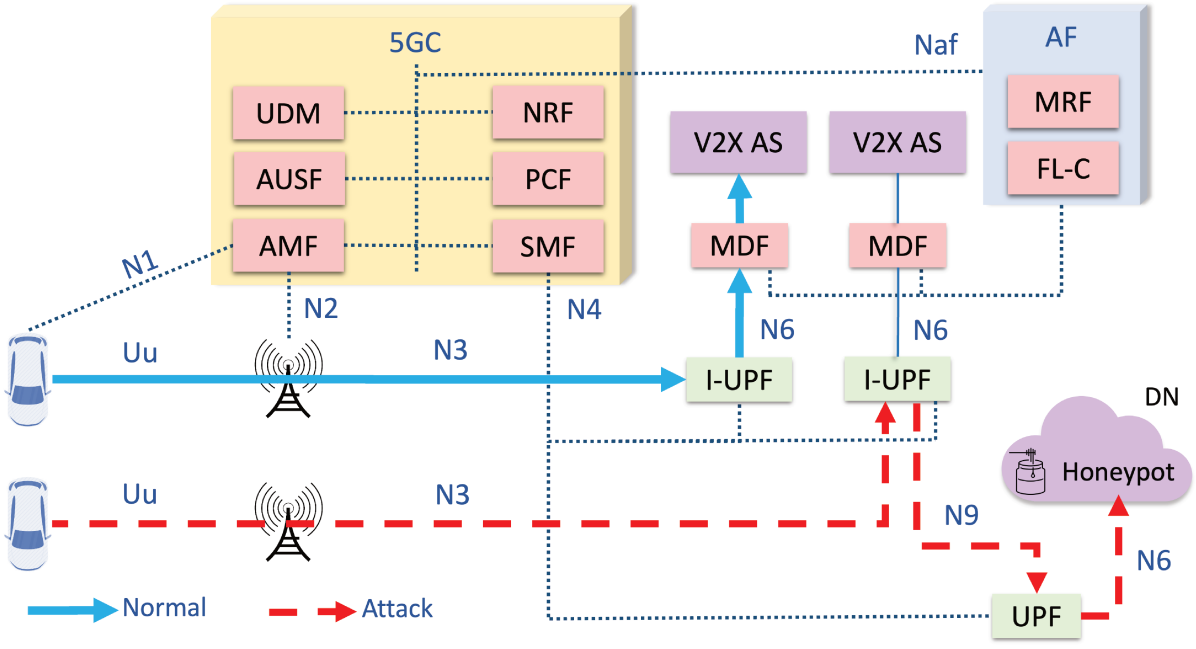

Figure 5.2: Proposed architecture for a Federated Learning-based 5G edge misbehavior detection system

C-ITS services requiring low latency may need to be hosted on the edge network deployed as multiple application instances. In this case, the misbehavior detection system needs to be implemented accordingly.

The proposed architecture, depicted in Figure 5.2, consists of:

- Several instances of Misbehavior Detection Functions (MDFs) distributed geographically on Edge networks.
- Federated Learning Central (FL-C) server.
- Misbehavior Reporting Function (MRF) in the central cloud coupled with the 5G core network.
- Honeypot server.

The MDFs are in charge of real-time processing and monitoring of V2N traffic packets. Each MDF instance is connected to an Intermediate-UPF (I-UPF) on a Local Access Data Network (LADN). The MDF is also considered as a Federated Learning client, where it shares its local model parameters with the FL-C server without sharing local training data collected from vehicles.

The Federated Learning Central server acts as a central hub for organizing and compiling improved machine learning models derived from the received models from MDF instances. Without having any access to local data, it averages the models' weights to enable collaborative learning. This methodology preserves data security and privacy while enabling scalability and leveraging the collective intelligence of all MDF instances.

Misbehavior Reporting Function (MRF) is considered an Application Function (AF) co-located with the 5GC network. In case an attack is detected, the MRF has the ability to initiate traffic-steering requests to the core network, which can re-route the attacker's traffic to the Honeypot server instead of the C-ITS application instance, protecting the application from falsified data. It also records security incidents and notifies Telecommunication network operators or legal authorities, who can completely revoke access for the misbehaving vehicle or take legal action if necessary.

The Honeypot server has the role of a decoy server that will collect malicious V2X messages, creating a new dataset of attacks that can be studied

and leveraged to improve the training and efficiency of misbehavior detection models.

Our proposed architecture aims to offer enhanced data privacy, improved application security, and effective misbehavior detection for V2X ASs in C-ITS systems. It achieves this goal by combining the power of Federated Learning, 5G control functions and traffic steering functionality, and cybersecurity best practices.

5.4 . Detection Model Proposal

5.4.1 . Deep Learning

5.4.1.1 . Artificial Neural Networks (ANNs)

Deep learning is a sub-branch of machine learning, and both are within the field of artificial intelligence. Machine learning algorithms, including linear regression, decision trees, and support vector machines, acquire knowledge from data and use it to generate predictions or make judgments. In contrast, deep learning utilizes neural networks characterized by several layers, hence the term "deep". Deep neural networks are specifically designed to autonomously and flexibly acquire complex data representations, making them highly suitable for tasks involving large quantities of data, such as the recognition of images and sounds. Machine learning methods often need feature engineering and user intervention, while deep learning models usually have the capability to autonomously extract features from unprocessed data. On the other hand, deep learning requires larger datasets and is demanding when it comes to computing resources, hence cutting-edge GPUs are usually needed.

Deep learning, has emerged as a very influential domain in the field of AI. It is inspired by the structure of the human brain, particularly the neural networks that mimic brain's neurons connections. Artificial Neural Networks (ANNs) are composed of many neurons and they are capable of complex data processing, hence empowering computers to do activities that were previously believed to be within the realm of human capabilities.

A single neuron inside a neural network, as depicted in Figure 5.3, performs a computation to generate an output, which is determined by the in-

put it receives. This output is frequently attained by calculating a weighted sum of the inputs, including a bias factor, and then passing the the sum to an activation function.

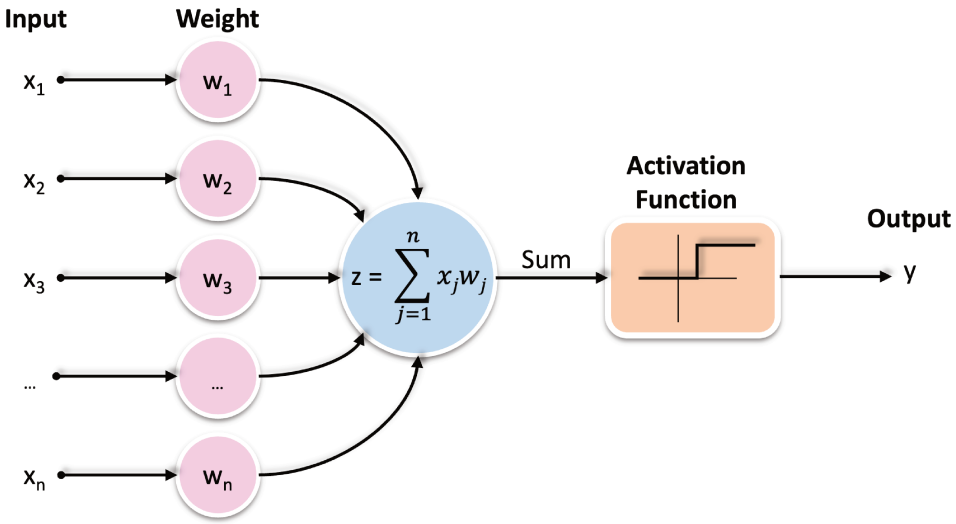

Figure 5.3: Neuron, neural network node

Given the following notation:

$x_1, x_2, \ldots, x_n$= input values

$w_1, w_2, \ldots, w_n$= weights associated with each input

f= activation function

The output y of a single neuron can be calculated as:

1. Compute the weighted sum:

$$z = w_1 x_1 + w_2 x_2 + \cdots + w_n x_n$$

2. Pass the weighted sum through an activation function:

$$y = f(z)$$

Some commonly used activation functions include:

- Sigmoid function, which maps any input value to the range $(0, 1)$:

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

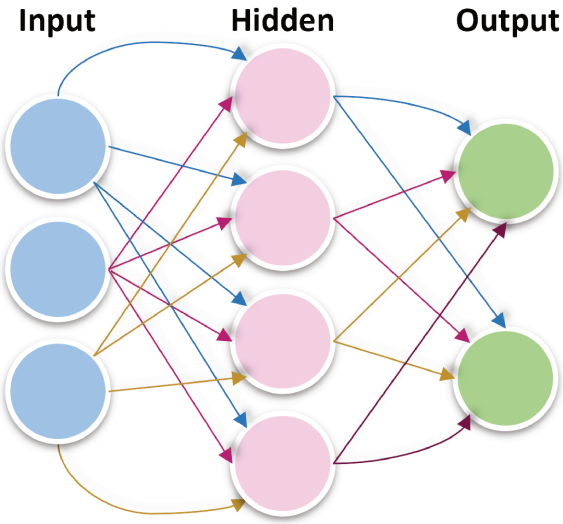

Figure 5.4: Artificial Neural Network (ANN)

- Hyperbolic Tangent, which maps any input value to the range $(-1, 1)$:

$$\tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$$

- Rectified Linear Unit (ReLU), which maps any negative value to zero:

$$\text{ReLU}(z) = \max(0, z)$$

An ANN, represented in Figure 5.4, is comprised of connected neurons which are arranged in layers. These layers include an input layer, one or more hidden layers, and an output layer. The weight of each connection between these neurons is modified throughout the training process. Neurons are responsible for the processing of incoming input, whereby they apply various transformations via the use of activation functions, afterwards transmitting the resulting output to the subsequent layer. These networks are highly sophisticated, enabling them to effectively represent and analyze complicated patterns and interconnections within datasets.

