Efficient Authentication of Vehicle-to-Vehicle (V2V) Safety Messages

A thesis submitted in partial fulfilment of the requirements of Birmingham City University for the degree of

Doctor of Philosophy

By

Mujahid Muhammad

School of Computing and Digital Technology,

Faculty of Computing, Engineering and the Built Environment

January 2024

Declaration

I hereby declare that this thesis, titled 'Efficient Authentication of Vehicle-to-Vehicle (V2V) Safety Messages,' submitted by me for the award of the degree of Doctor of Philosophy to Birmingham City University, is work carried out by me under the supervision of Dr. Junaid Arshad (current Director of Studies), Prof. Paul Kearney (previous Director of Studies), and Prof. Adel Aneiba.

I further declare that the work presented in this thesis has not been submitted and will not be submitted, either in part or in full, for the award of any other degree in any other University or Institute

Place: Birmingham, UK

Signature of the Candidate: 

Date: 12/01/2024

Abstract

Cooperative Intelligent Transport Systems (C-ITS) extend traffic awareness beyond individual vehicles by enabling the exchange of information through messages shared among vehicles and roadside units (RSU) via direct vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) broadcast communications. This shared information is processed by safety applications, such as intersection collision warning systems, designed to detect and mitigate potentially dangerous situations. The aim is either to alert the vehicle driver or to take corrective action through the vehicle's control system. Ensuring the authenticity and integrity of these messages is crucial, as acting on incorrect information could put human safety at risk. However, securing direct V2V/I safety messages poses challenges due to the strict performance requirements of safety applications (particularly low end-to-end message latency and reliable message delivery), the dynamic topology and large scale of the peer-to-peer network, and the high volume of message traffic that vehicles are expected to process.

Standards organisations and government transportation authorities recommend the use of digital signature schemes based on public key cryptography to authenticate and protect the integrity of messages. These require a vehicular public key infrastructure (VPKI) to manage certificates and deliver them to vehicles. The first research contribution of this thesis is to confirm, through theoretical modelling and realistic simulation, that performance issues inherent in the VPKI-based scheme, owing to the computational expense of asymmetric cryptography, result in message latency and dropped message frequency that exceed acceptable limits in moderate to high road traffic density situations. As an alternative, an approach derived from a symmetric cryptography-based protocol called Time Efficient Stream Loss-tolerant Authentication (TESLA) is proposed. We show that its performance is within the requirements of basic safety applications. Furthermore, it requires less infrastructure and administration.

Applying standard TESLA in the context of V2V, has its own challenges. One is the difficulty of distributing authentication information called commitments in the dynamic V2V environment. We identify two solutions to this problem (VAS-centric and Vehicle-centric) and show, through analysis and simulation, that up to 94% of commitments delivered to vehicles by the VAS are timely. The VAS-centric solution is preferred, except in areas where cellular

network coverage is poor. A second challenge is the so-called authentication delay, a fixed latency overhead inherent to TESLA, found to be at least 12ms in the V2V context, and which results in poorer performance than the VPKI-based approach when message traffic is light. To address this, we propose a modified version of standard TESLA, called prompt verification (PV), which eliminates the authentication delay. Unfortunately, this is vulnerable to impersonation attacks in some circumstances. We devise and study a number of mitigations for this vulnerability, notably a method (RMCCS) for detecting inconsistencies between reported and actual vehicle positions that is based on the physical characteristics of transmission signals. This is employed in a hybrid approach (PV+TESLA), in which a selected minority of messages are subject to standard TESLA verification. Information from RMCCS and from other sources are factors influencing PV+TESLA's decision on how many and which messages to verify with TESLA In experiments, PV+TESLA resulted in an 85% reduction in authentication delay, although there is a trade-off between reducing latency and increasing risk of accepting fake messages. RMCCS can also be applied independently to detect vehicles giving incorrect position information.

List of Publications

i. **Muhammad, M**., Kearney, P., Aneiba, A. and Kunz, A., 2019. Analysis of security overhead in broadcast V2V communications. In International Conference on Computer Safety, Reliability, and Security (pp. 251-263). Springer.

ii. **Muhammad, M**., Kearney, P., Aneiba, A. and Kunz, A., 2020. Efficient Distribution of Key Chain Commitments for Broadcast Authentication in V2V Communications. In 2020 IEEE 92nd Vehicular Technology Conference (VTC2020-Fall) (pp. 1-6). IEEE.

iii. **Muhammad, M**., Kearney, P., Aneiba, A., Arshad, J. and Kunz, A., 2021. RMCCS: RSSI-based Message Consistency Checking Scheme for V2V Communications. In Proceedings of the 18th International Conference on Security and Cryptography (SECRYPT 2021), (pp 722-727). SCITEPRESS.

Acknowledgment

The first and foremost gratitude is to Almighty God who gave me the opportunity and determination to undertake this research. His continuous blessings and guidance have always relieved me during this challenging journey.

My deepest appreciation goes to my previous Director of Studies (DoS) Prof. Paul Kearney for his immense contributions in focusing the scope of the research work, paying close attention to details, and insightful discussions. He has guided me throughout the research work and dedicated a lot of his time to support it.

I would like to express my appreciation to my current DoS, Dr. Junaid Arshad, and my second supervisor, Prof. Adel Aneiba, for their academic support, contributions, and continuous guidance.

I am also grateful to Prof. Ali Abdallah who initially facilitated my movement to Birmingham City University to start the PhD.

I appreciate Dr Andreas Kunz from *Lenovo* in *Oberursel, Germany*. This study has benefitted greatly from his industry experience and involvement in standardisation activities in the cellular-based V2X domain.

A special thanks goes to the Faculty of Computing, Engineering and the Built Environment (CEBE), which provided all necessary resources to complete this research, particularly the tuition fee waiver that was granted to me throughout my studentship. Among the CEBE faculty staff, I would like to express my gratitude to Prof. Mak Sharma and Sue Witton, for their usual encouragement, advice, and assistance.

To my family and friends back home, especially my parents, thank you all for the support and encouragement.

Table of Contents

List of Figures

List of Tables

Abbreviations

3GPP	3rd Generation Partnership Project
BSM	Basic Safety Messages
CA	Cooperative Awareness
CAM	cooperative awareness messages
CDT	Commitment Distribution Table
C-ITS	Cooperative Intelligent Transport Systems
C-V2X	Cellular-V2X
DENM	Decentralised Environment Notification messages
DSRC	Dedicated Short-Range Communication
DoS	Denial of Service
ECDSA	Elliptic Curve Digital Signature Algorithm
ETSI	European Telecommunications Standard Institute
ICRW	Intersection collision risk warning
IEEE	Institute of Electrical and Electronic Engineers
ITS	Intelligent Transport System
LDM	Local Dynamic Map
LDPLM	log-distance path loss model
LCRW	longitudinal collision risk warning
LoS	Line-of-sight
MM	Multiple impersonated vehicles, multiple fake messages
MS	multiple impersonated vehicles, single fake message
NloS	Non line-of-sight

OBU	Onboard unit
PDR	Packet Delivery Ratio
PKI	Public key infrastructure
PV	Prompt Verification
PV+TESLA	PV with TESLA as a backup
RC	Research Contribution
RMCCS	RSSI-based Message Consistency Checking Scheme
RBs	Resource Blocks
RSU	Roadside Infrastructure Units
RSS	Received Signal Strength Indicator
SAE	Society of Automotive Engineers
SM	Single impersonated vehicle, multiple fake messages
SS	Single impersonated vehicle, single fake message
SPS	semi-persistent scheduling
SUMO	Simulation for urban mobility
TESLA	Time Efficient Stream Loss-tolerant Authentication
TTC	Time to Collision
USDOT	United State Department of Transportation
V2V	Vehicle-to-Vehicle
V2I	Vehicle-to-Infrastructure
V2N	Vehicle-to-Network
V2X	Vehicle-to-Everything
V2P	Vehicle to Pedestrian

VAS	V2X Application Server
VCF	V2X Control Function
VPKI	Vehicular Public Key Infrastructure
WAVE	Wireless Access in Vehicular Environment

Chapter 1: Introduction

1.1 Vehicle to Everything (V2X) Communication

The desire to enhance road transport efficiency and safety, coupled with a reduction in accidents due to human error, is driving the automotive industry to deploy innovative solutions leveraging technological advances. Modern vehicles are equipped with various smart devices that sense and process information from their surroundings, enabling automated driving functions and assisting drivers in making safer decisions. For example, avoid making unsafe lane changes or drifting into adjacent lanes.

These advancements, however, highlight the need to extend perceptual bounds beyond individual vehicles. Information exchange through direct vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communications allows vehicles to cooperate, improving overall situational awareness. This digital connectivity, forming a Cooperative Intelligent Transport System (C-ITS), is expected to significantly enhance road user experiences by improving driving safety, increasing road transport efficiency, and reducing environmental impact.

Within a C-ITS, vehicles regularly broadcast traffic-related messages containing information on position, time, speed, etc. In addition, event-triggered messages are broadcast when an abnormal situation, such as emergency braking or accidents, occur. These messages are the basis for safety and traffic efficiency applications, including collision avoidance warnings, safe intersection, and vehicle platooning. [8][9][10]. In particular, the safety applications envisioned to enhance safety on roads have stringent performance requirements, with some use-cases demanding ultra-reliable communication links and a maximum end-to-end delay of 100ms or less (as little as 3ms in some advanced use cases, e.g. cooperative sensing). For this, two main approaches to short-range, broadcast message transport communications have been adopted. One is based on Wi-Fi, with the IEEE 802.11p standard forming the basis for Dedicated Short-Range Communication (DSRC) in the US [11] and ITS-G5 (being standardised by ETSI) in the European C-ITS initiative [12]. The second is based on the Cellular-V2X (C-V2X) standard defined by the 3GPP as part of its LTE and ongoing 5G-new radio (5G-NR) families of standards [13].

1.2 Security in V2V/V2I

The level of efficacy and reliability required in a system where information is shared among independent mobile entities raises concerns about the trustworthiness and timeliness of the received information. This is particularly true for direct V2V/V2I, where shared messages may contain information that could jeopardise. Malicious agents could disseminate false information or prevent the processing of received messages to cause accidents, disrupt traffic or otherwise manipulate other drivers and vehicles in pursuit of their own goals. Thus, security is critical for the operation and wide acceptability of direct V2V/V2I applications and services. A security mechanism should, at least, enable a receiving vehicle to (a) authenticate the source of a received message, and (b) verify the integrity of the received message. While similar security concerns exist in other wireless communication networks, direct V2V/V2I broadcast communications pose unique challenges due to the mobility of vehicles, frequent dynamic topology change, and latency constraints of safety applications.

Cryptography is widely used to implement a variety of security properties including authentication, data integrity, and confidentiality. In the domain of direct V2V/V2I broadcast messaging, both the DSRC and ETSI ITS-G5 standards recommend the use of IEEE 1609.2 [14], which is based on asymmetric digital signatures and a Vehicular Public Key Infrastructure (VPKI). In this, the sender signs an outgoing message with its private key, and the receivers of the message verify it using the corresponding public key. The benefits of the VPKI-based approach are accompanied by some penalties e.g., increase in message verification time due to high computational cost of asymmetric operation, additional bandwidth consumption by adding an average of 108 bytes per message [15], etc. Previous research works (e.g. [16] [17]) have shown that in conditions of high traffic density (around 200 vehicles per km^2), these issues cause messages to be delayed excessively up to 500ms or dropped. As reported in Chapter 3, we have confirmed this claim and further investigated the issue. Delays and dropped messages reduce vehicles' awareness of surrounding traffic, diminish the effectiveness of the safety applications, and may even threaten life if safety-critical functions are dependent on timely and reliable delivery of messages. To address some issues associated with the VPKI-based approach, a number of research works (e.g., [18] [19]) have proposed the use of identity-based cryptography (IBC) schemes to replace the need for public key

certificates. However, IBC-based schemes are one type of VPKI-based approach as they are mostly built on asymmetric crypto primitives like bilinear pairing which are known to be relatively expensive, and thus may not be suitable for time critical safety applications.

Another option is symmetric cryptography, which offers lower message verification delays and small message sizes as it performs lightweight symmetric crypto operations. However, distributing the secret key to legitimate receivers without it becoming known to malicious agents becomes a challenge. One approach involves introducing a form of asymmetry using time to differentiate when a secret key can be used for signing messages and when it can be used to verify the same messages. A protocol called Timed Efficient Stream Loss-tolerant Authentication (TESLA) [20] uses this timing concept to provide lightweight authentication in a broadcast setting. But the conventional TESLA also has some drawbacks when directly applied in V2V environment. For example, the time spent waiting for the secret key to be announced increases the message delay. Governmental agencies such as USDOT [21], consider symmetric cryptography-based schemes like TESLA among the competing security approaches for V2V, whereas some members of the research community see it as a viable option provided its problems can be tackled. For example, research studies in [22] [23] [24] have proposed solutions to some of TESLA's drawbacks e.g. the fixed latency overhead. Not all issues have been fully addressed, and the suggested schemes themselves have vulnerabilities that malicious users could exploit.

1.3 Research Aim

The aim of the research described in this thesis is to investigate and propose solutions that provide authentication and verification of safety messages more efficiently than VPKI-based approaches. The solution needs to demonstrate the following features:

- ***Low Latency**:* Messages must be verified sufficiently quickly under all reasonable traffic conditions, that information can be processed by applications while it is still relevant. More specifically, delays arising from the security mechanisms should not cause the latency constraints of these applications to be violated.
- ***Standard based**:* The C-ITS have developed standards and protocols that work collectively to support the generation, transmission and reception of safety messages.

The security mechanism should conform to this standard framework without requiring major revision.

- ***Minimal Infrastructure****:* The security mechanism should be able to maintain its operation with minimal support regarding a security infrastructure e.g., PKI.
- ***Lightweight:*** The vehicle's on-board units (OBU) have constrained memory and processing capacities. The security mechanism should have minimal computational and storage overheads on the current-technology OBUs.

We focus on approaches using symmetric key cryptography, in particular hash chain-based schemes similar to TESLA.

1.4 Research Questions and Objectives

This research aim is addressed by achieving the following objectives:

1. Develop a threat model describing the range of potential attacks within scope.
2. By means of simulation and theoretical modelling, investigate the performance limitations associated with the VPKI-based security solution and their implications in time-constrained mobile environments. Show that with current technology OBUs, in some traffic conditions, the VPKI solution cannot meet latency requirements. Also, perform similar study with TESLA to show that, despite the latency penalty of delayed authentication, it can meet latency requirements.
3. Investigate and analyse TESLA to identify the drawbacks that deter its usage in the direct V2V/V2I context. The research questions addressed concern overcoming these drawbacks.
4. Propose and evaluate solutions to **Research Question 1**: How can the information required to authenticate disclosed symmetric keys in TESLA be distributed in a timely and efficient fashion?
5. Propose and evaluate solutions to **Research Question 2:** How can the fixed latency overhead inherent in TESLA due to delay in disclosing verification keys be avoided or reduced?

6. Investigate and evaluate solution to **Research Question 3**: how can other sources of position information like the physical features of received signal be use efficiently to confirm the correctness of position information reported in a message?
7. Analyse the proposed solutions in combination (Objectives 4, 5 and 6) to show their effectiveness and practicality in V2V environment.