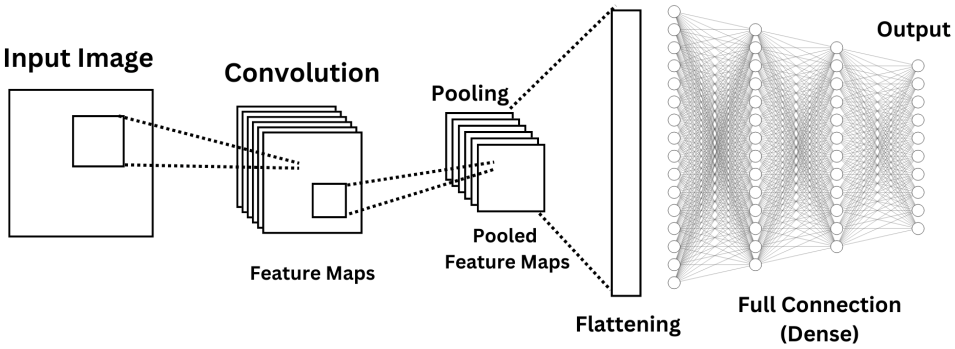

Figure 5.5: Convolutional Neural Network (CNN)

The process of training a deep learning model entails providing it with large quantities of data and iteratively modifying the weights of its connections in order to decrease the difference between its predictions and the data labels. The accomplishment of this task is often realized thanks to the use of algorithms such as gradient descent. The objective is to identify the most favorable combination of weights that leads to the minimal prediction error, as measured by a loss function.

5.4.1.2 . Other Forms of Neural Networks: CNNs and RNNs

ANNs are considered the basic form of deep learning, more sophisticated deep learning designs are able to unlock more advanced capabilities. An example of that are Convolutional Neural Networks (CNNs), which are designed for the purpose of image processing. Convolutional layers are used to perform localized scanning of input pictures, enabling the detection of various patterns such as edges, textures, and forms, as depicted in Figure 5.5.

In contrast, Recurrent Neural Networks (RNNs), depicted in Figure 5.6, are specifically designed to handle sequential input, rendering them well-suited for applications such as language modeling or time series forecasting.

An RNN operates on sequences of data. At each timestep t, given:

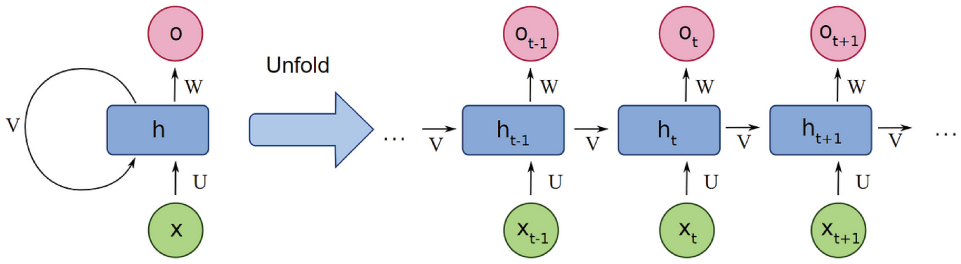

Figure 5.6: Recurrent Neural Networks (RNNs)

- h_{t-1} - previous hidden state
- x_t - current input
- U, V, W - weight matrices

The hidden state h_t for the current timestep is updated as:

$$h_t = \sigma(Vh_{t-1} + Ux_t)$$

Where σ is an activation function, such as the hyperbolic tangent or the sigmoid function.

For certain tasks, there is an output at each timestep. This output is computed as:

$$o_t = Wh_t$$

Where W is the weight applied to the respective hidden state.

Deep learning models have the ability to produce exceptional performance on tasks such as picture and voice recognition, surpassing typical machine learning algorithms, mostly thanks to their capability of achieving high degrees of abstraction. Nevertheless, they do not come without their challenges. In order to train effectively, a large quantity of labeled data and considerable computer resources are necessary. Moreover, deep learning models, especially those of higher complexity, might sometimes referred to as "black boxes", hence presenting difficulties in comprehending their decision-making mechanisms.

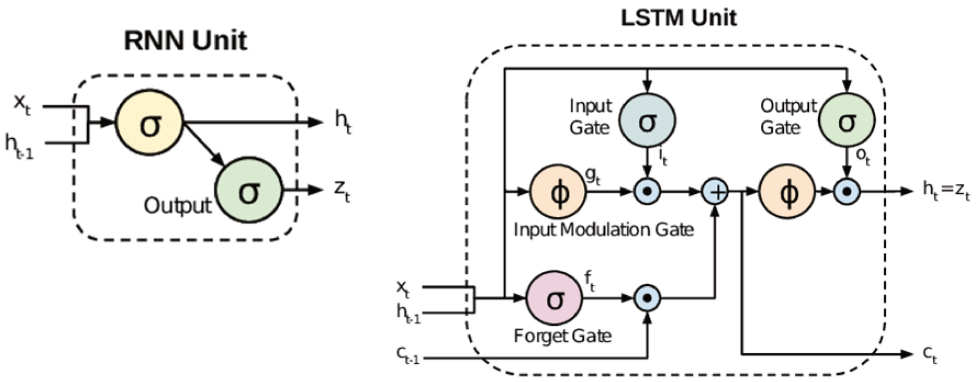

Figure 5.7: RNN and LSTM Units

5.4.1.3 . Long Short-Term Memory (LSTM)

LSTM [98] is a type of RNNs that has been developed to identify patterns within sequential data, such as time series or natural language. Conventional RNNs have difficulties in retaining long-term dependencies inside a sequence. This is precisely where LSTM networks seem to be advantageous.

One notable advantage of LSTM is in its capacity to retain information from extended sequences, particularly those that are highly suitable for Natural Language Processing (NLP) applications. This ability surpasses that of conventional RNNs, enabling LSTM to effectively process far longer sequences. This is accomplished by the use of a gating mechanism that exhibits selectivity in permitting the passage of information both into and out of the system. The use of memory cells in LSTM models enables the retention or elimination of information, enhancing hence their ability to comprehend context and provide predictions accordingly. A window, in LSTM, is subset of sequential data that is analyzed at each time step. It denotes a predetermined amount of time or a specified number of prior time steps that the LSTM model will take into account while making predictions or doing analysis. The input sequence can be overlapping or non-overlapping. The LSTM model uses these windows as input, which enables it to recognize temporal correlations and patterns in the data. In other words, the amount of historical data that the LSTM takes into account at each time step depends on the size or duration of the window.

The LSTM memory cell, depicted in Figure 5.7, is comprised of four primary components, namely the input gate, the input modulation gate, the for-

get gate, and the output gate. The function of these gates is to regulate the processes of information storage, information removal, and value output in the present time step.

Given the notation:

- i_t: Activation vector of the input gate at time t.
- f_t: Activation vector of the forget gate at time t.
- o_t: Activation vector of the output gate at time t.
- g_t: Candidate value for the memory cell at time t.
- c_t: Memory cell state at time t.
- h_t: Hidden state at time t.
- x_t: Input vector at time t.
- W: Weight matrix (subscript denotes the gate or operation it's associated with).
- σ: Sigmoid activation function.
- $\tanh$: Hyperbolic tangent activation function.

1. Input Gate: It determines how much of the new information should be stored in the memory cell.

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t])$$

2. Input Modulation Gate: This gate generates a candidate value that could be added to the state.

$$g_t = \tanh(W_g \cdot [h_{t-1}, x_t])$$

3. Forget Gate: It decides which parts of the memory cell's previous state should be discarded or retained.

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t])$$

4. Output Gate: It determines what portion of the memory cell's current state should be outputted.

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t])$$

$$h_t = o_t \times \tanh(c_t)$$

5. Memory Cell Update: The memory cell's state is updated based on the decisions made by the above gates.

$$c_t = f_t \times c_{t-1} + i_t \times g_t$$

5.4.1.4 . Federated Learning

Introduced by Google in [99], Federated Learning enables the decentralization of the training process, allowing for model training directly on devices or nodes where the data is hosted. In contrast, traditional centralized machine learning requires the data to be sen to a central server. The primary objectives behind the development of Federated Learning are related to data privacy, scalability, bandwidth efficiency, and latency.

The fundamental objective of Federated Learning is to facilitate the training of a global model by using data from several devices, while avoiding the need of directly sharing raw data. Every individual device updates local model, using its own local data, and thereafter transmits just this updated model to a central server. The models are collected and aggregated by the server in order to enhance the global model, which is then sent back to the devices for further local training. This iterative process is repeated until the model convergences or satisfies a predetermined criteria.

Federated Learning offers several advantages. It enhances data privacy since raw data is not shared, reducing the risk of data breaches. It also permits the training on edge devices, making it suitable for applications where real-time insights are crucial, and bandwidth or connectivity is limited. However, Federated Learning also presents challenges. The non-IID (independent and identically distributed) nature of local datasets can affect model convergence. Additionally, devices with limited computational resources might struggle with complex model training.

Given the notation:

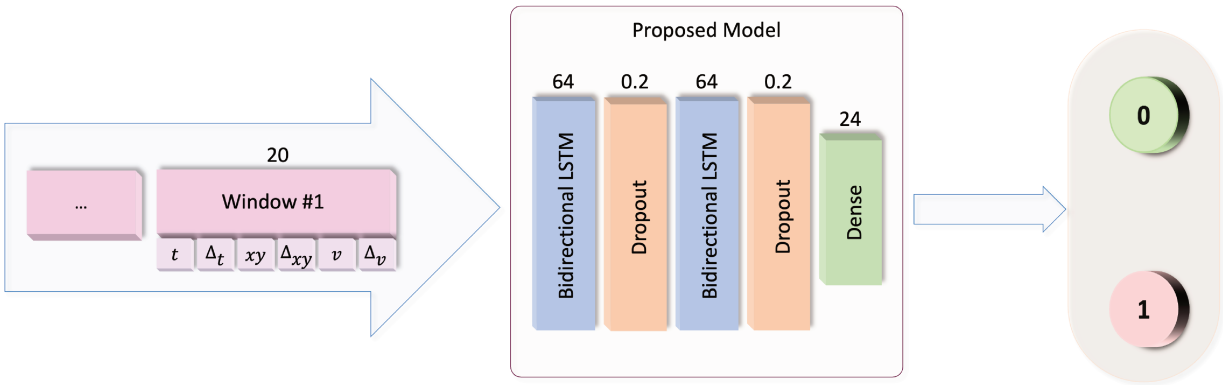

Figure 5.8: Proposed LSTM Model

- θ: Global model parameters.
- D_k: Local dataset on device k.
- $\mathcal{L}_k$: Loss function computed on the local dataset D_k.
- $\Delta\theta_k$: Local model update computed on device k.
- w_k: Weighting factor for device k, often proportional to the size of D_k.
- α: Learning rate.

Local Model Update

Each device k computes a local update $\Delta\theta_k$ based on its local dataset D_k and the current global model θ.

$$\Delta\theta_k = \text{argmin}_{\Delta\theta}\mathcal{L}_k(\theta - \Delta\theta)$$

Model Aggregation

The central server aggregates the local updates from all participating devices to update the global model.

$$\theta_{\text{new}} = \theta - \alpha \sum_k w_k \Delta\theta_k$$

5.4.2 . Proposed Model: LSTM/Federated Learning

We propose an AI model that leverages the advantages of both LSTM and Federating Learning. LSTM emerges as an intuitive option for processing vehicle movement data because of its distinct memory features. Positions, velocities, and other variables that make up vehicle movement data are fundamentally time-dependent, therefore, LSTM can successfully capture the underlying

patterns and dependencies by maintaining an internal memory state. It is capable of modeling the dynamic characteristics of vehicle movement. In tasks such as abnormal route detection or route prediction, understanding the links between past and future positions is essential, and it can be helpful in vehicle movement analysis. This gives the model the ability to accurately classify normal and abnormal trajectories and forecast future positions, ensuring a thorough understanding of the vehicle's movement dynamics.

In Federated Learning, data processing and model training can be directly performed on edge nodes, enabling the protection of V2X server in the edge networks. In doing so a significant reduction in latency can be achieved. Federated Learning effectively manages large training datasets by utilizing the processing capacity of multiple devices. It also favors better data representation. The training method captures rich variations and nuances by combining a variety of data sources from diverse devices or places. The resulting global model benefits from a wider range of viewpoints, guaranteeing improved generalization across various contexts. Federated Learning can also enable cooperation between multiple Telecommunications operators. It makes it easier for them to collaborate without exchanging users' data or location. It promotes collaboration and information sharing while adhering to privacy laws and protecting company interests by allowing model training without location data sharing.

5.4.2.1 . Features

Basic vehicle movement features are mainly related to time, position, and speed in both longitude (x) and latitude (y) directions. They might also include altitude, acceleration, and heading. Training a deep learning misbehavior detection model using only raw features does not always lead to optimal performance. Also, using the time and position values of a dataset in their raw format may cause the model to overfit, which might degrade accuracy with new data while performing well on training data. To avoid these problems, we add differential features like *deltaTime* and *deltaPosition*, which represent the differences between the current received position and time and their recently received values from the same vehicle. Also, to verify the consistency between the received values of time, position, and speed, we calculate a new speed value based on *deltaTime* and *deltaPosition*, independently from the received

speed value. After that, we deduct the newly calculated speed value from the received speed value to obtain *deltaSpeed*. In summary, the features we used comprise: *time*, *deltaTime*, *position*, *deltaPosition*, *speed*, and *deltaSpeed*.

5.4.2.2 . Federated Averaging

The implementation of the Federated Averaging function, summarized in Algorithm 1, accepts three parameters, namely *client_models*, *data_sizes*, and *performances*. The *client_models* parameter is a set of client models which are participating in the federated averaging process. The *data_sizes* parameter denotes the corresponding size of training data utilized by each client model. The *performances* parameter represents the accuracy scores of each client model. To determine the coefficient assigned to each client model during the averaging process, we utilize training data sizes and performance, giving the former double the importance of the latter. To obtain the averaged model, the algorithm aggregates the multiplications of the layer weights of each model by its corresponding client model coefficient, calculated in the previous step. The averaged model will then be shared with clients. This process is considered one cycle and can be repeated as necessary.

5.5 . Performance Evaluation

This section outlines the 5G network evaluation environment specifications, dataset criterias, dataset splitting methodology, deep learning and federated learning hyper-parameters utilized in the evaluation, performance metrics, simulation results, and results discussion.

5.5.1 . Evaluation environment, dataset considerations, and models parameters

The assessment of performance was conducted on a modern testing platform, utilizing a machine with high specifications, equipped with an Nvidia GeForce RTX 3090 graphics card, an Intel Core i7-11700KF processor, 64 GB of DDR4 RAM, and 2 TB of NVMe SSD storage. The computational capacity of the machine facilitated the efficient processing of demanding deep learning algorithms that were utilized for the misbehavior detection model. The experimental software setup consisted of VirtualBox 6.1 as the hypervisor, Ubuntu

Algorithm 1: Federated Averaging Function

Parameters:

client_models: set of participating client models

data_sizes: corresponding training data size per client model

performances: accuracy score per client model

Function:

```
norm_data_sizes ← normalize ( data_sizes )
norm_perf ← normalize ( performances )
combine ← 2/3 * norm_data_size + 1/3 * norm_perf
client_norm_coeffs ← normalize ( combine )
averaged_weights ← [ ]
client_models_weights ← [ ]
for model in client_models do
    client_models_weights.append ( model.get_weights ( ) )
for weights in client_models_weights do
    initialize ( averaged_layer )
    for model_id in client_models do
        averaged_layer ← averaged_layer +
        client_norm_coeffs [ model_id ] * weights
    averaged_weights.append ( averaged_layer )
averaged_model.set_weights ( averaged_weights )
return averaged_model
```

20.04 as the operating system, TensorFlow 2.12.0 [100] as the deep learning framework running on Python 3.9.16, free5GC 3.1.1 [95] as the 5G core network emulator, and UERANSIM [97] as the user equipment and radio network simulator.

To enhance the privacy of vehicles in a V2V ad-hoc environment, each vehicle is assigned multiple pseudo certificates and pseudo IDs, which are frequently swapped to protect its real identity. When a pseudo ID change occurs, IP and MAC addresses are also replaced in parallel on the PC5 interface, which is the interface used for V2V, V2I, and V2P communications in C-V2X. The intention is to protect the vehicle from being tracked by malicious peers. However, in a 5G V2N environment, which can be considered as client-server architecture, the interface utilized is the Uu interface, and the privacy require-

ments are different. To communicate on V2N, the vehicle has to pass the 5G authentication process to receive an IP address during the PDU session establishment procedure. Therefore, assuming that 5G security best-practices are followed and IP spoofing attacks are not possible, the 5G core network can always identify the source of V2N traffic and link it to a specific vehicle or UE, even if it uses multiple pseudo IDs. For this reason, our model, which intends to protect V2N traffic, analyzes the dataset based on sender IDs instead of pseudo IDs.

The VeReMi-extension dataset consists of more than 63 gigabytes of recorded vehicle traces in JSON format. It covers 19 types of V2X attacks in a total of 39 datasets. Two datasets per attack (high density: 37.03 vehicles/km^2 and low density: 16.36 vehicles/km^2), each containing two hours of recorded vehicle messages; and one comprehensive dataset named MixAll, which encompassed all 19 attacks that occurred within a 24-hour timeframe (23.29 vehicles/km^2). Note the all of the datasets have a misbehaving vehicle ratio of 30%.

After converting all datasets to CSV format, we appended them with the three differential features discussed in Section 5.4.2.1 (deltaTime, deltaPosition, and deltaSpeed). Then, in order to implement federated learning, we split the map into six equal geographical areas. The data size distribution between areas 1, 2, 3, 4, 5, and 6 is approximately 22%, 9%, 22% 31%, 13%, and 3% respectively. To correctly split vehicles' data between areas, we utilize the Ground Truth files provided in the VeReMi-extension dataset. They contain authentic data that would have been transmitted by the attacking vehicles if they were not malicious.

To evaluate the performance and efficiency of our proposed federated learning scheme, we compare three approaches:

- A single centralized model, trained on all data from all areas.
- Six standalone models, each trained exclusively on the subset of its respective area.
- Our proposed scheme, which consists of six federated models, each trained solely with its corresponding subset while leveraging the Federated Averaging process.

The models' design, depicted in Figure 5.8, consists of two bidirectional LSTM layers with 64 nodes each, two Dropout layers with the rate set to 0.2 to minimize overfitting, and one Dense layer with 24 nodes. During the training phase, we optimized the selected hyperparameters of the model. It utilizes a window size of 20, allowing the inclusion of the preceding 20 vehicle messages. The loss function is binary cross-entropy with Adam used as an optimization algorithm. The latter is widely utilized in neural network models, with a learning rate of 0.0001. The batch size is set to 64. The number of epochs is not fixed, since we implemented an Early Stop mechanism with a patience value of 5, leading to early termination of training if the validation loss did not improve for 5 consecutive epochs.