1.5 Research Contributions

The contributions of this research work are as follows:

- A comparative evaluation and quantitative assessment of VPKI-based and TESLA V2V security schemes in realistic vehicular scenarios through analytical modelling and simulations. The assessment of the VPKI-based scheme uses the Elliptic Curve Digital Signature Algorithm (ECDSA). The performance evaluation reveals that, with traffic-related messages sent every 100ms, ECDSA results in dropped messages and message latency exceeding the 100ms safe limit in traffic situations where a receiver has more than 75 senders within reach. These outcomes are deemed unacceptable as they significantly impact the performance of safety applications. Conversely, the performance of the TESLA approach remains acceptable in such situations.
- Two approaches (VAS-centric and vehicle-centric) to implementing the distribution of commitments to vehicles in a dynamic environment for validation of message verification keys in order to enable the use of TESLA-like scheme in V2V context. The VAS-centric approach, operating at the application-level, involves the selective unicasting of commitments to vehicles by a central server, the V2X Application Server (VAS), and the vehicle-centric approach is a pre-emptive scheme whereby vehicles periodically broadcast commitments themselves. The performance of these approaches is evaluated and compared using theoretical analysis and simulation. Desirable features of the VAS-centric approach include: an average of 94% of commitments delivered before they were needed, high distribution efficiency (with fewer than 2% of commitments sent remaining unused), and a greater resilience to attacks than the vehicle-centric solution and a reactive commitment distribution scheme proposed in [25].

- An analysis of different approaches to reduce or eliminate an inherent limitation (the so-called authentication delay) of TELSA-based security schemes that may satisfy the latency constraints of future safety applications.
- A new protocol based on TESLA called prompt verification (PV) with TESLA as a backup (referred to as PV+TESLA) that optimises the effective authentication delay while keeping the risk due to the vulnerability of pure PV to an acceptable level. PV+TESLA achieves up to 85% reduction in authentication delay while mitigating risk by selectively invoking TESLA for suspicious messages.
- A novel technique based on physical characteristics of radio signals, called RMCCS, for checking the truthfulness of vehicles' position reports. It can be used to enable fast verification of received messages. RMCCS provides an estimate of the actual distance between sending and receiving vehicles that can be compared with the separation implied by the sender position reported in the status message. The main advantage over other approaches to distance estimation based on the LDPLM model of radio signal propagation is that it does not require information about environmental conditions (i.e., built-up areas, traffic, etc.) from external sources in order to determine the path loss exponent. Instead, it extracts this parameter value from the random noise component of the radio signal. The method also generates a measure of uncertainty on the distance estimate. The performance of RMCCS is evaluated using a simulation framework that embodies a realistic representation of signal propagation taking account of factors including the presence of buildings and other vehicles. RMCCS performs well in terms of distance estimation with an accuracy level of about 90% for separation distances less than 100m, making it a promising source of evidence for use as part of the PV+TESLA approach. When integrated in PV+TESLA, RMCCS along with a plausibility check function and a message-clash based detection approach achieves a 97% precision rate.

1.6 Thesis Structure

This thesis is composed of seven chapters, of which this is the first.

Chapter 2 provides an overview of V2X direct broadcast messaging in C-ITS and its security aspects. The first part of the chapter describes the direct V2X communications architecture, the ITS protocol structure designed for the transmission and reception of safety messages among all ITS-enabled entities, and the various access communication technologies proposed by standardisation bodies for providing direct V2V/V2I connectivity. It outlines the safety and non-safety applications envisioned for C-ITS and their corresponding performance requirements. The second part offers an in-depth discussion of security issues in direct V2V/V2I, outlines the requirements for ensuring secure communication and message exchange between vehicles, presents a threat model describing potential attacks in scope. It concludes with a discussion and analysis of different direct V2V security solutions including the standard VPKI-based and TESLA-based solutions.

Chapter 3 presents the comparative evaluation of the performance of standard VPKI-based and conventional TESLA security solutions through simulations and theoretical modelling, demonstrating latency penalties and other performance issues. It also analyses specific problems associated with conventional TESLA when applied in V2V environment.

Chapters 4 presents mechanisms for commitment key distribution in TESLA-like schemes. It discuss implementation details and analyse the performance of the proposed commitment distribution solutions. Also, a formal and informal security analysis of the proposed commitment distribution solution is presented.

Chapter 5 presents and analyses approaches to avoid or effectively reduce authentication delay inherent in TESLA-based approach. It introduces a novel method known as prompt verification (PV) and extensively analyse its susceptibility and risk to impersonation attacks. The chapter then presents a combined approach referred to as PV+TESLA to mitigate the risk associated with the PV's vulnerability. The performance evaluation of PV+TESLA is described in more detail in the following chapter.

Chapter 6 presents a novel method referred called RMCCS that uses received signal strength measurements and their variability to detect when a vehicle is reporting false position

information. The Chapter then provides a performance evaluation of RMCCS using simulation studies. Following that, it illustrates how RMCCS is integrated within PV+TESLA model to provide evidence of suspicious activity resulting from the exploitation of PV's vulnerability and presents the performance assessment of this integrated mechanism.

Chapter 7 summaries the main research contributions, and outlines ideas for future research work and topics for further investigation.

As this thesis is addressing a number of distinct but related research questions, topic -specific literature reviews, problem statement, gap analysis, background discussion, methods used in addressing the problem, and evaluation techniques employed are included in the relevant chapters rather than presented in a single place.

Chapter 2: Direct V2X Communications and its Security Aspects

This chapter provides an overview of direct V2X, specifically focussing on short-range direct broadcast communications and their security in C-ITS. It begins by describing the various direct V2X (V2V/I/P) communications modes, the layered communication stack, and the access technologies adopted by different standards development organisations (SDOs) to enable connectivity between vehicles and other ITS entities. Following this, the chapter offers an overview of the different types of messages shared between ITS-enabled entities, outlining various categories of application envisioned to provide safety and non-safety services.

The narrative then shifts to the security aspects of direct V2X communications addressed in this thesis. The chapter begins by defining a threat model in terms of a range of potential attacks that could affect direct V2V messaging systems. It then outlines the security measures required to ensure the protection of direct V2V messages against these threats. A review and analysis of the asymmetric cryptography-based VPKI security solution recommended by standardisation bodies are presented to demonstrate its capabilities and identify its challenges. This is followed by a description of alternative schemes including identity-based cryptography, and a symmetric cryptography based solution called TESLA. An analysis of TESLA's benefits and drawbacks, when applied in the V2V environment is presented. Finally, a review of existing TESLA-based research works for V2V is also presented.

2.1 Background on Direct V2X Communications

Figure 2.1 shows the three main types of short-range direct broadcast communication namely: V2V, V2I, and V2P. In particular, the V2V communications differ from other kinds of direct V2X, in terms of traffic pattern, high volume of message transmissions, deployment scenarios, and node characteristics. It utilises broadcast communication for delivery of messages to all other vehicles and RSUs within range. The vehicles are equipped with a wireless communication module known as an on-board unit (OBU), which has the capabilities of transmitting, receiving and processing messages. Also, each OBU has a Tamper-Proof Device (TPD) which is used to store private credentials, such as key pairs for security operations (e.g. signing and verifying messages) and vehicle's identity. Furthermore, the OBU contains a communication control unit (CCU) that executes the communication stack and the

application unit (AU) which runs the safety applications and uses the CCU communication capabilities.

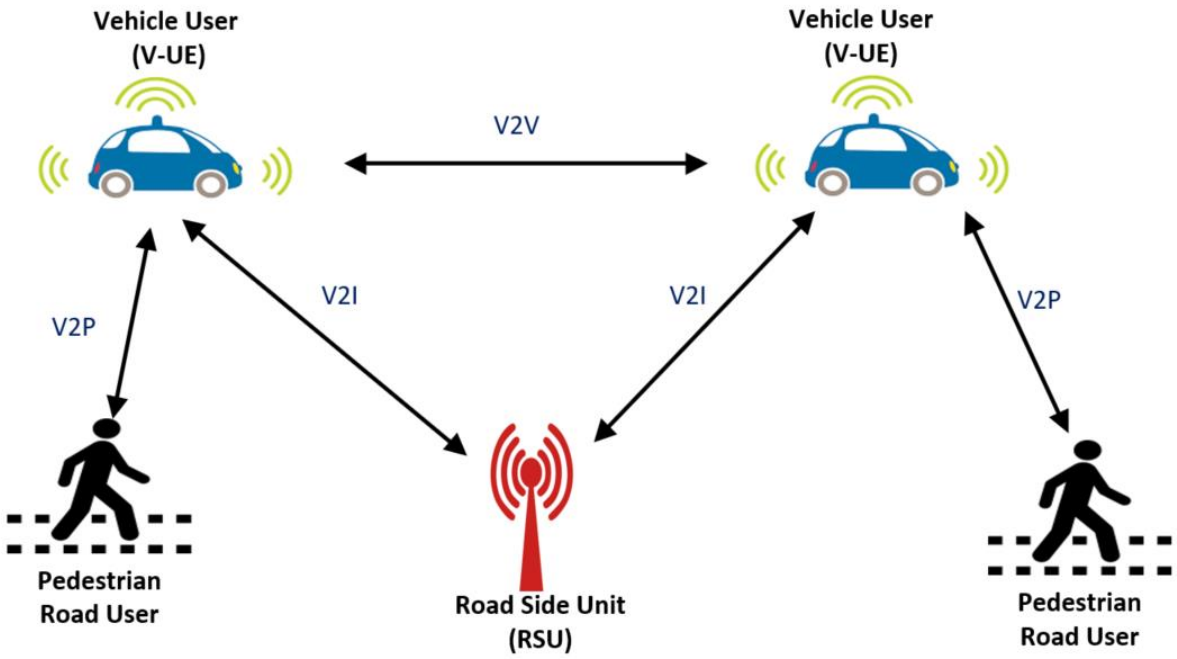


Figure 2.1: Types of Direct V2X Communications

Direct V2X communications have to follow a carefully designed networking model to enable seamless interoperability and exchange of information among devices from different equipment manufacturers. Figure 2.2 shows the block diagram of the layered ITS station reference architecture for V2X communications described in ETSI 302 665 [26].

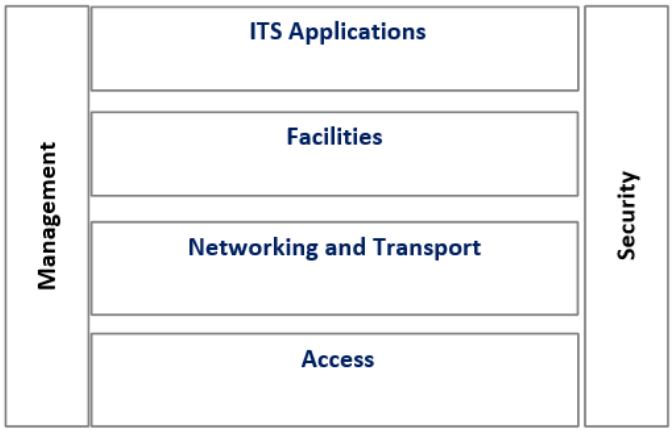


Figure 2.2: Direct V2X layered reference architecture (adapted from ETSI 302 665 [22])

The ITS applications layer hosts a range of direct V2X applications that are expected to contribute to a safer, more efficient, and comfortable transportation system. The ITS facilities layer contains functionality from the OSI application layer, the OSI presentation layer (e.g. ASN.1 encoding and decoding, and encryption) and the OSI session layer (e.g. inter-host communication) with amendments dedicated to ITS. It also provides support to different direct V2X applications by facilitating the access of information on the ITS entity. The ITS Networking and Transport layers are merged together. The Transport sub-layer is responsible for end-to-end message delivery, reliable data transfer, while the Network sub-layer handles message dissemination including point-to-point and point-to-multipoint dissemination for both IP and non-IP based transmissions. The access layer defines the radio technology that will enable efficient and reliable wireless connectivity among vehicles and other ITS devices for the transmission and reception of direct V2X messages. It includes a physical layer (PHY) connecting physically to the wireless medium, and a medium access control (MAC) layer that defines rules for accessing the common wireless medium so that it can be shared efficiently and fairly among a set of vehicles and other ITS entities.

The management component is one of the two vertical components defined with cross-layer functionalities. Its main responsibilities include management and maintenance of the policies for the other layers, congestion control and transmission power allocation, and cross-layer resource optimization. The security component is also defined with cross-layer functionalities that interact with all other layers. It provides security services including ensuring authenticity

and integrity of exchanged messages, authorisation and certificate management. More details about the security component are presented later.

2.1.1 Direct V2X Communications Technologies

This sub-section describes the two main candidate technologies for short-range direct broadcast communication: IEEE 802.11p and cellular V2X (C-V2X) using the so-called direct PC5 interface.

2.1.1.1 Direct Short-Range Communication Technology: IEEE 802.11p

DSRC utilises IEEE 802.11p at the access layer. This is a modified version of the popular IEEE 802.11 WiFi standards specifically designed to support inter-vehicular communications. It employs a carrier sense multiple access with collision avoidance (CSMA/CA) technique to handle channel contention at the MAC layer. This allows for fully distributed and uncoordinated access to the channel, with no need for a resource allocation procedure.

DSRC standard vary significantly from one region to another. Figure 2.3 illustrates the layered protocol stack for DSRC communication in the US and Europe, including names of protocols and standards intended for use at the various layers. In the US, the IEEE 1609.x family of standards is utilised for message transmission and security services. Meanwhile, the SAE J2735 Message Set Dictionary standard, providing a set of message formats, is used to support several safety and traffic efficiency applications. In Europe, ETSI defines its own set of standards and protocols (e.g. GeoNetworking EN 302 636-4-1 [28] and basic transport protocol (BTP) EN 302 636-5-1 [29]) while adopting the IEEE 802.11p standard in the access layer, which they referred to as ITS-G5 [30]. Similar to US DSRC profile, ETSI also uses the IEEE 1609.2 for security services, but refers to it as ETSI 102 940. In contrast to the US DSRC version, ETSI specifies two message formats at the facilities layer to support the direct V2X applications: a periodic message type called CAM and an event-triggered message type called DENM.

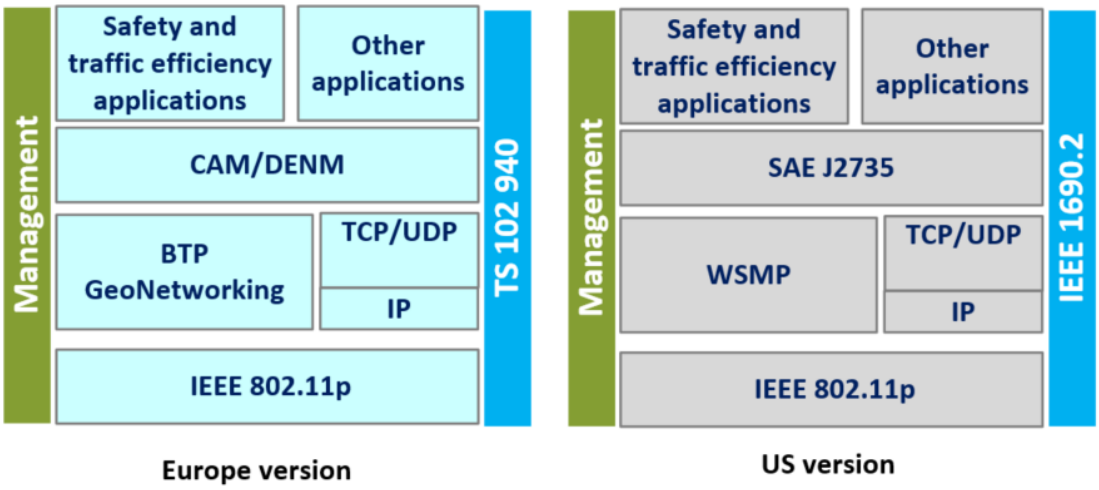


Figure 2.3: Layered Architecture for DSRC Communications in US and Europe

2.1.1.2 Cellular Direct V2X by 3GPP

The direct communications standard known as the PC5 interface or sidelink defined by 3GPP is considered an alternative to the DSRC standard due to its bandwidth capacity and mobility support. After years of activity in various working groups (e.g., Radio Access Network), 3GPP, the standardisation body for cellular communications introduced support for direct V2X communications under LTE Release 14 [34], and subsequently enhanced under 5G, which is referred to as direct C-V2X. The sidelink will be utilised for direct V2V broadcast communications between vehicles to support the low latency and high reliability V2V services and some V2I based safety services.