To compare the models, we use Accuracy, Precision, Recall, and F1-score metrics which are detailed in Section 3.5.2.

The evaluation is conducted in two stages. The initial phase involves a numerical evaluation, wherein the models' performance is assessed offline by directly analyzing the datasets without transmitting the data through the network. The second phase involves utilizing 5GC emulation and UE simulation, wherein the data is transmitted through the 5G network and analyzed on the edge servers.

5.5.2 . VeReMi Extension Offline Analysis

For the numerical analysis stage, the three previously mentioned models are trained on 90% of the MixAll dataset, 10% used for validation, and the remaining 38 datasets are utilized for extensive testing. All datasets were split into six subsets, representing six areas. The results of this comprehensive evaluation are presented in Table 5.1 and Table 5.2. The table represents a weighted average aggregation of the results from all the areas.

The performance of the three models was not satisfactory when tested on three attack datasets, specifically the Constant Position Offset, low density, and the two datasets related to the Eventual Stop attack.

Additionally, we can observe that distributed schemes (standalone and federated) significantly improved detection rates on Delayed Messages attacks compared to centralized model. This attack consists of sending accurate position and speed information after a time delay. It is possible that a

certain vehicle has moved into a new area while reporting its old position in a previous area, which makes it easier for the distributed models to detect out-of-area positions.

Furthermore, we noticed that the areas with the lowest amount of training data, namely area 2 and area 6, performed poorly in standalone mode. Which is even more obvious with Random Position Offset and Random Speed Offset attacks. On the other hand, federated models were able to mitigate the lack of sufficient training data and greatly improve performance in these two areas due to federated averaging.

The primary outcome of this offline stage indicates that our proposed federated learning model has a slightly superior performance across all metrics. However, it is worth noting that the aim of our study is not centered on enhancing performance but rather on demonstrating the feasibility of utilizing federated learning to achieve privacy and scalability benefits without sacrificing the level of protection of C-ITS application servers.

Table 5.1: Offline Results: Centralized and Federated Models Comparison

		Centralized				Federated (our proposal)			
Dataset	**Density**	*Acc.*	*Prec.*	*Recall*	*F1-score*	*Acc.*	*Prec.*	*Recall*	*F1-score*
1-ConstPos	High	0.9903	0.9942	0.9728	0.9834	0.9859	0.9960	0.9553	0.9753
	Low	0.9841	0.9915	0.9575	0.9742	0.9845	0.9949	0.9557	0.9749
2-ConstPosOffset	High	0.9903	0.9942	0.9728	0.9834	0.9859	0.9960	0.9553	0.9753
	Low	0.8563	0.8309	0.5517	0.6631	0.8468	0.9751	0.5187	0.6772
3-RandomPos	High	0.9895	0.9941	0.9709	0.9824	0.9988	0.9960	1.0000	0.9980
	Low	0.9976	0.9925	1.0000	0.9962	0.9985	0.9951	1.0000	0.9976
4-RandomPosOffset	High	0.9978	0.9943	0.9980	0.9962	0.9878	0.9955	0.9625	0.9788
	Low	0.9974	0.9925	0.9995	0.9960	0.9886	0.9949	0.9686	0.9816
5-ConstSpeed	High	0.9939	0.9942	0.9846	0.9894	0.9911	0.9956	0.9737	0.9845
	Low	0.9925	0.9918	0.9843	0.9880	0.9915	0.9950	0.9778	0.9864
6-ConstSpeedOffset	High	0.9973	0.9943	0.9963	0.9953	0.9925	0.9961	0.9782	0.9871
	Low	0.9959	0.9919	0.9951	0.9935	0.9894	0.9949	0.9713	0.9830
7-RandomSpeed	High	0.9983	0.9943	1.0000	0.9971	0.9988	0.9959	1.0000	0.9980
	Low	0.9974	0.9919	1.0000	0.9960	0.9985	0.9951	1.0000	0.9976
8-RandomSpeedOffset	High	0.9968	0.9943	0.9949	0.9946	0.9853	0.9957	0.9539	0.9744
	Low	0.9962	0.9919	0.9962	0.9941	0.9858	0.9949	0.9600	0.9771
9-EventualStop	High	0.8525	0.9899	0.5020	0.6662	0.8895	0.9943	0.6263	0.7685
	Low	0.8360	0.9816	0.4870	0.6510	0.8848	0.9923	0.6388	0.7772
10-Disruptive	High	0.9983	0.9943	1.0000	0.9971	0.9987	0.9957	1.0000	0.9979
	Low	0.9971	0.9919	0.9989	0.9954	0.9978	0.9951	0.9978	0.9965
11-DataReplay	High	0.9962	0.9943	0.9929	0.9936	0.9949	0.9957	0.9874	0.9915
	Low	0.9920	0.9912	0.9831	0.9872	0.9892	0.9950	0.9707	0.9827
12-DelayedMessages	High	0.9036	0.9918	0.6761	0.8040	0.9882	0.9956	0.9640	0.9795
	Low	0.8861	0.9863	0.6462	0.7809	0.9845	0.9948	0.9552	0.9746
13-DoS	High	0.9989	0.9986	0.9997	0.9991	0.9993	0.9989	0.9999	0.9994
	Low	0.9989	0.9982	1.0000	0.9991	0.9991	0.9985	1.0000	0.9993
14-DoSRandom	High	0.9992	0.9987	1.0000	0.9993	0.9994	0.9991	1.0000	0.9996
	Low	0.9987	0.9979	1.0000	0.9990	0.9992	0.9986	1.0000	0.9993
15-DoSDisruptive	High	0.9991	0.9986	1.0000	1.0000	0.9994	0.9990	1.0000	0.9995
	Low	0.9987	0.9979	1.0000	0.9989	0.9992	0.9986	1.0000	0.9993
16-GridSybil	High	0.9993	0.9990	1.0000	0.9995	0.9995	0.9992	1.0000	0.9996
	Low	0.9988	0.9983	1.0000	0.9992	0.9993	0.9990	1.0000	0.9995
17-DataReplaySybil	High	0.9965	0.9943	0.9940	0.9942	0.9945	0.9962	0.9856	0.9909
	Low	0.9930	0.9918	0.9858	0.9888	0.9916	0.9950	0.9782	0.9865
18-DoSRandomSybil	High	0.9987	0.9975	1.0000	0.9988	0.9992	0.9985	1.0000	0.9992
	Low	0.9983	0.9967	1.0000	0.9984	0.9989	0.9979	1.0000	0.9989
19-DoSDisruptiveSybil	High	0.9987	0.9975	1.0000	0.9988	0.9992	0.9985	1.0000	0.9992
	Low	0.9982	0.9965	1.0000	0.9983	0.9989	0.9979	1.0000	0.9989
Overall		*0.9807*	*0.9890*	*0.9408*	*0.9589*	***0.9852***	***0.9963***	***0.9540***	***0.9709***

Table 5.2: Offline Results: Standalone and Federated Models Comparison

		Standalone				Federated (our proposal)			
Dataset	**Density**	*Acc.*	*Prec.*	*Recall*	*F1-score*	*Acc.*	*Prec.*	*Recall*	*F1-score*
1-ConstPos	High	0.9810	0.9927	0.9412	0.9663	0.9859	0.9960	0.9553	0.9753
	Low	0.9768	0.9866	0.9390	0.9622	0.9845	0.9949	0.9557	0.9749
2-ConstPosOffset	High	0.9810	0.9927	0.9412	0.9663	0.9859	0.9960	0.9553	0.9753
	Low	0.8639	0.9708	0.5778	0.7244	0.8468	0.9751	0.5187	0.6772
3-RandomPos	High	0.9981	0.9934	1.0000	0.9967	0.9988	0.9960	1.0000	0.9980
	Low	0.9964	0.9887	1.0000	0.9943	0.9985	0.9951	1.0000	0.9976
4-RandomPosOffset	High	0.8441	0.9280	0.4711	0.6249	0.9878	0.9955	0.9625	0.9788
	Low	0.8263	0.9016	0.4609	0.6100	0.9886	0.9949	0.9686	0.9816
5-ConstSpeed	High	0.9912	0.9930	0.9767	0.9848	0.9911	0.9956	0.9737	0.9845
	Low	0.9896	0.9885	0.9785	0.9835	0.9915	0.9950	0.9778	0.9864
6-ConstSpeedOffset	High	0.9921	0.9933	0.9793	0.9862	0.9925	0.9961	0.9782	0.9871
	Low	0.9887	0.9871	0.9770	0.9820	0.9894	0.9949	0.9713	0.9830
7-RandomSpeed	High	0.9980	0.9931	1.0000	0.9966	0.9988	0.9959	1.0000	0.9980
	Low	0.9962	0.9882	1.0000	0.9941	0.9985	0.9951	1.0000	0.9976
8-RandomSpeedOffset	High	0.9427	0.9912	0.8099	0.8914	0.9853	0.9957	0.9539	0.9744
	Low	0.9411	0.9802	0.8253	0.8961	0.9858	0.9949	0.9600	0.9771
9-EventualStop	High	0.8938	0.9896	0.6438	0.7801	0.8895	0.9943	0.6263	0.7685
	Low	0.8805	0.9808	0.6320	0.7687	0.8848	0.9923	0.6388	0.7772
10-Disruptive	High	0.9971	0.9931	0.9969	0.9950	0.9987	0.9957	1.0000	0.9979
	Low	0.9921	0.9875	0.9877	0.9876	0.9978	0.9951	0.9978	0.9965
11-DataReplay	High	0.9872	0.9930	0.9626	0.9775	0.9949	0.9957	0.9874	0.9915
	Low	0.9763	0.9870	0.9369	0.9613	0.9892	0.9950	0.9707	0.9827
12-DelayedMessages	High	0.9829	0.9926	0.9481	0.9699	0.9882	0.9956	0.9640	0.9795
	Low	0.9773	0.9871	0.9392	0.9626	0.9845	0.9948	0.9552	0.9746
13-DoS	High	0.9985	0.9982	0.9993	0.9988	0.9993	0.9989	0.9999	0.9994
	Low	0.9972	0.9965	0.9991	0.9978	0.9991	0.9985	1.0000	0.9993
14-DoSRandom	High	0.9989	0.9982	1.0000	0.9991	0.9994	0.9991	1.0000	0.9996
	Low	0.9980	0.9969	1.0000	0.9984	0.9992	0.9986	1.0000	0.9993
15-DoSDisruptive	High	0.9988	0.9980	1.0000	0.9990	0.9994	0.9990	1.0000	0.9995
	Low	0.9980	0.9968	1.0000	0.9984	0.9992	0.9986	1.0000	0.9993
16-GridSybil	High	0.9980	0.9988	0.9993	0.9990	0.9995	0.9992	1.0000	0.9996
	Low	0.9979	0.9977	0.9993	0.9985	0.9993	0.9990	1.0000	0.9995
17-DataReplaySybil	High	0.9876	0.9930	0.9642	0.9784	0.9945	0.9962	0.9856	0.9909
	Low	0.9783	0.9873	0.9434	0.9648	0.9916	0.9950	0.9782	0.9865
18-DoSRandomSybil	High	0.9985	0.9970	1.0000	0.9985	0.9992	0.9985	1.0000	0.9992
	Low	0.9975	0.9952	1.0000	0.9976	0.9989	0.9979	1.0000	0.9989
19-DoSDisruptiveSybil	High	0.9986	0.9971	1.0000	0.9986	0.9992	0.9985	1.0000	0.9992
	Low	0.9975	0.9952	1.0000	0.9976	0.9989	0.9979	1.0000	0.9989
Overall		*0.9755*	*0.9902*	*0.9250*	*0.9505*	***0.9852***	***0.9963***	***0.9540***	***0.9709***

5.5.3 . Online Scenario

For online analysis, we utilize the MixAll dataset, divided into six subsets representing the six coverage areas. These subsets were further partitioned into three parts: training, validation, and testing. The size of the data allocated for training is 72%, enabling the model to learn and generalize patterns from a significant portion of the available samples. The validation set, which forms 8% of a subset, to enable fine-tuning of the model's hyperparameters. Finally, the testing part, which corresponds to the remaining 20% of the data, wasn't exposed to the model during training and validation. It was rather used to assess the performance of the model. Our online testing scenario contains 4544 vehicles: 3176 normal and 1368 misbehaving, sending more than $400,000$ messages.

Table 5.3 compares the performances of the models per area during the online scenario. The Federated model demonstrated similar to slightly better performance compared to the Centralized model, with both models outperforming the Standalone model.

The Federated model significantly enhanced the performance of areas 2 and 6, which have the least amount of training data. This further validates the previous observation made during numerical analysis.

In Table 5.4, we present a summary of the models' detection performances at vehicle-level during the online scenario. Although the Federated model demonstrated superior performance in offline analysis and online window-level results, the Centralized model detected five additional vehicles.

The majority of misbehaving vehicles that were undetected by the models can be attributed to Constant Position Offset and Eventual Stop attacks. The former can be effectively resolved by using the plausibility check we proposed in Chapter 3, or alternatively, by utilizing a map. The latter can be attributed to the labeling methodology employed by the authors of the dataset, which is a persistent problem inherited from the original VeReMi dataset. The authors assert that they have addressed the labeling issue of this attack by individually labeling each message. However, in the publicly available version of the dataset, the labeling is still assigned on a vehicle basis.

Lastly, in the Delayed Messages attack, the Centralized model managed to detect at least one window per vehicle, while missing many others. Such

a result can explain the difference between window-level and vehicle-level detection scores in this particular attack.

In summary, our proposed Federated model demonstrated comparable performance to the Centralized model while offering the advantages of scalability and privacy.

Table 5.3: Online Results: Models Performance Comparison (window-level)

Metric	*Model*	*Area 1*	*Area 2*	*Area 3*	*Area 4*	*Area 5*	*Area 6*	*Overall*
Accuracy	Centralized	0.9817	0.9769	0.9838	0.9838	0.9795	0.9570	09813
	Standalone	0.9640	0.9533	0.9780	0.9553	0.9555	0.9620	0.9619
	Federated	0.9891	0.9816	0.9847	0.9803	0.9833	0.9851	**0.9838**
Precision	Centralized	0.9945	0.9938	0.9929	0.9959	0.9962	0.9613	0.9934
	Standalone	0.9988	0.9860	0.9961	1.0000	0.9905	0.9941	**0.9958**
	Federated	0.9946	0.9888	0.9897	0.9925	0.9992	1.0000	0.9933
Recall	Centralized	0.9609	0.9570	0.9684	0.9656	0.9573	0.9667	0.9631
	Standalone	0.9140	0.9141	0.9513	0.8944	0.9081	0.9417	0.9159
	Federated	0.9789	0.9720	0.9738	0.9606	0.9631	0.9750	**0.9692**
F1-score	Centralized	0.9775	0.9751	0.9805	0.9805	0.9764	0.9640	0.9780
	Standalone	0.9545	0.9487	0.9732	0.9442	0.9475	0.9672	0.9542
	Federated	0.9867	0.9804	0.9817	0.9763	0.9808	0.9873	**0.9811**

Table 5.4: Online Results: Correctly Classified Vehicles

Vehicle Type	*Veh. Count*	*Cent.*	*S.A.*	*Fed.*
0-Normal	3176	3121	3141	3116
1-ConstPos	72	72	68	71
2-ConstPosOffset	64	50	35	41
3-RandomPos	71	71	71	71
4-RandomPosOffset	77	77	16	75
5-ConstSpeed	70	70	68	70
6-ConstSpeedOffset	56	56	56	56
7-RandomSpeed	78	78	78	78
8-RandomSpeedOffset	72	69	49	70
9-EventualStop	81	51	57	57
10-Disruptive	75	75	75	75
11-DataReplay	68	68	64	68
12-DelayedMessages	59	59	59	59
13-DoS	64	64	64	64
14-DoSRandom	78	78	78	78
15-DoSDisruptive	83	83	83	83
16-GridSybil	88	88	88	88
17-DataReplaySybil	61	61	61	61
18-DoSRandomSybil	77	77	77	77
19-DoSDisruptiveSybil	74	74	74	74
All Attacks	1368	1321	1221	1316

5.6 . Conclusion

In this chapter, we proposed a novel approach for protecting V2X application servers for 5G edge networks through the implementation of a distributed V2X misbehavior detection system that relies on Federated Learning. After rigorous testing using a large public dataset, we demonstrated the feasibility and advantages of Federated Learning in V2X misbehavior detection in protecting V2X application servers in 5G core networks.

6 - CONCLUSION AND FUTURE WORK

Contents

6.1 . Conclusion

This thesis encompassed an in-depth investigation of C-ITS and C-V2X architecture and concepts. Moreover, it conducts an in-depth study of 5G NR-V2X, the latest cellular technology for vehicles communications and part of 3GPP Release 16. It includes comprehensive information on the 5G core network and its functions supporting cellular V2X communications. Our research focused on the integration of 5GC and misbehavior detection systems protecting V2X application servers from a variety of attacks that my occur during V2N communications.

The thesis highlights the importance of implementing security measures on V2N. Specifically, we explored the aspects of security in C-ITS and NR-V2X, including: i) the use of machine learning to improve detection results, ii) creating countermeasures to stop detected attacks, iii) leveraging collaboration between detection nodes to improve performance, and iv) leveraging Federated Learning to enhance scalability of the detection system.

Therefore, this thesis provided an in-depth analysis of the related work on security in V2X communications. The manuscript discussed different approaches and algorithms proposed in the literature, pointing out their added-values and drawbacks in 5G V2X scenarios.

Security of Vehicle-to-Everything (V2X) communications is of significant importance due to the possible threats to vehicle networks. This thesis in-

cluded three main contributions:

Firstly, we addressed the pressing requirement to implement and integrate effective misbehavior detection systems that **leverage machine learning** in the recently introduced environment of 5G Vehicle-to-Network (V2N) communications. This is essential **to protect V2X application servers** from common V2X attacks such as position falsification.

Secondly, a noteworthy discovery derived from our second contribution was the added benefit of **collaboration among edge detection nodes**. By enabling the exchange of reputation metrics, such as attack ratio, the precision and effectiveness of misbehavior detection can be improved.

Lastly, another notable finding is the promising use of **Federated Learning** for the deployment of V2X misbehavior detection systems in 5G. By using a decentralized approach to the learning process, we can leverage scalability benefits offered by Federated Learning while still attaining detection performance that is on par with centralized systems. Our exploration of the use of **LSTM** networks has shown their effectiveness in modeling and classifying sequential data, making them an important resource in the domain of V2X misbehavior detection. Additionally, the capacity to grasp extended temporal relationships enables them to proficiently identify complex patterns that may indicate misbehavior.