C-V2X covers only the access and transport/network layers in Fig. 2.3; the same mechanisms as for DSRC can be used for the upper layers. Effectively, the direct C-V2X provides an alternative transport mechanism only i.e., the radio transmission links. Also, the 3GPP introduced some enhancements to its existing network architecture with the addition of two new components to handle V2X operations as described in TSG SA2 [35]. These are the V2X control function (VCF) and V2X application server, as shown in Figure 2.4.

The VCF is responsible for functions such as service authorization and provisioning. It also coordinates resource sharing for communications between vehicles operating under PC5 mode 3 that subscribe to different cellular operators. In contrast, the DSRC is a fully

distributed system that does not require an entity for service provisioning or resource coordination.

The V2X application server provides information related to road safety and traffic efficiency (e.g., notification of traffic congestion) to vehicles. It could be deployed either within the cellular network or outside it, hosted and operated by e.g., a government bodies like the US Department of Transportation, or vehicle manufacturers. The DSRC-based V2V system also requires an application server similar to the one proposed in C-V2X for providing safety and non-safety related information to vehicles and other roadside infrastructures. In fact, both the direct C-V2X and DSRC enabled vehicles could be linked to the same V2X application server.

The direct C-V2X will also reuse the existing uplink/downlink radio interface (known as Uu interface) for communication between vehicle and a base station in order to reach an edge server, back-end server (e.g. to obtain traffic updates from traffic centres). For example, a vehicle can transmit information via an uplink to a relevant V2X application server. The application server can also distribute messages via the downlink to a group of vehicles in a geographical area through the evolved multimedia broadcast or multicast service (MBMS) in the cellular network [36].

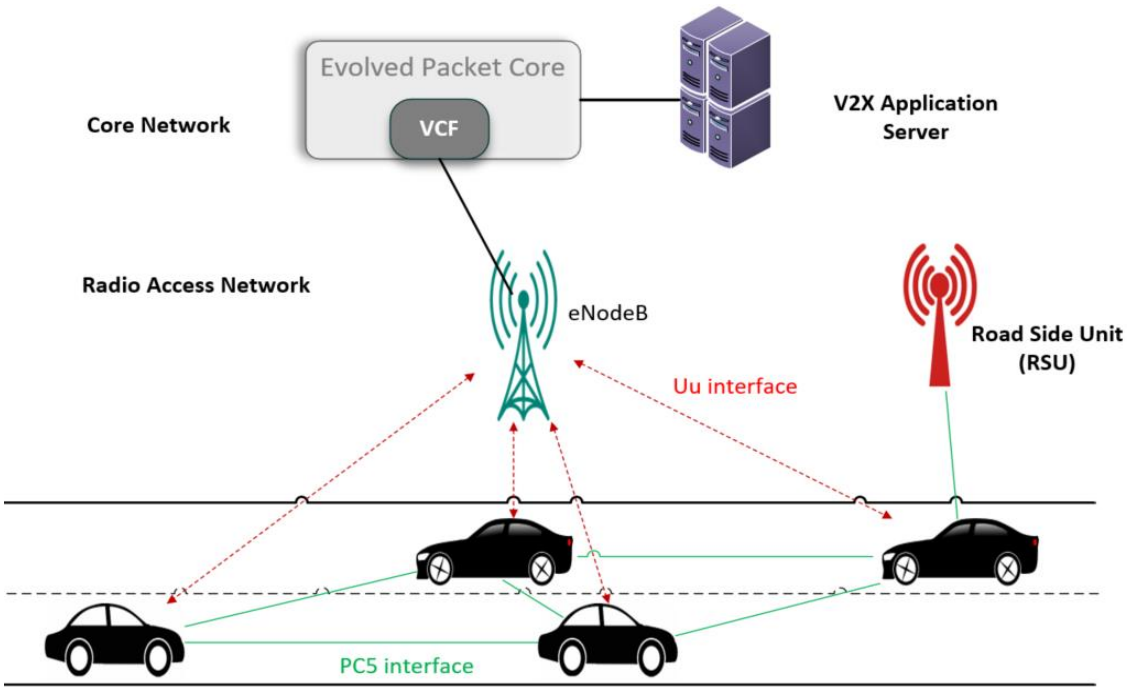


Figure 2.4: 3GPP Architecture Enhancement for V2X Communications

2.1.2 Direct V2X Message Types

While most of the technology involved in V2X communications has been coordinated globally, there are some few regional differences. One is the difference in the set of messages defined for V2X communications. The ETSI standards adopted by the EU have defined two separate classes of message for safety applications: the cooperative awareness message (CAM) [5] and the decentralised environmental notification message (DENM) [7]. The CAM contains vehicle kinematic information e.g., position, speed, acceleration, etc., and it is sent at regular intervals, usually between 1 to 10 messages per second, but they could be sent more frequently depending on the safety applications' needs. Through transmission and reception of CAMs, vehicles are able to track the movement of other vehicles in their vicinity, as well as receive information about possible road hazards. CAMs are specifically intended for use by safety applications, but the information contained in a CAM is also usable by non-safety related applications.

The DENM is an event-driven message that is mainly used to alert vehicles of sudden or catastrophic road events ,e.g., imminent crashes, or detected hazardous conditions. Typical information in a DENM message contains the event attributes such as event type, event location and the area affected by the event. Unlike a CAM, a DENM is not broadcast on a regular basis. The application layer determines when to request the facilities layer to generate a DENM.

The SAE in the US has defined the so-called basic safety message (BSM) [6] in the SAE J2735 standards [43] to convey both a vehicle's kinematic information such as position, speed and heading, and event-triggered information originating from vehicles, such as emergency brake light. The BSM is further subdivided into two types; type 1 contains the kinematics information that is sent at regular intervals (which is equivalent to a CAM), and type 2 which conveys safety events and its related information (which is equivalent to a DENM).

2.1.3 V2X Applications and Requirements

V2X applications are grouped into three main categories according to the purpose they serve: road safety, traffic efficiency, and infotainment. Here, we briefly described the road safety and traffic efficiency applications, while leaving out the infotainment category as it is not relevant to security.

- **Road Safety Applications**: These applications are aimed at reducing the frequency and severity of vehicle crashes, property damage, and human casualties. This includes applications concerned with critical decision making, like coping with abnormal vehicular behaviour, protection of vulnerable users like cyclists and pedestrians, and making way for passage of emergency vehicles. Applications in this category are characterised by their very strict requirements in terms of message latency, communication reliability, and frequency of message broadcast. For example, the basic set of C-ITS use-cases (such as slow vehicle warning, intersection collision warning, slow vehicle warning, etc.) mostly require a message latency of 100ms or less and message frequency of about 10Hz [44], whereas the advanced road safety use-cases (e.g. vehicle platooning and extended sensor driving) require a message latency as low as 3ms and a message frequency as high as 50Hz [45]. It is because of these strict requirements that safety messages are sent in direct broadcast mode as opposed to V2N, which uses unicast and/or multicast mode.
- **Traffic Efficiency Applications:** This category covers a broad range of applications and use-cases intended to optimise the flow of vehicular traffic on roads. This includes dissemination of information about road condition and congestion. Applications in this group mostly use the V2N/V2I communications modes with few of them using V2V. Also, they have less challenging delay and reliability requirements than the road safety use-cases since there is no need for prompt (re)action on the vehicle side. For instance, the low speed following use-case can tolerate up to 500ms message delay and requires a message frequency of just 2Hz.

This research work focuses on ensuring that the strict requirements in terms of low message latency, reliability and high message frequency of road safety applications are met.

2.2 Security Aspects of V2V Communications

In this subsection, we describe the security issues and threats facing V2V messaging systems, V2V security requirements, and different V2V security solutions including the VPKI-based approach defined by standard organisations and alternatives proposed by the research community. We begin by describing an attacker model and the attacks that are within scope.

2.2.1 Attacker Model

Here, we develop an attacker model for V2V messaging systems following the widely known Dolev-Yao threat model [46]. In this model, all communicating nodes are not trusted, and an adversary has the following features: it can (a) obtain any message passing through the network or system from any party (b) act as a legitimate user (i.e., it can initiate a conversation with any other user) (c) become the receiver to any sender, and (d) send messages to any entity by impersonating any other entity. The V2V broadcast communication scenario fits the Dolev-Yao model, because any node with V2V-enabled interface can transmit, receive, and read broadcast messages. Moreover, any V2V-enabled node could be an adversary (i.e., broadcast message containing misleading information, replay legitimate messages, etc.), and adversaries could exist anywhere in the system. However, there are some unique characteristics of V2V messaging system that constrain the adversary's capabilities. First, an attacker cannot prevent a message being received due to the V2V broadcast nature. Secondly, message delivery is unreliable (i.e., broadcast messages could be lost due to transmission errors), and the transmission range is finite i.e., only vehicles within the attacker's range are likely to receive the message. Thirdly, the message delivery channel is a finite resource. Lastly, vehicles have finite message processing and caching capacity. These factors limit the attacker's ability to achieve its goals and targets. Below we give some dimensions by which an attacker can be classified as adopted from [47]:

Insider vs Outsider: The insider is an authenticated member of the V2V messaging system that possesses valid cryptographic credentials and can communicate with other vehicles. They can use their own identity to carry out their malicious actions, for example, broadcasting messages containing misleading information, etc. Detecting this type of adversary is a challenge due to the possession of valid credentials. The outsider is considered by other vehicles as an intruder and thus is limited in the diversity of attacks he can perpetrate. Both Insider and outsider are able to generate and broadcast safety messages. Also, they can extract the identity of a legitimate vehicle from its received message, and then use it to broadcast their own messages. This helps conceal the attackers' identity as the messages will appear to come from the legitimate vehicle and prevent the attacker from being detected.

Malicious vs. Rational: This dimension is about the attacker's motivation i.e., whether the attacker is motivated by causing harm to others or by benefitting itself. A malicious attacker

aims to intentionally disrupt or harm the functionality of the C-ITS or its member (vehicles, RSUs). For example, causing accidents, physical damage, blocking road, etc. On the contrary, a rational attacker seeks personal benefit from the attack and hence is more predictable in terms of the attack means and target. For example, clearing the road traffic on their own route by convincing other vehicles to take alternative route, target authority vehicles (e.g., police, ambulances) in order to destroy their reputation, etc. Note that there is a possibility for an attacker to be malicious and rational at the same time, for example, when the disruption is beneficial to itself.

Active vs. Passive: An active attacker can generate and broadcast different types of message related attacks. These are messages containing misleading information, replaying a genuine message at a later time, generating high volume of messages more than the required message frequency in order to overwhelms receiver's processing and caching capacity, etc. Most of the V2V message attacks are carried out by active attackers. A passive attacker listens to a single or sequence of messages being transmitted in order to analyse or obtain any useful information from it. But since safety messages will not contain any sensitive information, a passive attacker has little or no effect by listening to the messages.

2.2.2 V2V message-based attacks

In this sub-section, we describe different types of message-based attacks that can be perpetrated against the V2V system, as indicated in some V2V security studies like and ETSI TR 102 893 [49]:

Bogus Messages: Here a vehicle broadcasts a message conveying misleading information in order to affect the behaviour of other vehicles within the vicinity e.g., to divert traffic from a given road, to report a false position information, etc. Such type of messages can be generated by an active insider or outsider. But to make the bogus message more convincing and increase the chances of being accepted and verified by the receivers, it should be sourced by an active insider. The attacker's motivation could be both malicious and/or rational depending on the circumstance. For example, the adversary could report false information about bad traffic conditions to its neighbours, forcing them to take alternate path, while freeing the path for itself.

Identity Impersonation: Typically, in V2V scenarios, the receiver does not care about the actual identity of sender, but to filter-out bogus messages, will want to know that the sender is trustworthy/authorised to send such messages i.e., an insider. To counter this, an adversary (insider or outsider) may assume the identity of one or more legitimate vehicles and use it broadcast a message of its choice (bogus). Similar messages received apparently from multiple senders could make the information contained in the messages more credible. Some of the reasons for impersonating insider's identity in V2V context are: to make it more likely that the messages are accepted, conceal the adversary's real identity, or implicate a given target vehicle. Identity impersonation can be motivated by malicious and/or rational objectives.

Denial of Service (DoS) attack: The main objective of this attack is to prevent legitimate users from using the network services by tying up finite resources. An example of a DoS attack in the context of V2V, is the situation whereby a vehicle receives a large number of fake or genuine signed messages and is unable to verify them all in time to react. Similarly, the receiver's input message buffer could overflow causing messages to be dropped. Such so-called computation-based DoS can easily occur among vehicles in high traffic density regions even without any malicious intention.

Replay Messages: Here an adversary (either insider or outsider) records a message that was transmitted by a legitimate vehicle and later re-broadcasts it one or more times for malicious and/or rational objectives. Although the message content was genuine at the original time of transmission, it could be misleading if received at a later time or multiple times.

From the attacker classification and types of message attacks, together with the Dolev-Yao threat model features and the characteristics of V2V, we now define the capabilities of the adversary considered in this work as follows: The adversary can either be an insider or outsider exploiting V2V broadcast messaging as the main attack surface. It can be a rogue individual (stationary or mobile) with V2V-enabled device capable of transmitting and receiving safety messages, or a set of mobile or stationary nodes colluding together. We can consider that the adversary is active with the goal of influencing vehicle behaviour by generating bogus messages. The type and content of the bogus messages constructed by the attacker depend on the attacker's aim, which could be malicious, rational or combination of both. In addition, the adversary can be replaying legitimately generated messages at a later

time. The adversary can impersonate any legitimate vehicle (or other ITS-enabled entities e.g., RSU) by extracting the vehicle's identity from its broadcast message. It is also assumed that the adversary can circumvent the security measures that are put in place in order to avoid being detected and sanctioned or excluded from the V2V system. For example, an adversary may deny responsibility of sending a given bogus message, or may falsely claim that another vehicle sent (or did not send) a given message in order to mislead an investigation. The adversary can use transmission power different from that defined by the C-ITS standard in order to extend his message dissemination range. However, we assumed that all computations including storage of secret values are secure.

To provide protection for V2V broadcast messaging system, the global ITS standardisation effort in the V2X domain including the 3GPP working group on security [50] and ETSI ITS WG5 specify the following security requirements:

Source authentication: receiving vehicles shall be able to authenticate and verify that the sender of the received broadcast messages has a valid identity to participate in V2V broadcast communication. Hence, ensuring source authenticity prevents outsider attackers.

Message Integrity: In the context of V2V scenario, the integrity of the received broadcast messages shall be checked and verified to ensure that the message content was really generated by the claimed sender. This is important because the safety applications rely on the information reported in messages to make decisions and subsequently take actions. Hence a receiver should be able to verify that a message is indeed from the source named.

Replay Protection: Freshness of V2V broadcast messages shall be ensured so that receiving vehicles accepts only freshly generated messages, thus preventing replay attacks.

Non-repudiation: Ensures that once a vehicle broadcasts a message, it cannot deny that action later, in the event that some incorrect behaviour is detected. This property will allow a receiving vehicle to prove to a third party, who is accountable for generating the broadcast message.

2.2.3 V2V Security Solutions

In this sub-section, we describe and analyse different techniques proposed to address the security challenges of V2V broadcast communications. We start by an in-depth review of the

security framework suggested by the ETSI TC ITS WG5 working group and the IEEE 1609 WG, which rely on asymmetric cryptography and a vehicular public key infrastructure (VPKI). After that, we discuss other alternative techniques explored by the research community and other stakeholders including the USDOT.