6.2 . Future Work

While this thesis has shed light on several aspects of V2X security, it also opens the door to numerous avenues for future contributions.

6.2.1 . Short-term

Exploring the integration of the Location Management Function (LMF) of the 5G core network with the proposed 5G V2X misbehavior detection systems will be a crucial area for future research. This integration has the potential to enhance detection accuracy and system efficiency, thanks to the additional location information supplied by the NG-RAN, which can be leveraged to validate the integrity of the received UEs' positions.

Advanced Neural Architectures: Beyond LSTMs, exploring the potential of other neural architectures, such as Transformers, could yield even more

robust detection systems.

Autoencoders for Zero-day Attack Detection: The use of autoencoders presents a promising avenue for detecting zero-day attacks, given their ability to reconstruct input data and identify anomalies. Leveraging their efficacy and scalability in the context of V2X security, could enable the early detection of novel threats.

6.2.2 . Medium-term

Real-world Implementation: Moving from theoretical models to real-world implementations and testing these systems in live vehicular networks will be a crucial step forward. It will be an opportunity to capture real-world delays, and adapt the proposed solutions accordingly.

Scalability of Detection Mechanisms: As vehicular networks grow and become more complex, the scalability of detection systems will be paramount. Exploring scalable architectures and algorithms, especially those that can handle vast networks with minimal latency, will be crucial.

Real-world Attack Datasets: Creating new attack datasets based on recorded traces from real vehicles is very important to optimize the proposed misbehavior detection solutions for real-world scenarios.

6.2.3 . Long-term

3GPP Framework for V2X Misbehavior Detection System: An important milestone will be the standardization of a unified framework for V2X misbehavior detection systems in upcoming 3GPP releases.

Misbehavior Authority and 5G: The integration and standardization of 5G V2N misbehavior detection solutions with the V2V Misbehavior Authority defined by ETSI can be an important step to eliminate vehicles misbehaving on V2N from V2V networks as well.

Bibliography

[1] WHO, *Death on the roads*, Accessed: 2023-09-25. [Online]. Available: https://extranet.who.int/roadsafety/death-on-the-roads/.

[2] FCC, *Fcc modernizes 5.9 ghz band for wi-fi and auto safety*, Accessed: 2023-09-25. [Online]. Available: https://docs.fcc.gov/public/attachments/DOC-368228A1.pdf.

[3] "Ieee standard for information technology– local and metropolitan area networks– specific requirements– part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications amendment 6: Wireless access in vehicular environments," *IEEE Std 802.11p-2010 (Amendment to IEEE Std 802.11-2007 as amended by IEEE Std 802.11k-2008, IEEE Std 802.11r-2008, IEEE Std 802.11y-2008, IEEE Std 802.11n-2009, and IEEE Std 802.11w-2009)*, pp. 1–51, 2010. doi: 10.1109/IEEESTD.2010.5514475.

[4] "Ieee trial-use standard for wireless access in vehicular environments - security services for applications and management messages," *IEEE Std 1609.2-2006*, pp. 1–116, 2006. doi: 10.1109/IEEESTD.2006.8691892.

[5] "Ieee trial-use standard for wireless access in vehicular environments (wave) - networking services," *IEEE Std 1609.3-2007*, pp. 1–99, 2007. doi: 10.1109/IEEESTD.2007.353212.

[6] "Ieee trial-use standard for wireless access in vehicular environments (wave) - multi-channel operation," *IEEE Std 1609.4-2006*, pp. 1–82, 2006. doi: 10.1109/IEEESTD.2006.254109.

[7] I. 8. bd, *802.11bd-2022 - ieee standard for information technology–telecommunications and information exchange between systems local and metropolitan area networks–specific requirements part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications amendment 5: Enhancements for next genera-*

tion v2x. [Online]. Available: https://ieeexplore.ieee.org/document/10063942.

[8] IEEE, *Ieee p802.11—next generation v2x study group accessed: May 30, 2019.* [Online]. Available: http://www.ieee802.org/11/Reports/tgbd_update.htm.

[9] ETSI, "Intelligent transport systems (its); its-g5 access layer specification for intelligent transport systems operating in the 5 ghz frequency band," European Telecommunications Standards Institute (ETSI), EN 302 663, Jan. 2020, V1.3.1. [Online]. Available: https://www.etsi.org/deliver/etsi_en/302600_302699/302663/01.03.01_60/en_302663v010301p.pdf.

[10] ETSI, "Intelligent transport systems (its); communications architecture," European Telecommunications Standards Institute (ETSI), EN 302 665, Sep. 2010, V1.1.1. [Online]. Available: https://www.etsi.org/deliver/etsi_en/302600_302699/302665/01.01.01_60/en_302665v010101p.pdf.

[11] J. Petit, F. Schaub, M. Feiri, and F. Kargl, "Pseudonym schemes in vehicular networks: A survey," *IEEE Communications Surveys Tutorials*, vol. 17, no. 1, pp. 228–255, 2015. doi: 10.1109/COMST.2014.2345420.

[12] R. W. van der Heijden, S. Dietzel, T. Leinmüller, and F. Kargl, "Survey on misbehavior detection in cooperative intelligent transportation systems," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 1, pp. 779–811, 2019. doi: 10.1109/COMST.2018.2873088.

[13] SARWS, *Sarws project*, Accessed: 2023-09-25. [Online]. Available: https://sarws.eu.

[14] ETSI, "Intelligent transport systems (its); users and applications requirements; part 1: Facility layer structure, functional requirements and specifications," *ETSI*, TS 102 894-1 V1.1.1 (2013-08). [Online]. Available: https://www.etsi.org/deliver/etsi_ts/102800_102899/10289401/01.01.01_60/ts_10289401v010101p.pdf.

[15] ISO, “Information technology — open systems interconnection — basic reference model: The basic model,” ISO/IEC 7498-1:1994. [Online]. Available: https://www.iso.org/standard/20269.html.

[16] ETSI, “Intelligent transport systems (its); vehicular communications; basic set of applications; definitions,” European Telecommunications Standards Institute (ETSI), TR 102 638, Jun. 2009, V1.1.1.

[17] ETSI, “Intelligent transport systems (its); vehicular communications; basic set of applications; local dynamic map (ldm),” European Telecommunications Standards Institute (ETSI), EN 302 895, Sep. 2014, V1.1.1. [Online]. Available: https://www.etsi.org/deliver/etsi_en/302800_302899/302895/01.01.01_60/en_302895v010101p.pdf.

[18] ISO, “Intelligent transport systems — Co-operative ITS — Local dynamic map,” Tech. Rep. 18750, May 2018. [Online]. Available: https://www.iso.org/standard/69433.html.

[19] ETSI, “Intelligent transport systems (its); vehicular communications; basic set of applications; part 2: Specification of cooperative awareness basic service,” European Telecommunications Standards Institute (ETSI), EN 302 637-2, Sep. 2014, V1.3.1. [Online]. Available: https://www.etsi.org/deliver/etsi_en/302600_302699/30263702/01.03.01_30/en_30263702v010301v.pdf.

[20] ETSI, “Intelligent transport systems (its); vehicular communications; basic set of applications; part 3: Specifications of decentralized environmental notification basic service,” European Telecommunications Standards Institute (ETSI), EN 302 637-3, Sep. 2014, V1.2.1. [Online]. Available: https://www.etsi.org/deliver/etsi_en/302600_302699/30263703/01.02.01_30/en_30263703v010201v.pdf.

[21] E. ETSI, “Intelligent transport systems (its); vehicular communications; geonetworking; part 5: Transport protocols; sub-part 1: Basic transport protocol,” *Intelligent Transport Systems (ITS)*, 302

636-5-1 V2. 1.0 (2017–05). [Online]. Available: https://www.etsi.org/deliver/etsi_en/302600_302699/3026360501/02.01.00_20/en_3026360501v020100a.pdf.

[22] E. ETSI, "Intelligent transport systems (its); vehicular communications; geonetworking; part 4: Geographical addressing and forwarding for point-to-point and point-to-multipoint communications; sub-part 1: Media-independent functionality," *ETSI*, 302 636-4-1 V1.3.1 (2017-08). [Online]. Available: https://www.etsi.org/deliver/etsi_en/302600_302699/3026360401/01.03.01_60/en_3026360401v010301p.pdf.

[23] *Transmission Control Protocol*, RFC 793, Sep. 1981. doi: 10.17487/RFC0793. [Online]. Available: https://www.rfc-editor.org/info/rfc793.

[24] *User Datagram Protocol*, RFC 768, Aug. 1980. doi: 10.17487/RFC0768. [Online]. Available: https://www.rfc-editor.org/info/rfc768.

[25] S. Deering and R. Hinden, *Rfc 8200: Internet protocol, version 6 (ipv6) specification*, 2017. [Online]. Available: https://www.ietf.org/rfc/rfc2460.txt.

[26] ETSI, "Intelligent transport systems (its); lte-v2x access layer specification for intelligent transport systems operating in the 5 ghz frequency band," European Telecommunications Standards Institute (ETSI), EN 303 613, Jan. 2020, V1.1.1. [Online]. Available: https://www.etsi.org/deliver/etsi_en/303600_303699/303613/01.01.01_60/en_303613v010101p.pdf.

[27] ISO, "Intelligent transport systems — ITS station security services for secure session establishment and authentication between trusted devices," Tech. Rep. 21177, Aug. 2019. [Online]. Available: https://www.iso.org/standard/70056.html.