2.2.3.1 VPKI-based Security Solution

The IEEE 1609.2 [51] and ETSI TS 102-940 [52] are both based on the concept of asymmetric cryptography and the use of VPKI to provision and manage the distribution of certificates to vehicles. In these VPKI-based solutions, vehicles are either issued a set of short-lived public/private key pairs, or a long-term public/private key pair but with a certificate revocation procedure to sanction vehicles identified to be misbehaving. The private keys are used to digitally sign each outgoing safety message using the elliptic curve digital signature algorithm (ECDSA). Vehicles receiving a signed message are able to authenticate the message provided that they have the corresponding public key certificate of the sender issued by a trusted certificate authority (CA). The scale and dynamic nature of V2V systems preclude prior distribution of certificates. Consequently, a broadcast packet from a sender V_s typically contains the message *M*, a timestamp *T*, the sender's signature *σ* using its private key, and the public key certificate of the sender *Cert*, as shown below:

$$V_s = \{[M|T]_\sigma, Cert_s\}$$

The signature, *σ,* ensures that the sender is accountable for this message, and the message is not altered during transmission. The timestamp T is for freshness checking to prevent replay attacks. $Cert_s$ is used to announce the sender's identifier and public key and assert the sender's legitimacy via signature of the certificate by a trusted CA. In this way, any vehicle receiving a packet can confirm its authenticity by checking that a trusted authority has signed the sender's certificate, and then verify the sender's digital signature of the received broadcast message was generated using the public key specified in the signature. This provides evidence of the trustworthiness of the sender and the integrity of the received message without a prior security relationship with the sending vehicle.

The VKPI based security approach satisfies all the V2V security requirements outlined above, and can thus provide the needed protection to V2V messages. However, its advantages are accompanied by some challenging problems and practical concerns as follows:

- **High Signature and Certificate Verification Delay:** Asymmetric cryptography operations introduce relatively high computational demands, which might affect the performance of time-critical safety applications. The main overhead is the time spent verifying certificates and signatures attached to received messages. In high traffic density conditions, where hundreds of vehicles may be within communication range, broadcasting safety message every 100-300ms, each receiver needs to verify a large number of messages. If their processing capacity is not sufficient, time-sensitive messages may be queued and consequently delayed, or even get dropped, which can affect the performance of some safety use-cases, particularly those with stringent latency requirements. Previous studies have investigated the effects of signature verification processing overhead under different road traffic conditions using simulations works (e.g. [16] [17]). They find that the VPKI-based security schemes suffer from high signature verification delays on received messages. This conclusion is confirmed by the original work reported in Chapter 3.
- **High Communication Cost:** Public key signature operations require a significant amount of additional information to be sent with the safety message, including the signature itself and the certificate containing the verification key, CA's identifier, etc. This cryptographic content requires at least 240 bytes of space [53], and needs to be included in every safety message. The increased message length adds to memory requirements, transmission and reception overheads, and the radio channel occupancy.
- **Vulnerable to Denial of Service Attack:** In high traffic density conditions, the above-mentioned verification delay leads to what is termed as computation-based DoS attacks, even without any malicious intention.
- **Requirement for an Infrastructure:** Deploying a VPKI on a national or regional scale is a major operation requiring substantial time and investment due to administrative policies, regulatory limitations and installation costs. It is still not yet clear whether governmental transportation authorities or vehicle manufacturers should handle the VPKI operation. Furthermore, the security infrastructure should be capable of dealing with the scale of entities (vehicles, RSU) in the ITS system. For instance, it is not yet clear how VPKI design would support vehicles crossing regulatory boundaries.

- **VPKI Operation and Management Issues:** In the IEEE 1609.2 version, a certificate revocation list (CRL) is expected to be distributed on a regular basis to all vehicles in order to inform them of certificates that should no longer be trusted. Vehicles that are out of network coverage will not receive the CRL and thus will be unaware of newly-discovered malicious vehicles. Dissemination of the list also generates extra communication overhead. Moreover, the CRL checking process itself increases the verification time of received broadcast messages. The ETSI TS version does not use a CRL, but instead issues short-lived certificates that are not renewed if a vehicle is judged to be untrustworthy. However, renewal of such certificates on-demand also requires always-on and reliable connection. In addition, the certificate reloading process incurs a delay of up to 500ms as mentioned in [54]. Such a delay could affect the performance of the underlying V2V safety application.

The above issues suggests the need to evaluate other security solutions that incur low computational overhead in congested environments, have less requirement for an infrastructure to function and low implementation complexity and cost compared to the VPKI-based approach.

2.2.3.2 Identity-based Cryptography Solutions

Identity based schemes are proposed to be applied in V2V in order to simplify the complexity of VPKI-based schemes. The concept of identity-based schemes is that instead of using public certificates to announce message sender's public key, a vehicle uses its identity (e.g. vehicle registration number) to uniquely determine its public key. The advantage of this approach is that it reduces the communication overhead since there is no need to attach certificates to every broadcast message and also the time required to process and verify the public certificates is saved. In addition, it eliminates the need for a large-scale infrastructure to generate and manage the distribution of public key certificates to the vehicles. Several existing research works have proposed using an identity-based method to authenticate vehicles and protect the safety messages.

The work in [55] proposes a V2V authentication scheme using an identity-based approach. It uses a key management centre (KMC) that generates public keys to registered vehicles using their identities, and a private key generator (PKG) that generates and securely communicates

corresponding private keys to the vehicles. Vehicles authenticate each other's public and private key pairs through a challenge-response, a random number generator and an asymmetric encryption algorithm that encrypts and decrypts the messages exchanged between them. In [56] , the authors propose an efficient identity-based signature scheme that uses Elliptic Curve Cryptography (ECC) and general one-way hash functions to authenticate safety messages. Similarly, [57] [58] [59] propose an ID-based signature scheme based on bilinear pairings for V2V message authentication. The sender uses Map-To-Point hash functions to generate anonymous-identities, and receiving vehicles uses both bilinear pairing and Map-To-Point hash function operations to verify signed messages.

Similar to VPKI-based schemes, the identity-based approaches can satisfy the V2V security requirements, but they still suffer from signature verification delays because they also utilise asymmetric cryptographic operations. As reported in the works of [56], the time taken to verify a message signed using bilinear pairing operations ranges between 2ms to 18ms. Compared this to VPKI's message processing time of around 0.2ms to 8ms for different ECDSA algorithms and key sizes as shown in [60], the identity-based approaches have higher computational cost.

2.2.3.3 TESLA based Security Solution

The TESLA protocol was first introduced in [61] as a lightweight protocol that uses symmetric cryptographic primitives and time synchronization to provide source authentication and message integrity in a broadcast setting. It makes use of one-way hash functions (e.g. SHA 256) to generate key chains, which are subsequently employed for authenticating broadcast messages. TESLA uses a symmetric Message Authentication Code (MAC) algorithm to protect the integrity of messages but introduces an element of asymmetry by delaying the disclosure of the secret key used. A given key may only be used by a sender to generate MACs within a well-defined time window referred to as a TESLA interval, after which it is made public and may be used by receivers to verify the integrity of messages sent within that window. A new key is then used for the next window. A sequence of keys used by a given sender is generated such that the *nth* key used is the result of applying a hash function to the *n+1th* key. Thus, the hash function can be used to verify that a key is part of the same chain as one used previously, but cannot be used to predict the next key to be used. To be sure of the identity of the sender

of a sequence of messages, the first key (called the commitment key K_0) in the sequence must be known to the receiver and reliably attributed to that sender.

Fig. 2.5 describes the operation of the TESLA protocol at both the sender and the receiver. At the sender, a chain of secret keys is pre-computed from a seed value using the hash function. The commitment key (K_0) of the chain currently in use by the sender is securely sent to all receivers likely to receive the sender's message. The sender broadcasts signed messages, disclosing the key and changing to the next key in the chain after each TESLA time interval. The keys are then used in the reverse order to which they were generated. At the receiver side, the K_0 and the TESLA time interval are stored. Each message received from the sender is temporarily buffered, and later verified after the corresponding disclosed key is received and is proven to be part of the sender's chain. Once a key is disclosed, it is no longer valid to sign another message. The main advantage of TESLA is rapid processing of packets at both sender and receiver, because symmetric key operations are much faster to perform than asymmetric key operations. In addition, messages are smaller than in digital signature schemes as symmetric MACs are smaller than asymmetric signatures and there is no need to send a certificate with the message., This results in lower communication overheads.

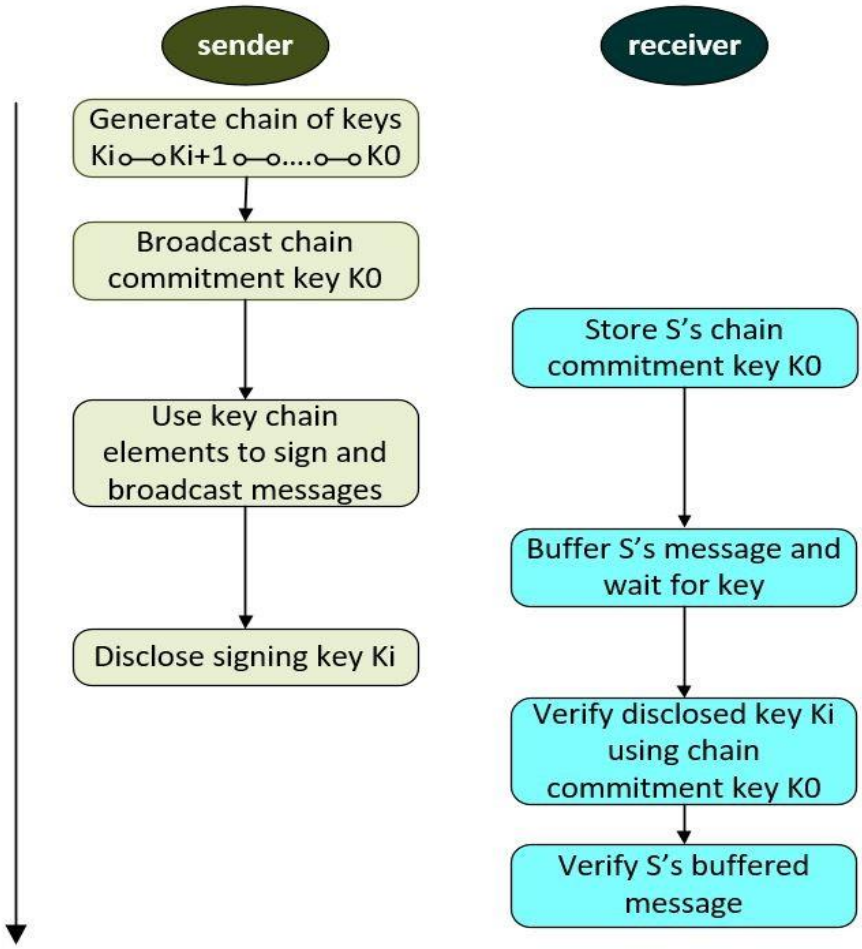


Figure 2.5: The TESLA Protocol

TESLA can be used to protect V2V broadcast communication, but it has some characteristics that are undesirable in such an environment, as outlined below:

- **Delayed Authentication**: In TESLA, on receiving a message, a node must wait at least until the next time interval, when the key used to sign it is disclosed, before it can authenticate it. This waiting time is referred to as the authentication delay. However, V2V broadcast messages, particularly those used by critical road safety applications, are time-sensitive and need to be verified and acted upon while they are still relevant.
- **Distribution of the chain commitment key**: TESLA requires nodes to distribute a commitment key, K_0, for the node's current key chain to all nodes likely to receive a message broadcast by it. The commitment key must arrive before the receiver needs to verify the key used to sign a message from that sender. In a large-scale distributed system with a large population of independently-moving nodes such as is the case in

V2V, the neighbourhood of a node is changing continuously. Deciding which nodes need to know a given commitment at a given time and ensuring they receive it in time is non-trivial.

- **Lack of a non-repudiation Mechanism**: TESLA alone does not provide non-repudiation. Once a key has been disclosed, it can be used by a malicious agent to generate a MAC for a fabricated message purporting to have been sent at an earlier time. Thus, without additional measures such as some form of trusted time stamp or logging mechanism, it would not be possible to distinguish it from a genuine message sent at the time claimed. A malicious agent could thus plausibly deny responsibility for a message it had sent, dispute the content of a message presented as evidence of dangerous driving, and so on.

2.2.3.4 Previous research addressing TESLA's limitations for V2X

Several research works in the academic and industry communities have adapted the original version of TESLA to address some of the problems described above.

For the authentication delay problem, the authors of [62] [63] [64] and [65] use a prediction-based approach to achieved timely verification of messages. They exploit the ability of a sending vehicle to predict its own future position. The sender constructs what is called a Merkle Hash Tree (MHT) to generate a prediction outcome, which is sent to the receiver in advance to enable instant message verification. Each message is signed with a TESLA key and sent together with a prediction that is used to estimate the sender's future position in advance. With the prediction outcome known in advance, a receiver uses this information to reconstruct the sender's position and then compare it with the position information received in the following message. If the predicted position matches the reported position, then the message is verified, otherwise, it is verified later with the TESLA key. Their simulation results show that the overall message verification delay is around 10% of the pure TESLA value. However, the effectiveness of their approach depends on the accuracy and computational cost of the chosen prediction algorithms. Also, it is possible for an attacker to construct a valid prediction outcome based the on vehicle's past and current position information, road layout, etc. In addition, when the message conveying prediction outcome is lost in transmission, the corresponding message is verified using standard TESLA approach. Thus, the more message loss particularly in high traffic density scenarios, the less effective approach becomes.

Few research works address distribution of commitment keys. In Bao et al [25], the authors propose a reactive approach. When vehicle V_A receives a safety message from V_B whose commitment key it does not have, it sends V_B a commitment key request. V_A also sends a request to a nearby roadside unit (RSU) to obtain a special token that allows it to verify the commitment key received from V_B. This approach is evaluated in Chapter 4, as it is a reactive version of our proposed vehicle-centric commitment distribution scheme. The TESLA-based cooperative message verification scheme described in [69] involves periodic broadcast of commitment keys at a fixed time interval. However, the authors did not evaluate the performance of this approach in terms of distribution efficiency, timeliness of commitment and impact of commitment messages on the delivery of safety messages.

2.2.3.5 Adaptive Solutions

To overcome the problems of the VPKI-based approach, particularly the high message verification delay, some research works have proposed different adaptive techniques. The idea is to prioritise verification of messages from nearby vehicles that are a source of safety concern to the receiver. In priority-based verification schemes, high-priority messages are verified first, then low-priority messages are verified later or even get discarded depending on the sufficiency of resources. For example, the studies in [70] [71] [72] prioritise received messages according to the physical attributes of the senders: position, acceleration, and speed. Then, verification is done based on priority class. For the same priority class, a round robin manner is employed. Also, the authors of [73] proposed the classification and verification of received messages based on the sender-receiver distance, which is estimated using the received signal strength. Based on the estimated distance, vehicles classify the geographical region around them into several safety areas (SA, e.g. 50m, 100m, etc.), and then the received messages are mapped to the corresponding SAs. The messages from senders located at low SAs are verified first followed by others.

Another adaptive solution is the random-based message verification scheme. In this, messages are selected at random and then verified particularly when too many messages are received e.g. in high traffic density conditions. The studies in [74] used this method to increase the scalability of signature verification. Similarly, several V2V message authentication schemes e.g., [75] [76] have applied this method because of its scalability and simplicity.

However, the main drawback of the random-based verification method is that it may cause some relevant and important messages not be verified or verified too late.

2.3 Summary

This chapter has provided an overview of direct V2X communications activities undertaken by various standards organisations and stakeholders. The main candidate technologies proposed to support direct V2X broadcast communications have been described, each with its unique characteristics. The two categories (i.e., road safety and traffic efficiency) of direct V2X applications have been presented. This thesis focuses on safety applications and their use-cases because safety is of paramount importance. These set of use-cases are designed to process information contained in messages exchanged between vehicles to detect dangerous situations on road and subsequently prompt the driver to take necessary action or takeover control of the vehicle. Thus, acting on incorrect information may lead to wrong decision making by the safety applications, putting human safety at risk. Therefore, it becomes necessary to ensure the authenticity and integrity of the shared messages in order not to jeopardise the benefits of the safety application and the C-ITS at large. However, providing such security features for safety messages is a challenge due to the strict performance requirements of the safety applications (i.e., low message latency, reliable message delivery), the dynamic topology and large scale of the V2V network, and the high volume of messages that vehicles are expected to process.

The chapter then describes the security concerns in the context of V2V and analyses different ways to address them. It provides an overview of various threats and attacks that present eminent risk to the V2V messaging system, highlights the required security features that need to be put in place, and then presents an analysis of the strengths and weaknesses of different security solutions envisioned for use in V2V. While the mainstream VPKI-based security solutions satisfy V2V security requirements, high message latency and potential dropping of messages, particularly in high traffic density conditions could limit their effectiveness in practice. TESLA, which is seen as an alternative to VPKI-based approach because it is based on lightweight symmetric cryptography operations has some challenges in V2V context. TESLA's fixed authentication delay which increases the overall message latency need to be

reduced or eliminated, efficient scheme for distributing commitments in the dynamic V2V environment is critical, and non-repudiation property need to be supported. Although some studies have proposed solutions to some of these challenges, more work needs to be done. Tackling these issues will make a TESLA-based solution practical and more attractive in the V2V context.

This chapter lays the foundation for this thesis. The next chapter will compare the performance of VPKI and TESLA through simulation and theoretical modelling, confirming the issues with VPKI and demonstrating the desirability of a TESLA solution in the V2V environment. Identity-based and other alternative schemes described above that also relies on asymmetric cryptographic primitives to protect safety messages, have not been included in the comparative evaluation. The reason for this exclusion lies in their similar processing times to the VPKI approach, making them less relevant to the specific focus of this comparison, which centres on symmetric and asymmetric cryptography solutions.

Chapter 3: Comparative Evaluation of VPKI-based and TESLA Security Solutions

The previous Chapter presented the standard VPKI-based security solution and other alternative schemes, and outlined the challenges associated with each. In particular, the VPKI-based solution has a number of potential issues including high signature and certificate verification delays due to the computationally expensive nature of ECDSA operations and the need to include a public key certificate in every message broadcast. These issues make the use of VPKI-based schemes for verification of time-critical V2V safety messages problematic, especially in high traffic scenarios where hundreds of messages are expected to be received and get processed by a given receiving vehicle. In addition, the high computational cost of a VPKI-based scheme makes it vulnerable to denial-of-service attacks even without malicious intent when messages are received at a rate higher than they could be processed by a receiving vehicle.

This chapter is going to explore and quantify the consequences of these properties to confirm whether a problem does exist with the standard VPKI-based security scheme. The chapter will also examine whether the fixed waiting time of the symmetric cryptography-based TESLA may be preferable to the long message latency introduced by the VPKI-based scheme in high traffic scenarios. In order to gain this knowledge, theoretical modelling based on queueing theory and simulations of realistic vehicular traffic scenarios are conducted. In this, the authentication of broadcast messages using VPKI-based, TESLA-based approaches are analytically modelled in a V2V setting, and their performances are analysed and compared using a range of performance evaluation indicators. The results obtained from the analytical models are compared and validated with those obtained from the simulation works. The comparison and analysis conducted will give a better understanding of the problems with VPKI-based security solution, and help to confirm whether the security scheme will be able or unable to function under high traffic conditions.

3.1 A Queuing Theory Model for V2V Communication

The delivery of a broadcast message has three main stages. First, the message is generated and formatted ready for transmission at the sender side, then it is broadcast on the wireless

channel, and finally, it is received and decoded/processed by all the receivers in range. All three steps involve shared use of a finite resource and take time. In the first and third steps, the resource is a processor assumed to be able to process one message at a time, and in the second step, it is the wireless medium. There are numerous strategies for sharing the available channel bandwidth among transmitters (e.g. carrier sense multiple access with collision avoidance in case of DSRC, Mode 3 or Mode 4 in case of C-V2V), but ultimately its message-carrying capacity is finite.

If a message arrives at a resource and finds it is busy, then it must either be added to a queue to wait its turn, or else be dropped. Conversely, if a resource finishes processing one message and finds its queue empty, then it will be idle until the next message arrives. Increasing the average message arrival rate makes it more likely that a given message will find the resource busy, and so the average queue length and time spent in the queue will grow. However, the greater the average queue length, the less likely that the resource will be idle, so that more messages will be processed per unit time. Provided that the average message arrival rate is less than the capacity of the resource, an equilibrium will be reached such that on average, input and output rates are equal. The higher the throughput, the greater will be the queue length and the longer the time taken.

The simplest model of a queuing system is a memoryless continuous time Markov chain denoted in the so-called Kendall notation as an M/M/1 model. In an M/M/1 queuing system, the is no limit to the length of the queue, the queuing discipline is 'first come first served', the distributions of message arrival intervals and of time take to process a message are both exponential, and there is a single processing resource. In such a case, the average time a message spends in the system (including time spent being processed) and the average queue length are respectively:

$$T = \frac{1}{(\mu - \lambda)} \; and \; Q = \mu(\lambda/\mu)^2 T = \frac{\mu(\lambda/\mu)^2}{(\mu - \lambda)}$$

where λ is the average message arrival frequency and μ is the average rate at which messages can be processed by the resource. Notice that there is a singularity when λ = μ indicating that the steady-state equilibrium model breaks down and the values of T for λ ≥ μ have no physical meaning. Performance targets will not typically be expressed in terms of averages, but rather

as expectations regarding exceptions to the norm. It is also useful, therefore, to consider the time within which a fraction x of messages is likely to be processed:

$$T_x = \frac{\ln(1-x)}{(\mu-\lambda)} \qquad (3.1)$$

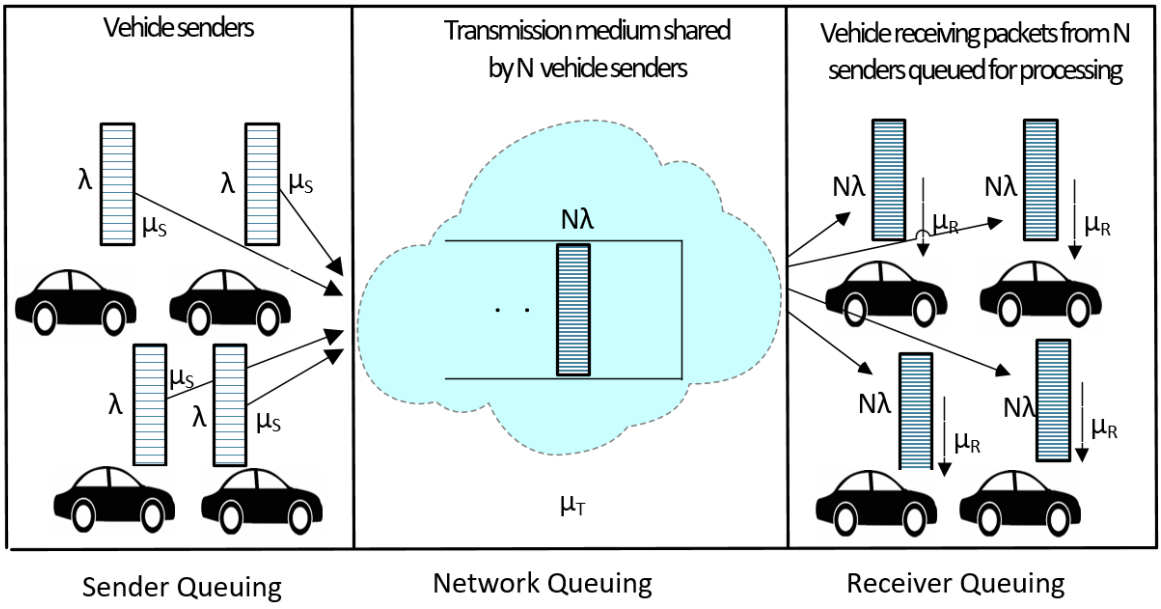


Figure 0.1: End-to-End Queuing Model

In this work, each of the three stages involved during the delivery of a broadcast message is modelled using an M/M/1 queuing system as illustrated in Fig 3.1. The following points are offered as justification of this modelling choice:

- The entity responsible for message generation at the facilities layer of each sender generates a message with a constant probability per small time interval, such that the average frequency of message generation is λ. The message is then forwarded to the lower layer to be sent. This is an approximation to the real situation in which messages would be generated at constant intervals of 1/λ. It is argued that if the number of vehicles is large and their sending behaviours are not synchronised, then the overall effect is similar. This means that the transmission medium (i.e., network queue) is receiving requests to send messages with exponentially distributed interarrival times.
- The time spent by a message waiting for transmission varies due to channel uncertainty. This is because in V2V scenarios, the channel condition changes fast due to vehicles mobility. The network may find that the quality of the available channel

that is to be allocated to a given vehicle is below a predefined channel quality indicator, or that transmission on that channel may lead to interference with other concurrent transmissions. Thus, the collision avoidance and resource allocation events at the network stage introduces some randomness, making the duration that a message sojourns in the transmission queue to be exponentially distributed. This assumption was also considered in the works of [77] when M/M/1 queueing system is used to model and analyse a resource allocation strategy for V2V.

- The non-synchronisation of vehicles combined with collision avoidance and resource allocation events encountered at the network queue results in the messages arriving at the receiver's queue to occur according to Poisson process with exponentially distributed interarrival times.
- The individual vehicles onboard unit operate in half-duplex mode, i.e., at any given time, the onboard unit is only able to carry out a single task for example, generating vehicle's own message, processing a received message, updating the vehicle's local dynamic map, processing information from the vehicles internal sensor system. So there is a probability that the onboard unit might be busy carrying out other tasks at the time the facilities layer needs to generate the periodic safety message. This introduces some randomness in the time at which the safety messages are generated at the sender queue.

Such an arrangement in which a message passes through multiple queue systems, one after the other is usually referred to as a tandem queuing network. Consequently, the average end-to-end delay is given by:

$$T_{total} = T_S + T_T + T_R + A; \tag{3.2}$$

$$T_S = \frac{1}{(\mu_S - \lambda)};\ T_T = \frac{1}{(\mu_T - N\lambda)};\ T_R = \frac{1}{(\mu_R - N\lambda)}$$

and the total average number of messages in the queues is expressed as:

$$Q_{Total} = Q_S + Q_T + Q_R; \tag{3.3}$$

$$Q_S = \frac{\mu_S(\lambda/\mu_S)^2}{\mu_S - \lambda};\ Q_T = \frac{\mu_T(N\lambda/\mu_T)^2}{\mu_T - N\lambda};\ Q_R = \frac{\mu_R(N\lambda/\mu_R)^2}{\mu_R - \lambda}$$

where the subscripts S, T and R stand for Sender, Transmission and Receiver respectively, N is the number of vehicles within the reception range and A is a constant term added for generality. λ is the rate at which messages are generated by the safety applications in each of the N vehicles. The multiplicative factor N is applied to the message traffic flowing through the wireless transmission medium as it is shared by all vehicles, and to that for reception because each message is received by all vehicles within the sender's broadcast range provided the communication is reliable with no message lost. Furthermore, it is assumed that the message processing times have a component proportional to message length and a fixed component independent of length, so that:

$$\mu_i = \frac{1}{(lr_i + c_i)}; i = S, T, R \qquad (3.4)$$

where l is the message length in bytes, r_i is the time to process one byte and c_i is the additional per-message processing time. Message authenticity and integrity measures affect both message length and the per-message processing time. The amount of security credentials appended to every message as well as the time required to process the secured message is different for each security mechanism. Therefore, the μ_S incurred during signing a message at the sender and μ_R incurred during verification of Nλ received messages at the receiver, is different for ECDSA and TESLA security approaches.

The model considers periodic safety messages i.e. CAMs, which are exchanged regularly among vehicles to share mobility information. These messages are the main concern because they have a higher generation rate, λ, (around 10 messages per second) than other types of V2V messages, and they have stringent latency requirement of 100ms end-to-end or less depending on the use-cases [78]. In addition, the model considers C-V2V PC5 resource allocation for the shared transmission medium, and use the direct C-V2X requirements and specifications to obtain some of the model parameters. Nevertheless, the basic issues and findings also applies to DSRC communication standard.

3.1.1 Analysis and Estimation of Network Queue System Parameters

This subsection describes how the values of the network queue parameters μ_T, A, N related to the transmission of messages on the shared medium are derived. As described in TR 36.885 [79], LTE-V2V supports 10MHz and 20MHz channels, where each channel is divided into frames, Resource Blocks (RBs), and sub-channels. An RB is the smallest unit of frequency

resources that can be allocated to an LTE user. It is 180 kHz wide in frequency (12 sub-carriers of 15 kHz) and one slot in time (i.e. 0.5ms). LTE-V2V defines a sub-channel as a group of RBs in the same sub-frame. Sub-channels are shared among vehicles for the transmission and reception of messages. The number of data bits carried by the group of RBs depends on the chosen Modulation and Coding Scheme (MCS). The number of RBs in each sub-channel for the transmission of messages depends on the available bandwidth and the network configuration. A typical LTE-V2V physical channel of 20MHz bandwidth can support a maximum data rate of 50Mbps (assuming a 16QAM modulation scheme is used). This corresponds to a μ_T with the capacity to process the transmission of approximately 21,000 messages per second, given a safety message size of 300 bytes. With this, the per-byte processing speed is estimated to be $r_T \approx 1.6x10^{-7}$s from equation (3.4).

As mentioned in Chapter 2, PC5 based communication can take place either with or without assistance from the network via an eNB. Hence, RBs may either be allocated to a transmitting vehicle by the eNB referred to as Mode 3 resource allocation or selected autonomously by the transmitting vehicles themselves using a sensing-based scheduling algorithm known as Mode 4 resource allocation. The following discussion applies to Mode 3 resource allocation; however, the results should presumably be the same with Mode 4 resource allocation because the capacity of the shared medium remains the same. The fundamental difference is the time vehicles wait to get transmission resources allocated by the eNB, and the time used by vehicles to sense and select the transmission resources themselves. This affects the fixed value of A in equation (3.2) above.

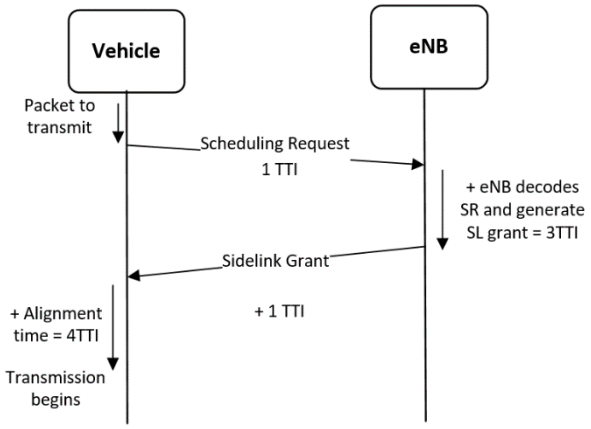


Figure 0.2: Network-assisted PC5 Resource Allocation Procedure

As shown in Fig. 3.2, in Mode 3, a vehicle V_S with data to transmit sends a scheduling request to the eNB and receives an allocation of slots in return. Vs then begins transmission after a so-called alignment time, defined as the waiting time for decoding the received scheduling grants and processing the packets ready for transmission. Time advances in units referred to as the transmission time interval (TTI = 1ms). The total time elapsed between Vs sending a scheduling request and being able to begin transmission is 9TTI, i.e. 9ms. This contributes a fixed increment to the overall delay, i.e. it is part of A in equation (3.2) above. The value can, however, be reduced using a semi-persistent scheduling technique whereby the grant of a resource block allocated to a given vehicle remains valid for a while so that a new request is not needed for every message.