[28] ETSI, "Intelligent transport systems (its); security; security header and certificate formats; release 2," European Telecommunications Standards Institute (ETSI), TS 103 097, Oct. 2021, V2.1.1. [Online]. Available: https://www.etsi.org/deliver/etsi_ts/

103000_103099/103097/02.01.01_60/ts_103097v020101p.pdf.

[29] IEEE, *Ieee standard for wireless access in vehicular environments–security services for application and management messages*, IEEE 1609.2, Dec. 2022. [Online]. Available: https://standards.ieee.org/standard/1609_2-2022.html.

[30] M. Msahli, N. Cam-Winget, W. Whyte, A. Serhrouchni, and H. Labiod, *TLS Authentication Using Intelligent Transport System (ITS) Certificates*, RFC 8902, Sep. 2020. doi: 10.17487/RFC8902. [Online]. Available: https://rfc-editor.org/rfc/rfc8902.txt.

[31] ETSI, "Intelligent transport system (its); security; its communications security architecture and security management; release 2," European Telecommunications Standards Institute (ETSI), TS 102 940, Jul. 2021, V2.1.1. [Online]. Available: https://www.etsi.org/deliver/etsi_ts/102900_102999/102940/01.03.01_60/ts_102940v010301p.pdf.

[32] ETSI, "Intelligent transport systems (its); security; trust and privacy management; release 2," European Telecommunications Standards Institute (ETSI), TS 102 941, Nov. 2022, V2.2.1. [Online]. Available: https://www.etsi.org/deliver/etsi_ts/102900_102999/102941/02.02.01_60/ts_102941v020201p.pdf.

[33] 3rd Generation Partnership Project (3GPP), "Nr; user equipment (ue) radio transmission and reception; part 1: Range 1 standalone," 3rd Generation Partnership Project (3GPP), Technical specification (TS) 38.101-1, Oct. 2021, v16.9.0. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/38_series/38.101-1/.

[34] 3rd Generation Partnership Project (3GPP), "Overall description of radio access network (ran) aspects for vehicle-to-everything (v2x) based on lte and nr," 3rd Generation Partnership Project (3GPP), Technical report (TR) 37.985, Apr. 2022, v16.1.0. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/37_series/37.985/.

[35] 3rd Generation Partnership Project (3GPP), "Evolved universal terrestrial radio access (e-utra) and nr; service data adaptation protocol (sdap) specification," 3rd Generation Partnership Project (3GPP), Technical specification (TS) 37.324, Jul. 2021, v16.3.0. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/37_series/37.324/.

[36] 3rd Generation Partnership Project (3GPP), "Nr; packet data convergence protocol (pdcp) specification," 3rd Generation Partnership Project (3GPP), Technical specification (TS) 38.323, Mar. 2021, v16.3.0. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/38_series/38.323/.

[37] 3rd Generation Partnership Project (3GPP), "Nr; radio link control (rlc) protocol specification," 3rd Generation Partnership Project (3GPP), Technical specification (TS) 38.322, Jul. 2022, v16.3.0. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/38_series/38.322/.

[38] 3rd Generation Partnership Project (3GPP), "Nr; medium access control (mac) protocol specification," 3rd Generation Partnership Project (3GPP), Technical specification (TS) 38.321, Apr. 2022, v16.8.0. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/38_series/38.321/.

[39] 3rd Generation Partnership Project (3GPP), "Nr; physical layer; general description," 3rd Generation Partnership Project (3GPP), Technical specification (TS) 38.201, Jan. 2020, v16.0.0. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/38_series/38.201/.

[40] 3rd Generation Partnership Project (3GPP), "Nr; services provided by the physical layer," 3rd Generation Partnership Project (3GPP), Technical specification (TS) 38.202, Jan. 2022, v16.3.0. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/38_series/38.202/.

[41] 3rd Generation Partnership Project (3GPP), "Nr; physical channels and modulation," 3rd Generation Partnership Project (3GPP), Technical specification (TS) 38.211, Jan. 2022, v16.8.0. [Online].

Available: https://www.3gpp.org/ftp/Specs/archive/38_series/38.211/.

[42] 3rd Generation Partnership Project (3GPP), "Nr; multiplexing and channel coding," 3rd Generation Partnership Project (3GPP), Technical specification (TS) 38.212, Jan. 2022, v16.8.0. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/38_series/38.212/.

[43] 3rd Generation Partnership Project (3GPP), "Nr; physical layer procedures for control," 3rd Generation Partnership Project (3GPP), Technical specification (TS) 38.213, Jan. 2022, v16.8.0. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/38_series/38.213/.

[44] 3rd Generation Partnership Project (3GPP), "Nr; physical layer procedures for data," 3rd Generation Partnership Project (3GPP), Technical specification (TS) 38.214, Jan. 2020, v16.0.0. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/38_series/38.214/.

[45] 3rd Generation Partnership Project (3GPP), "Nr; physical layer measurements," 3rd Generation Partnership Project (3GPP), Technical specification (TS) 38.215, Apr. 2022, v16.5.0. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/38_series/38.215/.

[46] 3rd Generation Partnership Project (3GPP), "Release 14 description; summary of rel-14 work items," 3rd Generation Partnership Project (3GPP), Technical report (TR) 21.914, Jun. 2018, v14.0.0. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/21_series/21.914/.

[47] 3rd Generation Partnership Project (3GPP), "Architecture enhancements for v2x services," 3rd Generation Partnership Project (3GPP), Technical specification (TS) 23.285, Sep. 2016, v14.0.0. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/23_series/23.285/.

[48] 3rd Generation Partnership Project (3GPP), "Proximity-based services (prose); stage 2," 3rd Generation Partnership Project (3GPP), Technical specification (TS) 23.303, Dec. 2016, v13.6.0. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/23_series/23.303/.

[49] 3rd Generation Partnership Project (3GPP), "Technical specifications and technical reports for a utran-based 3gpp system," 3rd Generation Partnership Project (3GPP), Technical specification (TS) 21.101, Mar. 2016, v12.0.0. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/21_series/21.101/.

[50] 3rd Generation Partnership Project (3GPP), "Release 15 description; summary of rel-15 work items," 3rd Generation Partnership Project (3GPP), Technical report (TR) 21.915, Oct. 2019, v15.0.0. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/21_series/21.915/.

[51] 3rd Generation Partnership Project (3GPP), "System architecture for the 5g system (5gs)," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 23.501, Dec. 2017, v15.0.0. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/23_series/23.501/.

[52] 3rd Generation Partnership Project (3GPP), "Procedures for the 5g system (5gs)," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 23.502, Dec. 2017, v15.0.0. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/23_series/23.502/.

[53] 3rd Generation Partnership Project (3GPP), "Policy and charging control framework for the 5g system (5gs); stage 2," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 23.503, Dec. 2017, v15.0.0. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/23_series/23.503/.

[54] 3rd Generation Partnership Project (3GPP), "Nr; user equipment (ue) radio transmission and reception; part 2: Range 2 standalone," 3rd Generation Partnership Project (3GPP), Technical specification (TS) 38.101-2, Apr. 2021, v16.7.0. [Online]. Avail-

able: https://www.3gpp.org/ftp/Specs/archive/38_series/38.101-2/.

[55] 3rd Generation Partnership Project (3GPP), "Release 16 description; summary of rel-16 work items," 3rd Generation Partnership Project (3GPP), Technical report (TR) 21.916, Jun. 2021, v16.0.0. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/21_series/21.916/.

[56] 3rd Generation Partnership Project (3GPP), "Architecture enhancements for 5g system (5gs) to support vehicle-to-everything (v2x) services," 3rd Generation Partnership Project (3GPP), Technical specification (TS) 23.287, Sep. 2019, v16.0.0. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/23_series/23.287/.

[57] 3rd Generation Partnership Project (3GPP), "5g system; access and mobility management services; stage 3," 3rd Generation Partnership Project (3GPP), Technical specification (TS) 29.518, Dec. 2021, v16.1.0. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/29_series/29.518/.

[58] 3rd Generation Partnership Project (3GPP), "5g system; session management services; stage 3," 3rd Generation Partnership Project (3GPP), Technical specification (TS) 29.502, Dec. 2021, v16.1.0. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/29_series/29.502/.

[59] 3rd Generation Partnership Project (3GPP), "5g system; policy and charging control signalling flows and qos parameter mapping; stage 3," 3rd Generation Partnership Project (3GPP), Technical specification (TS) 29.513, Jun. 2022, v16.1.0. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/29_series/29.513/.

[60] 3rd Generation Partnership Project (3GPP), "5g system; authentication server services; stage 3," 3rd Generation Partnership Project (3GPP), Technical specification (TS) 29.509, Jun. 2022, v16.1.0. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/29_series/29.509/.

[61] 3rd Generation Partnership Project (3GPP), "5g system; unified data management services; stage 3," 3rd Generation Partnership Project (3GPP), Technical specification (TS) 29.503, Dec. 2021, v16.1.0. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/29_series/29.503/.

[62] 3rd Generation Partnership Project (3GPP), "5g system; unified data repository services; stage 3," 3rd Generation Partnership Project (3GPP), Technical specification (TS) 29.504, Dec. 2021, v16.1.0. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/29_series/29.504/.

[63] 3rd Generation Partnership Project (3GPP), "5g system; network data analytics services; stage 3," 3rd Generation Partnership Project (3GPP), Technical specification (TS) 29.520, Feb. 2022, v16.10.0. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/29_series/29.520/.

[64] 3rd Generation Partnership Project (3GPP), "5g system; network slice selection services; stage 3," 3rd Generation Partnership Project (3GPP), Technical specification (TS) 29.531, Sep. 2021, v16.8.0. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/29_series/29.531/.