To obtain an upper bound on N vehicles for which μ_T need to process $N\lambda$ transmission on the shared medium, the findings in [80] are used to obtain an initial 'worst case' assumption of 400 for the number of vehicles within the awareness range of a receiver. Combining this with the default message frequency of 10 messages per second per vehicle gives $N\lambda = 4000$ messages per second, meaning that μ_T must be much greater than this figure to avoid the rapid rise in latency and queue length as the singularities are approached. An estimate for μ_T, capable of transmitting 21000 messages per second was given above, yielding $N\lambda/\mu_T \approx 0.2$. However, if the receiver capacity just meets the direct C-V2X requirement mentioned in TS 22.186 [81] of being able to process one message in 2ms, $N\lambda/\mu_R = 8$, which is well outside the region of validity of the queuing model.

3.1.2 Estimation of Sender and Receiver Queue Systems Parameters

In this subsection, the parameter values related to μ_S and μ_R for the maximum message processing rates at the sender and receiver queues respectively, are determined. VPKI-based and TESLA security approaches incur different computational and communication costs per message at both the sender and receiver queue systems because they carry out different crypto operations. This directly affects the service rate or processing resource capacity at both the sender and receiver queue systems. In the case of VPKI, μ_S compute ECDSA signature of a message to be transmitted at the sender, and μ_R conduct signature (ECDSA) verification of N received signed messages at the receiver. In the case of TESLA, both μ_S and μ_R compute symmetric-key MAC of messages. However, TESLA requires an appropriate value of the fixed waiting time interval (δ) to be used so that the vast majority of messages arrive at the receiver

within the same time interval in which they were sent. The choice of δ value is dependent on the transmission latency and the separation distance between sender and receiver. To determine the value for δ, we adopt the estimate presented in a similar TESLA work conducted in [82]. The authors suggested that δ should be slightly longer than the time duration for a message to travel from a source to a recipient at a distance of 1000m in a V2V setting, which was estimated to be 0.010s. Therefore, we set δ to be 0.012s to account for MAC processing times. This value is deemed sufficient as our scenario considers a communication range less than that used in [82]. The δ value appears as an additional contribution to A in equation (3.2) when modelling TESLA.

To obtain values for μ_S and μ_R parameters, a common assumption in 3GPP specifications is that a 3ms processing time is deemed necessary at the user equipment (UE) for processing data packets. This processing time is divided into two: 1ms for encoding data to be transmitted, and 2ms for decoding received data. This corresponds to a lower limit of μ_S and μ_R at the sender and receiver respectively in this model. Reduced values could be achieved with an optimised hardware platform in the UE equipment. For instance, the NXP SAF5400 HSMs [83] and CRATON ATK4100 processor [84] V2X hardware security modules support signature generation and verification times of 0.125ms and 0.5ms respectively, as estimated from their specification sheets. Taking 300 bytes as the basic message size, with additional 194 bytes as the signature length in the case of ECDSA and 32 bytes as the MAC length in the case of TESLA, equation (3.4) was used to estimate the per-byte processing speeds needed to meet the 3GPP C-V2X requirements of $\mu_S \geq 1000$Hz and $\mu_R \geq 500$Hz as $r_S \approx 1.6\text{x}10^{-6}$s and $r_R \approx 2.7\text{x}10^{-6}$s. For simplicity, r_S was taken to be equal to $r_R = 1.5\text{x}10^{-6}$s.

3.1.3 Analysis of queuing theory results

This section presents the results obtained from the queuing theory modelling. The performances of ECDSA and TESLA are compared and analysed. Fig 3.3 uses equation (3.2) to compare the average end-to-end delay as a function of N for ECDSA and TESLA security approaches. Also included is a curve corresponding to the 99th percentile sender and transmission terms for a symmetric cryptography case that was used to derive a value for δ in the case of TESLA, i.e., with this value for δ it is expected that 99% of messages will be delivered before the key used to sign them becomes invalid. It is apparent that while the ECDSA overhead is small at a low traffic density of fewer than 70 vehicles within range of a

given receiver, it increases rapidly when the number of sending vehicles is above 70. At this point, the Nλ/μ ratio approaches 1 for one of the queuing systems, using the fixed value of λ = $10s^{-1}$ that is indicated for most safety applications defined by the standard. The receiver queuing system appears to be the critical one, with a singularity at N ≈ 70 for the estimated values. Notice that the TESLA curve is fairly flat in the range of traffic densities shown, whereas the ECDSA curve increases rapidly above N ≈72, in which case the traffic density is critical and its use becomes problematic. The two cross around N = 65. Below this number of vehicles, the VPKI-based mechanism is preferable, but above it incurs a severe penalty leading to high message delay. Given the many estimates made and the simplicity of the queuing theory model, it is not wise to infer much from the exact numbers.

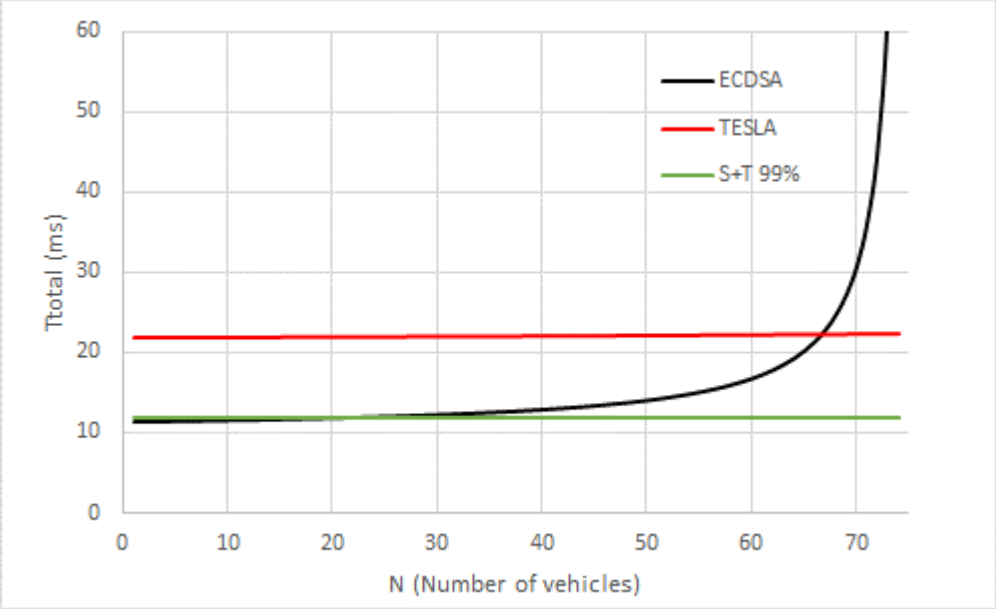


Figure 0.3: Delay from use of VPKI and TESLA security approaches

The characteristics of the curves in Fig.3.3 particularly the point at which the curves rises rapidly depends on the position of singularities. These singularities occur when λ= μ in the case of the sender queue and Nλ=μ for the transmission and receiver queues. For the parameter values estimated above, the positions of the singularities for ECDSA and TESLA can be determined from Table 3.1. The position of the receiver singularity appears to be the most critical, with the curve in Fig. 3.3 rising rapidly as the number of vehicles goes above 74. This is due to the estimated value of μ_R used for the receiver queue. If these parameter values of

μ_S and μ_R for processing messages at the sender and receiver queues respectively are changed, such that the vehicle's processing performance is increased sufficiently, then the shape of the curve changes and the positions of the singularities will also shift to the right of Fig. 3.3. Hence, more messages could be processed at the receiver queue. It would be necessary to increase the vehicle's processing performance to achieve acceptable latencies for city centre traffic densities (N >74). However, the bottleneck will then be the transmission processing rate μ_T, which cannot be altered easily as it is determined by the standard.

Table 0-1: Position of singularities in the three queues for ECDSA and TESLA

Security Approach	**Sender Queue μ_S (msg/sec)**	**Transmission Queue μ_T (msg/sec)**	**Receiver Queue μ_R (msg/sec)**
ECDSA	1039.50	11200.72	747.94
TESLA	1828.15	17170.33	1828.15

Fig. 3.4 uses equation (3.3) to show the average number of messages in the receiver queue as a function of the number of vehicles within the receiver's range when ECDSA and TESLA security approaches are used. It can be observed that with ECDSA, the receiver queue length grows rapidly when there are more than 65 vehicles within the receiver's range. At this point, the singularity when $N\lambda = \mu$ is reached. In the TESLA case, the receiver queue length shows a linear relationship with the number of vehicles. This is because TESLA buffers received messages while waiting for the keys used to sign them, contributing an additional term, N, to the queue length. In the range of vehicle densities shown, this is the dominant effect. The results indicate that ECDSA performs well when there are fewer than 65 vehicles within range. However, as the number of vehicles within range grows above 65, messages start to build up in the queue, requiring more storage space. This can be estimated using $(N\lambda - \mu)*T_{out}$, where T_{out} is the safety message expiry time. Using an upper bound of N to be 400 from [80], μ of 2000 messages per second from the NXP SAF5400 HSMs [83] specification sheet, an ECDSA message size of 494bytes and assuming a safety message expiry time of 1 min, the storage space required is approximately 60MB.

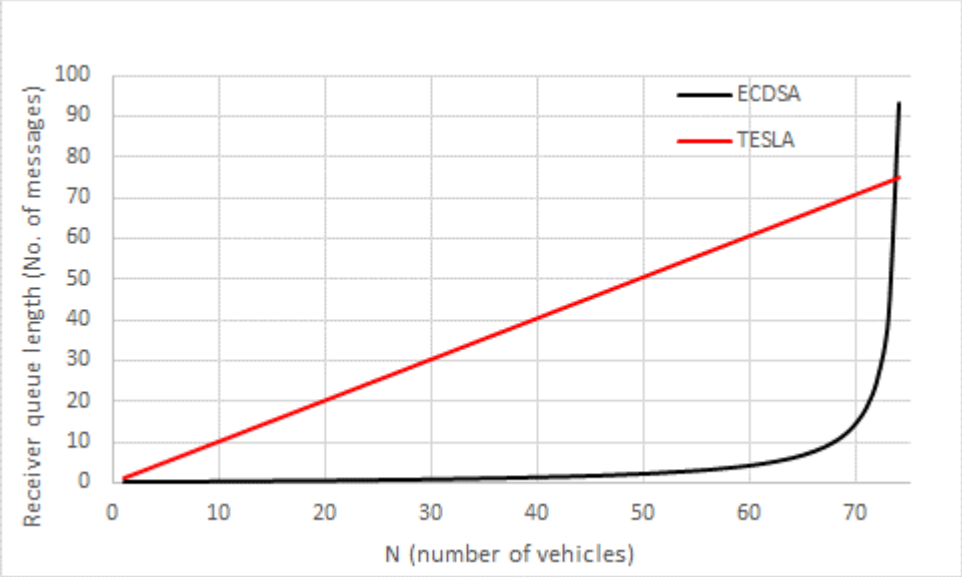


Figure 0.4: Queue length vs N for ECDSA and TESLA

3.1.4 Practical Implications

It is clear from the study that V2V message latency is a serious issue for ITS, particularly for scenarios with high traffic density. For the estimated parameter values used here, the main concern is with the ability of receivers to process messages sufficiently quickly and forward them to the upper layer for use by safety applications. The latency experienced by received messages during verification at the receiver could delay for example notifying a driver with a safety message warning, resulting in potential collisions. Sender performance is of much less concern as a vehicle only processes its own messages when sending, but it receives messages from all vehicles within its awareness range. Given that a receiver can have up to 200 neighbours within its range following the analytical results of [80], high traffic density will be common in V2V scenarios and the performance of ECDSA could be worse in such a situation. One obvious response to ECDSA's performance issue is that computing capabilities are more advanced year by year, and with extremely powerful processors on-board, vehicles can process messages at a much faster rate. But doing that will delay the problem until such computational power is readily available on vehicle on-board units. Also, as mentioned in [85], vehicle on-board computation is limited due to the requirements of small-scale and low-cost hardware to make V2V and ITS in general economically viable. Even with extremely fast processors, there could be other things that would still be of concern. For instance, If we consider other choices of parameter values for processing messages at the vehicle to say

$1.8x10^{-7}$s, a value more than the per-byte processing speed of the transmission medium. In this case, the receivers would be able to process messages sufficiently quick, but the shared transmission medium will now become the bottleneck. This is a more serious problem as network performance is determined by standards rather than by equipment.

It is apparent that it would be easy for a malicious agent to exploit ECDSA's computational overhead to launch DoS attack regardless of the traffic condition. For example, an attacker with a V2V-enabled device can position himself in an area of interest (e.g. road intersection) and simply increase its message generation rate, thereby sending signed messages more frequently than the rate specified by the standard. This results in an increase in the average arrival rate of messages to vehicles that are within the attacker's vicinity, and hence their queue length grows rapidly as does the waiting time. Given that a receiver needs to verify the ECDSA signature of every message in its queue, this results in the receiver wasting its resources verifying signatures of bogus messages, making genuine messages from surrounding vehicles stay longer in the queue or even get dropped. The consequences of such action become more severe when there is more than one attacker within the vicinity of a receiving vehicle.

TESLA's use of symmetric key cryptography means that the singularities occur at much higher traffic densities as indicated in table 3.1. However, it suffers the penalty of the receiver needing to wait for receipt of new keys to allow the preceding time interval's messages to be verified. This penalty is a fixed overhead that does not depend on λ or N, so that TESLA wins out eventually over ECDSA at high traffic densities and/or message frequencies. Based on our estimated parameters, the cross-over point occurs within the traffic density range of interest for V2V communications. Much of TESLA's fixed overhead is due to the 9ms required by the LTE network-assisted resource allocation procedure. This affects all options examined, of course, but TESLA suffers a double dose. There is a mechanism (known as semi-persistent scheduling, SPS) that avoids the need for a sender to request a new resource allocation for each message that should reduce the average delay. It has not been taken into account in the results presented here, and if the reduction is significant it would render TESLA more competitive with ECDSA. The 9ms will cause problems in any case for next-generation use cases such as Vehicle Platooning and Autonomous Driving that possess latency requirements of 10ms and 1ms respectively. However, there are latency reduction techniques currently

being considered by 3GPP to reduce the delay of obtaining resources. These include fast uplink access on the MAC layer, the concept of short TTI on the physical layer [86].

3.2 Vehicular Network Simulation

The analytical model developed above provides an intuition about the issues with the ECDSA security approach under dense traffic conditions. However, it assumes that the communications among vehicles are reliable; every message sent is successfully received at the receiver and is correctly decoded. This is not the case in a realistic V2V setting, as wireless communications are unreliable and are prone to several transmission errors including propagation effects on the wireless medium, packet collisions and interference from other transmitting nodes. In addition, the probability of message reception is a function of the separation distance between sender and receiver. To get a more realistic V2V setting, a simulation study is required that models and implements the protocols and communication standards defined by ITS standard bodies for transmission and reception of messages between vehicles and also consider the characteristics of the wireless radio links like attenuation, fading etc. It should also provide realistic vehicular movements on real road networks under a wide range of transmission parameters and traffic densities. The intuitions obtained from the analytical work can then be validated with quantitative and more accurate results from simulation.

Some of the existing research works that use these simulation tools to investigate the impact of ECDSA security overhead on V2V safety messages are as follows; [87] uses OPNET, [88] [89] [90] and [91] uses NS3, studies in [92] and [85]uses OMNet++, and the work in [93] uses NS2. However, none of these simulators has yet implemented the standard ECDSA security. Researchers develop their own ECDSA security model using cryptographic library extensions and attach it to the respective simulator. In all of the above-mentioned works, no reason was stated for chosen a given simulator, rather it is left as a personal preference or on the familiarity with the coding language. Nevertheless, NS3 appears to be used more frequently than the other network simulation tools.

3.2.1 Network Simulation

V2V simulators typically combine a network and a mobility simulator. Network simulators are responsible for modelling communication protocols and the exchange of messages among vehicles, whereas mobility simulators generate the movement of vehicles in accordance with the surrounding environment and determine the positions of vehicles in a network simulator. Examples of network simulators widely used in V2V research studies include NS3 [94], OMNet++ and OPNET. As for mobility simulators, Simulation for urban mobility (SUMO) [95] [96] is commonly used.

NS3 was used as a basis for the network simulation for the following reasons:

- it has no limitation regarding the number of nodes for a configured simulation, making it capable of to simulate different levels of traffic density;
- it models the complete WAVE stack and currently supports the PHY/MAC layer implementation of both 802.11p and direct C-V2V standards;
- it has an open-source library, which allows users to easily extend the implementation details or build their own custom extensions, protocols and applications;
- it can support the implementation of new technologies penetrating the V2V domain such as SDN, edge computing, etc.

The C-V2V Mode 3 stack implementation in NS3 was used to allocate radio channels to vehicles to transmit their messages. The application layer stack was configured such that the transmitting vehicles generate safety messages every 100ms. A custom security module was built and implemented in NS3 that handles the signature generation and verification of messages. The module is interfaced with the Network/Transport layer as defined by the ITS standard. At the sender side, each safety message is passed to the security module for signing using either the TESLA or ECDSA scheme, after which the message is placed in the secured message format, and then encapsulated within the C-V2V MAC packet. The message is then forwarded to the access layer for transmission on the allocated radio channel. At the receiver side, each received message is placed in a queue while waiting to get verified at the corresponding security module, after which the message is returned to the networking layer, and then forwarded to the application layer as illustrated in Fig. 3.5.

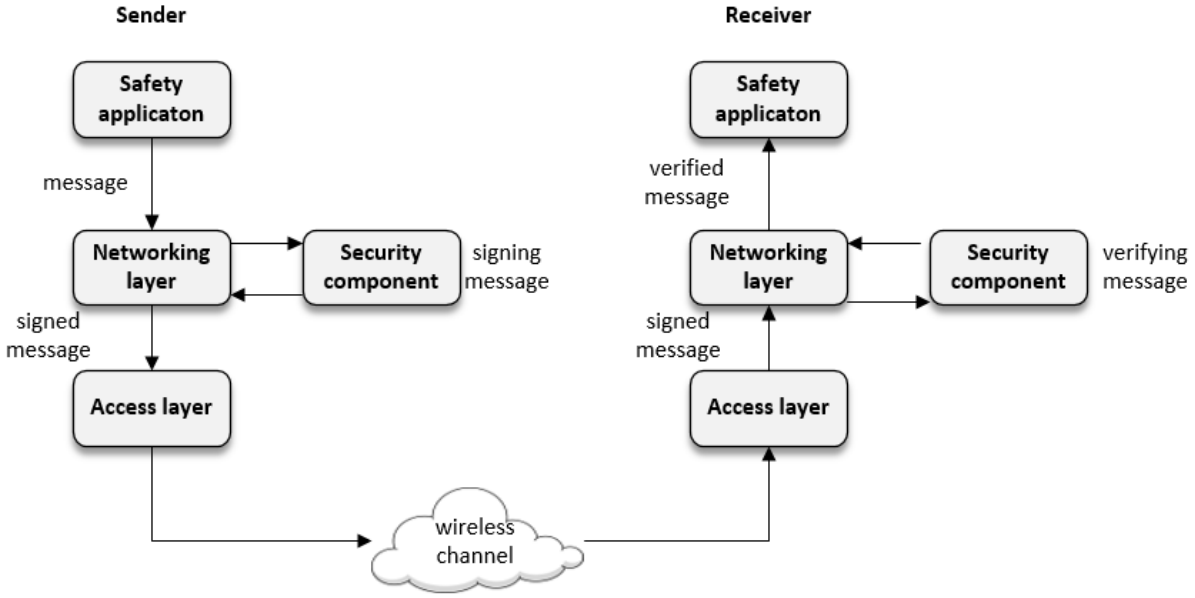


Figure 0.5: Secure message transmission procedure

The parameter values and configuration are summarised in Table 3.2. Where relevant, values have been chosen that are consistent with the queuing theory model.

Table 0-2: Simulation Settings

Simulation Settings		**Values**
Network	Channel bandwidth	20Mbps
	Transmission power	23dBM
	MSC and coding rate	QPSK, 0.7
	Packet transmission frequency	10Hz
	receiver sensitivity threshold	-85dBm
	Propagation model	WINNER+ B1
Security	ECDSA signing delay	0.125ms
	ECDSA verification delay	0.5ms
	Message size with VPKI	494bytes
	Message size with TESLA	332bytes
	TESLA time interval	100ms
	Hash or MAC operation	1µs
	TESLA δ value	12ms

3.2.2 Implementation of the VPKI-based Security Solution

Both IEEE 1609.2 and ETSI ITS security standards recommend the use of *ecdsa_nistp-256_with sha256* (i.e. Elliptic Curve Digital Signature Algorithm NIST P-256 with Secure Hash Algorithm-256 curve) for signature generation and verification of messages. There are two possible ways to implement the ECDSA security procedure in a simulation environment (a) to use a software implementation of *ecdsa_nistp-256_with sha256* algorithm available from cryptographic libraries like OpenSSL, Crypto++, and calling the ECDSA signature generation and verification functions each time a message needs to be sign or verified, and (b) to use the ECDSA signature generation and verification processing times of a V2X hardware security module and introduce the process delays in the security extension. In the case of (a), the execution times of the software implementation depends on the computational power of the machine used for the simulation. Existing research works that use an ECDSA software implementation reports different values in their evaluation. For example, [97] reported ECDSA execution times for signing and verifying messages to be 0.5ms and 4.2ms respectively, using OpenSSL running on a 1.5 GHz Centrino Processor. For the same OpenSSL algorithm and implementation, the work of [98] shows the execution times to be 2.339ms and 2.8676ms on an Intel Pentium 1.46GHz with 3GB memory. Therefore, the performance of a pure software implementation may not reflect real-world use-cases. Using the processing times as reported in the specification sheet of commercially available V2X hardware security modules is more realistic since these modules are specifically designed for processing in the vehicle's OBU. Similar to the analytical case, the same ECDSA signature generation and verification of 0.125ms and 0.5ms respectively for the NXP SAF5400 HSMs [83] and CRATON ATK4100 processor [84] V2X hardware security modules were used in the simulation. A processing delay of 0.125ms was added using a timer tick to each outgoing message from the sending vehicle. At the receiver side, every message received is placed in a security First-In-First-Out (FIFO) queue and verified at its turn by delaying it for 0.5ms using the same timer tick. In addition, the size of a message signed with ECDSA is increased by 281bytes to represent the addition of the ECDSA signature and the senders' certificate.

3.2.3 Implementation of TESLA-based Security Solution

TESLA signatures are MACs, which are generated using cryptographic hash functions (e.g. SHA). In this simulation, the vehicles divide their timelines synchronously into 100ms intervals

and assign one TESLA key (from a pre-computed key chain) to each time interval. Each vehicle broadcasts only one message within a time interval and uses the assigned TESLA key to sign the message. The key used to sign a message in a given interval is disclosed by the sender after a fixed time interval (δ), which is set to be 12ms, as mentioned above and taken from [82]. Thus, the receiver waits for at least 12ms before being able to verify the message. Messages are verified once the receiving vehicle checks the correctness of the disclosed key. This is done by validating it with the sender's chain commitment key. As TESLA uses a hash chain to compute the signing keys, the receiver is allowed to use any following key to verify previously received buffered messages from the same sender. Thus, a message can be verified by not just the intended next key but also by keys released later. As the focus of this simulation is on the overhead associated with message authentication, distribution and verification of key chain commitment are not considered here. Rather, it is assumed that all vehicles have each other's current commitment keys. The distribution of chain commitment keys is a separate issue that will be addressed in the next chapter.

3.2.4 Simulation Results and Discussion

The performance metrics used to evaluate the impact of using ECDSA and TESLA security solutions in the simulations are average message delay and message loss ratio. In addition, other performance metrics such as packet delivery ratio, the effective value of N, are introduced. Results obtained through simulations are also used to validate the results obtained in the analytical work.

There are two phases of simulation scenarios considered in this work. The first phase is a scenario with static vehicles that only models the behaviour of broadcast message communication. The second phase is another scenario that considers the realistic movement of vehicles with the exchange of messages between them. To validate and compare the analytical work with the simulation, we initially consider the static scenario. This scenario closely matches the analytical settings so that the two sets of results can be compared. In the static scenario, the transmission and reception of secured messages among vehicles is simulated using NS3. A receiving vehicle is placed at the centre of a 50m radius circle and all other vehicles are uniformly distributed around the circle. A small separation distance of 50m is chosen to ensure reliable communication between the vehicles and to minimise messages dropped due to transmission problems such as signal attenuation, fading, and channel losses.

In this case, a high proportion of messages sent are likely to be received at the receiver, which makes it comparable to the analytical case. To generate different traffic density situations, the number of transmitting vehicles around the receiver was varied from 1 to 100.

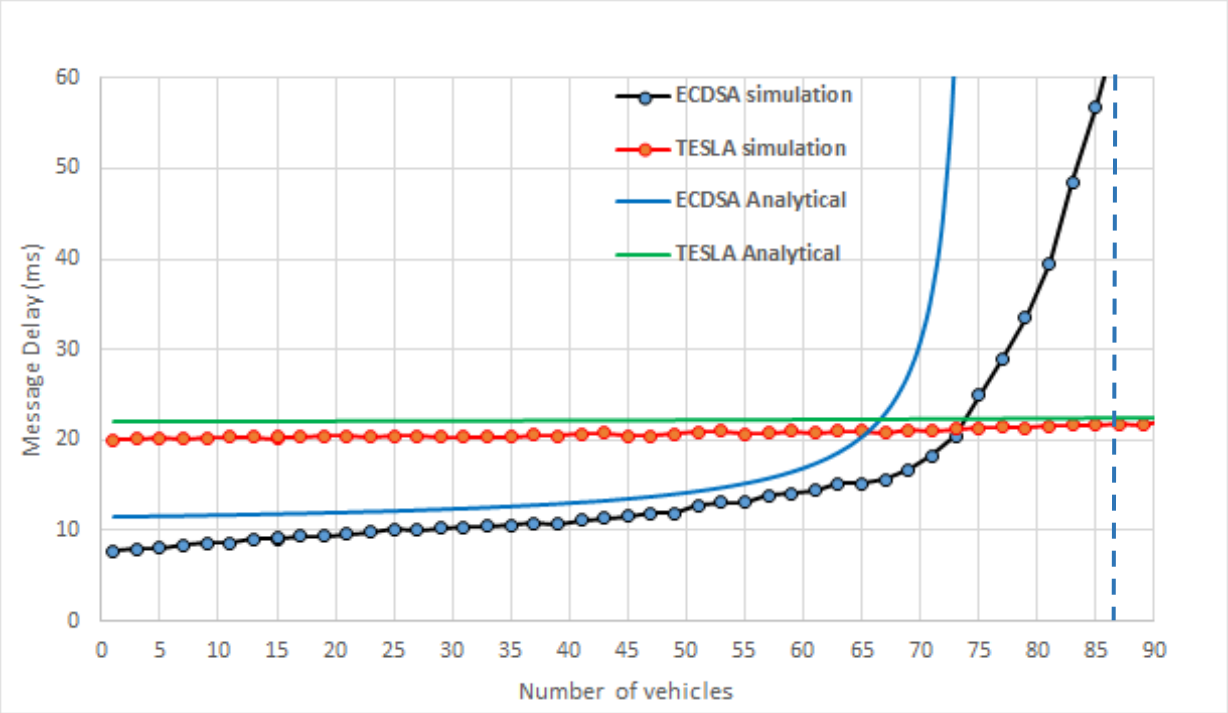


Figure 0.6: Latency overhead from the use of VPKI and TESLA Approaches

Average Message delay: Fig. 3.6 compares the average message delay as a function of the number of vehicles for ECDSA and TESLA security approaches obtained through simulations and the analytical work. It can be observed that the TESLA curves are similar; the average delay obtained with the analytical work closely matches those obtained by simulation. The two results confirm that TESLA outperforms ECDSA when there are more than 65 vehicles within range and validates the behaviour of TESLA in terms of message delay, with the dominant factors made up of the fixed waiting time to receive the disclosed key and time spent during the allocation of channel resources to the transmitting vehicles. For ECDSA, both the simulation and analytical curves show that the average message delay is less than 15ms with fewer number of vehicles around the receiver. The message delay increases rapidly when there are about 50 vehicles within range of the receiver in both the simulation and analytical curves. Our results are also comparable to those obtained in the works of [16] where the

authors investigate the impact of ECDSA based security on safety messages in dense traffic scenario. They use NS3 to simulate a high-density scenario with a total of 1800 vehicles in a 3km by 3km grid and set ECDSA generation and verification times to be 0.26 ± 0.012ms and 1.22 ± 0.047ms respectively, on the safety messages. Their result shows that for senders that are 50m away from a given receiver, the average message delay varies from 80ms up to 140ms. This further validates our simulation as well as the analytical results.

Looking at Fig. 3.6, there are notable differences between the two curves – (a) the displacement of the receiver singularity position to the right in the simulation case and (b) the vertical separation between simulation and analytical curves. The position of singularity occurs around N=75 in the analytical case, whereas it is displaced to the right in the simulation. The reason for this discrepancy in the singularity positions between the two curves is message loss during transmission due to channel losses or transmission errors, which is taken into account in the simulation, but not in the queuing theory model. This effectively reduces the value of N at the receiver. To confirm this, the probability of successful packet reception at different separation distances and values of N is investigated below. Concerning the vertical separation between the two curves, this could be due to MAC layer delay or the process of allocating transmission resources to sending vehicles by the scheduler in the eNB. In the analytical work, an estimated value of 9ms was used as the scheduling delay for all vehicles. This could be underestimated or overestimated depending on the traffic condition. In the case of simulation work, a scheduler in the eNB allocates the shared RBs to sending vehicles every scheduling cycle (TTI) based on the availability of RBs and the location of requesting vehicles. As ECDSA messages are larger and hence occupy more RBs, some vehicles may not get an allocation if there are insufficient RBs in that scheduling cycle. This results in messages been delayed while waiting for the next scheduling cycle. Such delay increases with the number of transmitting vehicles. Also, the scheduler may refuse to allocate adjacent RBs or reuse the same RBs to closely located vehicles to avoid creating reciprocal interference. All these factors could contribute to the differences between the ECDSA's simulation and analytical results.

Packet Delivery Ratio: PDR is defined as the ratio of correctly received messages to the number of transmitted messages and varies with distance between the sender and receiver. It provides an indication of the radio link quality and effective communication range. To

estimate PDR, simulations were run with N=1, 10, 50 and 100 simultaneous senders at separations from the receiving vehicle varying from 50-500m. The PDR was measured by averaging the PDR of N sending vehicles at the receiver. Fig. 3.7 shows the results. As expected, the probability of successful packet delivery goes down with increasing separation and with increasing N. The transmission range is not sharply defined i.e., a nearby vehicle is not guaranteed to receive a message and it is possible that a distant vehicle may receive one. Thus, the PDR values can be applied as a correcting factor to N in the receiver terms of equation 3.3 to take message loss into account. From Fig. 3.7, at a 50m separation and N=75 a PDR value of about 0.86 can be expected. Applying this value to the receiver queue term, moves the singularity position from N=75 to N=87 as marked by the dotted straight line in Fig 3.6, bringing the queuing theory model into better agreement with the simulation results.