[65] 3rd Generation Partnership Project (3GPP), "5g system; network function repository services; stage 3," 3rd Generation Partnership Project (3GPP), Technical specification (TS) 29.510, Dec. 2021, v16.10.0. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/29_series/29.510/.

[66] 3rd Generation Partnership Project (3GPP), "5g system; location management services; stage 3," 3rd Generation Partnership Project (3GPP), Technical specification (TS) 29.572, Sep. 2021, v16.8.0. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/29_series/29.572/.

[67] 3rd Generation Partnership Project (3GPP), "5g system; network exposure function northbound apis; stage 3," 3rd Generation Partnership Project (3GPP), Technical specification (TS) 29.522,

Mar. 2022, v16.10.0. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/29_series/29.522/.

[68] 3rd Generation Partnership Project (3GPP), "5g system; network exposure function southbound services; stage 3," 3rd Generation Partnership Project (3GPP), Technical specification (TS) 29.591, Sep. 2021, v16.8.0. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/29_series/29.591/.

[69] 3rd Generation Partnership Project (3GPP), "5g system; user plane function services; stage 3," 3rd Generation Partnership Project (3GPP), Technical specification (TS) 29.564, Jun. 2022, v17.1.0. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/29_series/29.564/.

[70] 3rd Generation Partnership Project (3GPP), "System architecture evolution (sae); security architecture (release 14)," 3rd Generation Partnership Project (3GPP), Technical specification (TS) 33.401, Sep. 2016, v14.0.0. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/33_series/33.401/.

[71] 3rd Generation Partnership Project (3GPP), "Security aspect for lte support of vehicle-to-everything (v2x) services (release 14)," 3rd Generation Partnership Project (3GPP), Technical specification (TS) 33.185, Jun. 2017, v14.0.0. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/33_series/33.185/.

[72] F. Nilofer and J. Qaddour, "Comparative study of vulnerabilities in lte cryptographic algorithm," *International Journal of Computer Applications*, vol. 180, no. 25, pp. 19–25, Mar. 2018, issn: 0975-8887. doi: 10.5120/ijca2018916587. [Online]. Available: http://www.ijcaonline.org/archives/volume180/number25/29112-2018916587.

[73] 3rd Generation Partnership Project (3GPP), "Security architecture and procedures for 5g system (release 16)," 3rd Generation Partnership Project (3GPP), Technical specification (TS) 33.501, Sep. 2019, v16.0.0. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/33_series/33.501/.

[74] 3rd Generation Partnership Project (3GPP), "Security aspects of 3gpp support for advanced vehicle-to-everything (v2x) services (release 16)," 3rd Generation Partnership Project (3GPP), Technical specification (TS) 33.536, Jul. 2020, v16.0.0. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/33_series/33.536/.

[75] A. Ghosal and M. Conti, "Security issues and challenges in V2X: A Survey," en, *Computer Networks*, vol. 169, p. 107 093, Mar. 2020, issn: 13891286. doi: 10.1016/j.comnet.2019.107093. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S1389128619305857 (visited on 10/10/2021).

[76] R. Lu, L. Zhang, J. Ni, and Y. Fang, "5G Vehicle-to-Everything Services: Gearing Up for Security and Privacy," en, *Proceedings of the IEEE*, vol. 108, no. 2, pp. 373–389, Feb. 2020, issn: 0018-9219, 1558-2256. doi: 10.1109/JPROC.2019.2948302. [Online]. Available: https://ieeexplore.ieee.org/document/8897696/ (visited on 10/10/2021).

[77] V. Sharma, I. You, and N. Guizani, "Security of 5g-v2x: Technologies, standardization, and research directions," *IEEE Network*, vol. 34, no. 5, pp. 306–314, 2020. doi: 10.1109/MNET.001.1900662.

[78] L. Codeca, R. Frank, and T. Engel, "Luxembourg sumo traffic (lust) scenario: 24 hours of mobility for vehicular networking research," in *2015 IEEE Vehicular Networking Conference (VNC)*, 2015, pp. 1–8. doi: 10.1109/VNC.2015.7385539.

[79] K. F. van der Heijden R.W. Lukaseder T., "Veremi: A dataset for comparable evaluation of misbehavior detection in vanets," *Chang B., Li Y., Zhu S. (eds) Security and Privacy in Communication Networks. SecureComm 2018*, vol. 254, 2018. doi: https://doi.org/10.1007/978-3-030-01701-9_18.

[80] P. A. Lopez *et al.*, "Microscopic traffic simulation using sumo," in *The 21st IEEE International Conference on Intelligent Transportation Systems*, IEEE, 2018. [Online]. Available: https://elib.dlr.de/124092/.

[81] C. Sommer, R. German, and F. Dressler, "Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis," *IEEE Transactions on Mobile Computing (TMC)*, vol. 10, no. 1, pp. 3–15, Jan. 2011. doi: 10.1109/TMC.2010.133.

[82] J. Kamel, M. Ansari, J. Petit, A. Kaiser, I. Ben Jemaa, and P. Urien, "Simulation framework for misbehavior detection in vehicular networks," *IEEE Transactions on Vehicular Technology*, vol. PP, Apr. 2020. doi: 10.1109/TVT.2020.2984878.

[83] S. So, P. Sharma, and J. Petit, "Integrating Plausibility Checks and Machine Learning for Misbehavior Detection in VANET," en, in *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, Orlando, FL: IEEE, Dec. 2018, pp. 564–571, isbn: 978-1-5386-6805-4. doi: 10.1109/ICMLA.2018.00091. [Online]. Available: https://ieeexplore.ieee.org/document/8614116/ (visited on 10/10/2021).

[84] P. Sharma and H. Liu, "A Machine-Learning-Based Data-Centric Misbehavior Detection Model for Internet of Vehicles," en, *IEEE Internet of Things Journal*, vol. 8, no. 6, pp. 4991–4999, Mar. 2021, issn: 2327-4662, 2372-2541. doi: 10.1109/JIOT.2020.3035035. [Online]. Available: https://ieeexplore.ieee.org/document/9245568/ (visited on 10/10/2021).

[85] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in python," *Journal of machine learning research*, vol. 12, no. Oct, pp. 2825–2830, 2011.

[86] N. Bißmeyer, J. Njeukam, J. Petit, and K. M. Bayarou, "Central misbehavior evaluation for vanets based on mobility data plausibility," ser. VANET '12, Low Wood Bay, Lake District, UK: Association for Computing Machinery, 2012, pp. 73–82, isbn: 9781450313179. doi: 10.1145/2307888.2307902. [Online]. Available: https://doi.org/10.1145/2307888.2307902.

[87] S. Gyawali and Y. Qian, "Misbehavior detection using machine learning in vehicular communication networks," in *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, 2019, pp. 1–6. doi: 10.1109/ICC.2019.8761300.

[88] C. Kim, S.-Y. Chang, D. Lee, J. Kim, K. Park, and J. Kim, "Reliable detection of location spoofing and variation attacks," *IEEE Access*, vol. 11, pp. 10 813–10 825, 2023. doi: 10.1109/ACCESS.2023.3241236.

[89] P. Lv, L. Xie, J. Xu, X. Wu, and T. Li, "Misbehavior detection in vehicular ad hoc networks based on privacy-preserving federated learning and blockchain," *IEEE Transactions on Network and Service Management*, vol. 19, no. 4, pp. 3936–3948, 2022. doi: 10.1109/TNSM.2022.3220779.

[90] J. Kamel, M. Wolf, R. W. van der Hei, A. Kaiser, P. Urien, and F. Kargl, "Veremi extension: A dataset for comparable evaluation of misbehavior detection in vanets," in *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, 2020, pp. 1–6. doi: 10.1109/ICC40277.2020.9149132.

[91] T. Alladi, V. Kohli, V. Chamola, and F. R. Yu, "A deep learning based misbehavior classification scheme for intrusion detection in cooperative intelligent transportation systems," *Digital Communications and Networks*, 2022, issn: 2352-8648. doi: https://doi.org/10.1016/j.dcan.2022.06.018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2352864822001407.

[92] R. Sedar, C. Kalalas, F. Vázquez-Gallego, and J. Alonso-Zarate, "Reinforcement learning based misbehavior detection in vehicular networks," in *ICC 2022 - IEEE International Conference on Communications*, 2022, pp. 3550–3555. doi: 10.1109/ICC45855.2022.9838796.

[93] 3GPP, "Spending Limit Control Service; Stage 3 (Release 15)," TS 29.594, Jun. 2018.

[94] EstiNet, https://www.estinet.com/ns/.

[95] free5GC Project, https://www.free5gc.org/.

[96] R. Akbani, T. Korkmaz, and G. V. S. Raju, "A machine learning based reputation system for defending against malicious node behavior," in *IEEE GLOBECOM 2008 - 2008 IEEE Global Telecommu-*

nications Conference, 2008, pp. 1–5. doi: 10.1109/GLOCOM.2008.ECP.408.

[97] UERANSIM, https://github.com/aligungr/UERANSIM.

[98] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997. doi: 10.1162/neco.1997.9.8.1735.

[99] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y. Arcas, "Communication-Efficient Learning of Deep Networks from Decentralized Data," in *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, A. Singh and J. Zhu, Eds., ser. Proceedings of Machine Learning Research, vol. 54, PMLR, Apr. 2017, pp. 1273–1282. [Online]. Available: https://proceedings.mlr.press/v54/mcmahan17a.html.

[100] Martín Abadi *et al.*, *TensorFlow: Large-scale machine learning on heterogeneous systems*, Software available from tensorflow.org, 2015. [Online]. Available: https://www.tensorflow.org/.