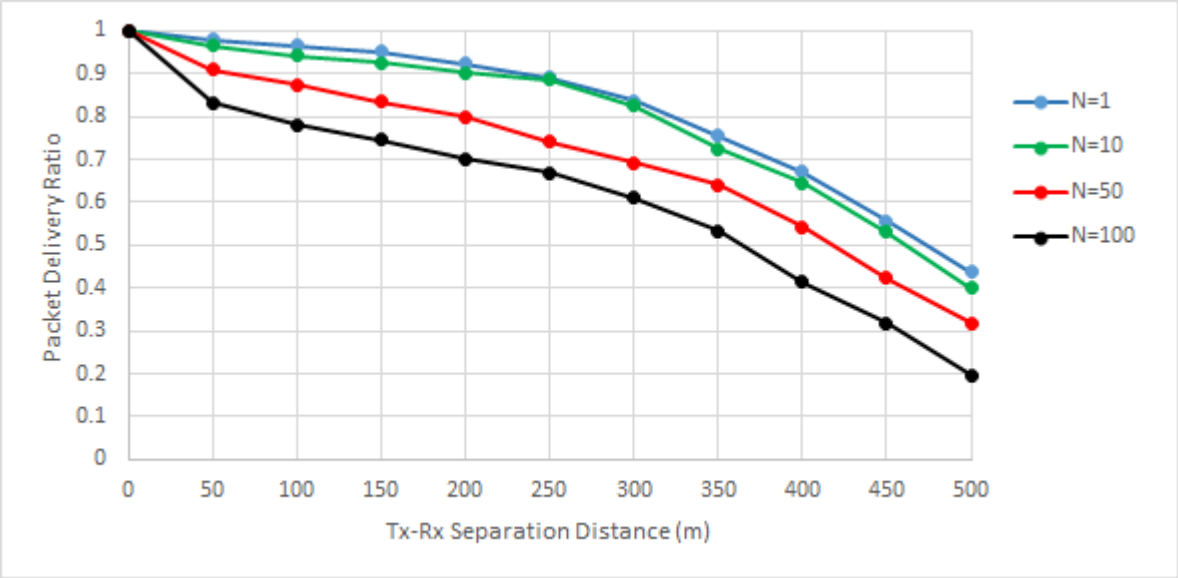


Figure 0.7: PDR as a function of the distance between sender and receiver for different N values

In the above scenario, the PDR was measured with all transmitting vehicles being located at the same distance to a given receiver. However, in a realistic V2V setting, a given vehicle will have neighbours within its vicinity distributed at different positions at any given time. In addition, since the PDR depends on the separation distance between sender and receiver as well as the number of sending vehicles, it is important to investigate the effective value of N

at a given receiving vehicle when vehicles are evenly distributed. This will show the possible number of messages per unit time that the receiver is likely to receive.

The effective value of N: In this simulation, a varying number of sending vehicles are distributed randomly over an area of size 1000mx1000m. A receiving vehicle is placed at the centre, and the simulation runs for 200 simulated seconds. To measure the effective value of N, the rate of arrival of messages at the receiver was counted divided by λ during each run. Fig. 3.8 shows the different effective values of N for different vehicle densities in the simulated area. As the transmission range is not well-defined, so neither is the number of vehicles within range. The effective N is an analogous parameter that is not dependent on range, but appears to be linear with increasing traffic density. It can be observed with 100 vehicles per km^2, each located at a different position to the receiver, the effective value of N is around 85. This means that at this vehicle density, the singularity of the queuing theory receiver term will be reached, and as this value is approached, it is expected that the messages would start to experience verification delay of more than 60ms or may subsequently get dropped. Above the 100 vehicles per km^2 density, the use of the ECDSA security approach would face a serious problem.

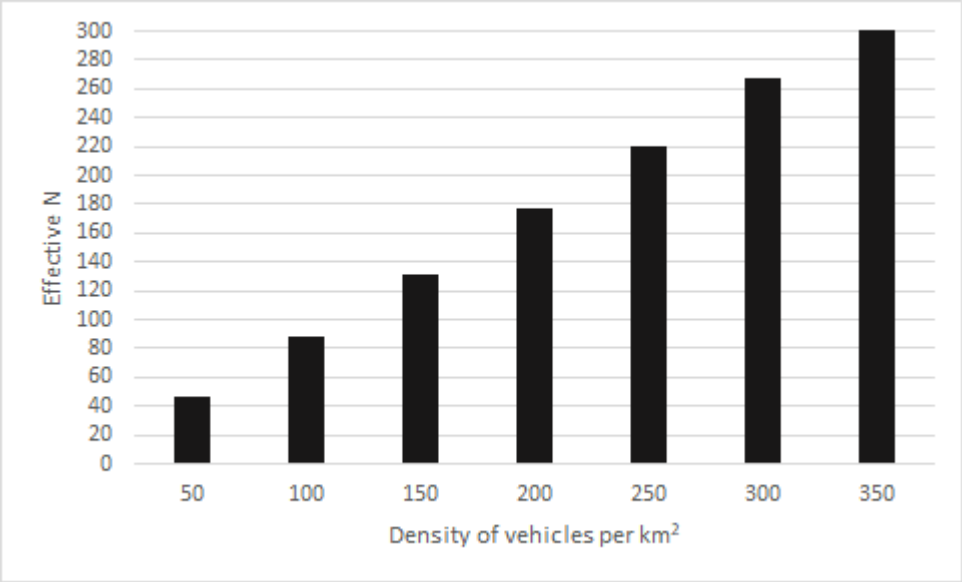


Figure 0.8: Effective value of N

Message loss ratio: The analytical model and the simulation used so far assumes infinite queues, which is obviously not realistic. In reality, messages would be dropped when queues reach their limit or messages become out of date as safety messages have a finite lifetime.

Thus, in the simulation, we investigate the performance of ECDSA and TESLA in terms of the percentage of messages dropped due to cryptographic overhead referred to as message loss ratio. This is expressed as the ratio of messages dropped from the receiver's queue to messages successfully verified at the receiver. In this, the receiving vehicle queues up received messages in its buffer while waiting for verification. A finite message lifetime value of 100ms is set, such that if a received message is not verified within this time, it is considered out of date and then dropped from the receiver's queue. Fig. 3.9 shows the message loss ratio as a function of the number of sending vehicles for both ECDSA and TESLA security approaches. As expected, the curves indicate a similar trend with those obtained for message latency, with ECDSA showing superior performance over TESLA when there are fewer than 70 sending vehicles within range of the receiver and runs into a problem when the number of senders goes above 70.

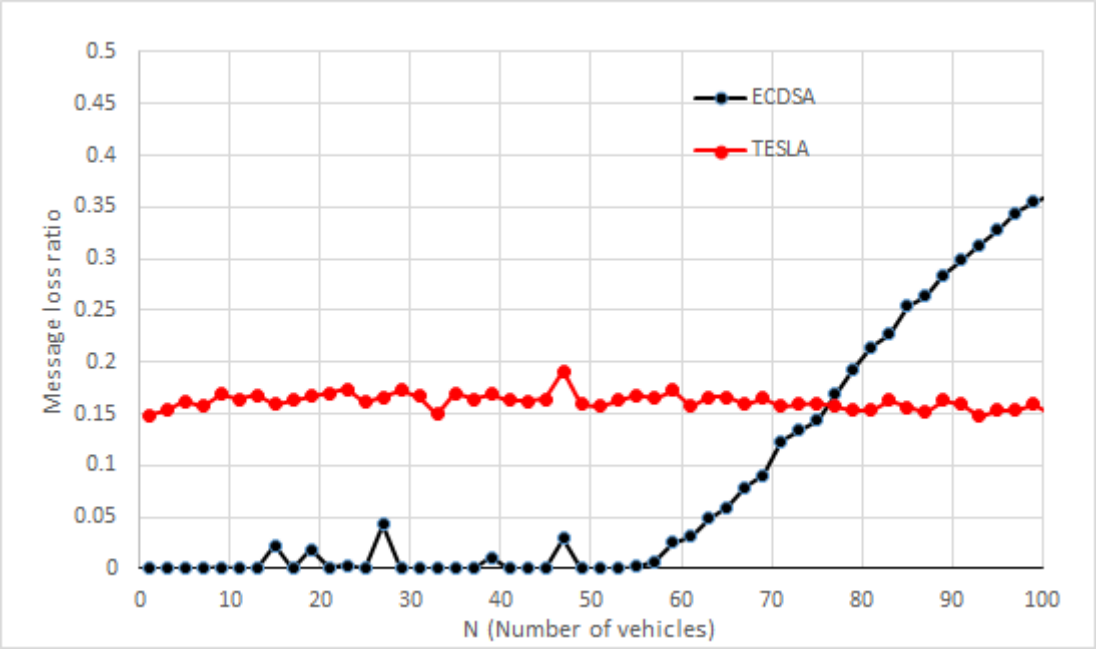


Figure 0.9: Message loss ratio

At N less than 65, ECDSA achieves a message loss ratio of less than 0.05%, which means it verifies nearly all messages in the receiver's queue because the rate of message reception is much less than the message processing rate. However, as the number of sending vehicles exceeds 65, more messages are queued for verification, and consequently, the time spent in the queue grows due to ECDSA's high message processing time. This results in a rapid increase in message loss ratio as more messages become out-of-date before they reach the head of

the queue, get verified, and forwarded to the relevant safety application. This effectively reduces the message delivery rate and could degrade the performance of safety applications that require frequent and fresh updates of messages in order to closely monitor vehicles' dynamic changes, particularly in critical traffic safety events. TESLA maintains less than 20% of message loss regardless of the traffic density and the time spent in the queue waiting for the arrival of the disclosed key. The reason for TESLA message loss is mostly due to key messages that are lost in transmission. With TESLA, the majority of received messages are verified and forwarded to the application layer before their deadline time.

3.3 Simulation with Realistic Vehicle Distribution

This section investigates whether traffic densities in a realistic V2V environment are likely to be high enough for ECDSA to encounter the problems identified above.

The simulation is conducted on a city centre road network of size 3km by 3km, as illustrated in Fig. 3.10. The RandomTrip tool in SUMO is used to generate a set of journeys to be taken by vehicles on the road network. The start times of the trips are distributed evenly in an interval between 0 to 13,000 simulated seconds, but a minimum trip length of 5000m is set in order to ensure vehicles stay in the scene and run for a longer period. The vehicles are introduced into the simulated area with the help of the so-called p-factor, which generates vehicles with a constant period and arrival rate. This is set to 0.3 seconds in order to create a high traffic density state and to maintain the congested state to the end of the simulation period. Vehicles move according to the Krauss car-following model that maintains a safe distance between a vehicle and its vehicle in front and selects the speed of vehicles so that vehicles can stop safely and avoids rear collision. A total of 2190 vehicles runs during the simulated time with an average trip length of 8149m. The vehicle mobility traces generated by SUMO was then imported into NS3 platform to simulate communications between the vehicles. The simulation run again with vehicles broadcasting safety messages every 100ms. The steady state vehicle density resulting from the SUMO simulation is indicated as a heatmap is generated and overlaid on the street map in Fig. 3.10.

Figure 0.10: Simulation Road Network with Heat Map indicating high vehicle density hotspot

The vehicle distribution generated using SUMO was then used as input to the NS3 simulation. We measured the approximate number of messages that each vehicle received from other vehicles within its reception range during the simulation. Fig. 3.11 shows the distribution of message reception rates over the population of vehicles. That is the number of messages that each vehicle received at its physical layer. It can be seen that about 42 vehicles did not receive any message, and up to 167 vehicles receive above 10,000 messages per second. Also, about 200 vehicles receive 6000 messages per second, which represents the mode of the distribution. As indicated on NXP SAF5400 HSMs [83] and CRATON ATK4100 [84] specification sheets, the security modules can support to up 2000 ECDSA (using NIST P-256 elliptic curve) verifications per second of incoming messages. However, looking at Fig. 3.11, only 78 vehicles receive messages below this rate. This amounts to less than 10% of the total number of vehicles that received messages during the simulation. Close to 90% of the vehicles receive more than 2000 messages per second in the given simulation scenario. This indicates that using ECDSA security approach in verifying the signature of every received message would result in a large proportion of the messages received experiencing a longer delay or eventually get dropped. Processing a greater number of messages has obvious benefits for safety

applications especially those that require tight coordination and awareness between vehicles. Therefore, the impact of using ECDSA on the performance of safety applications in such an environment is questionable. In similar existing work of [85], the authors measure the number of messages a given vehicle could receive in a highway scenario consisting of 12 lanes, where each lane is 3m wide, and vehicles are uniformly placed on the lanes with an inter-space gap of 30m. Their results show that vehicles that are located in the middle of the highway receive 224 messages every 100ms.

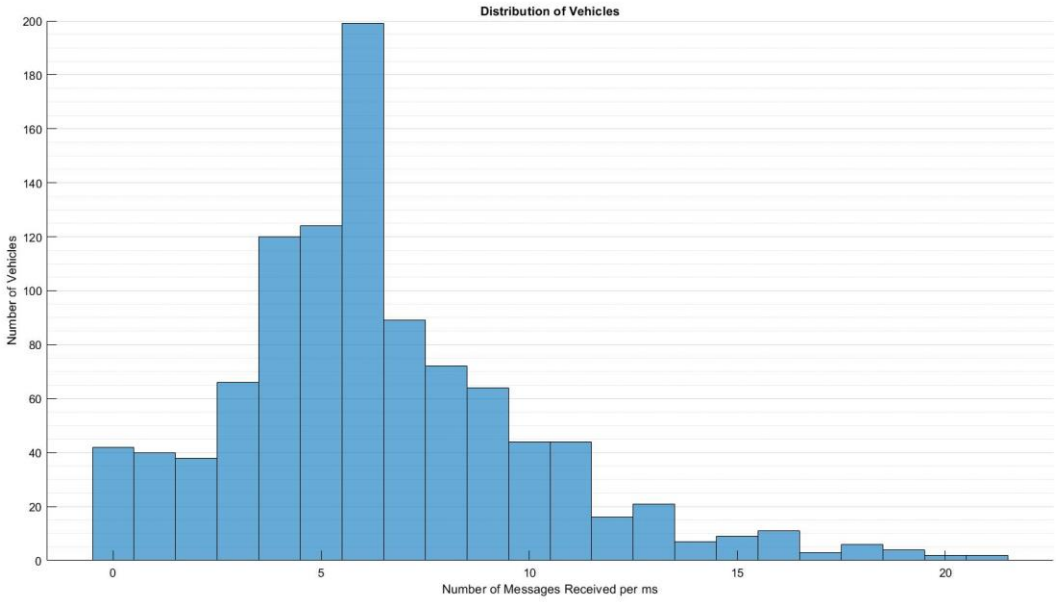


Figure 0.11: Distribution of Vehicles

3.3.1 Comparison with Existing Simulation Works

In this section, we compare our results with those found in other simulation studies that evaluated the performance of ECDSA in V2V.

[88] compares the message loss ratio when using two different processors at the receiver: a processor that can sign and verify messages at 2ms and 5ms respectively, and another higher capacity processor that can sign and verify messages at 0.22ms and 1.22ms. They set the message lifetime to be 100ms. Their results show that ECDSA achieves a message loss ratio of less than 0.05% with the high capacity processor and loss increased to 36% with the low capacity processor. However, the authors did not investigate the effect of increasing traffic

density on the message loss ratio. In the simulation model used in [99], their results show that the position of receiver singularity occurs at around a density of 30 vehicles per km per lane using ECDSA sign and verification times of 2.5ms and 4.97ms. In another work [89], the results indicate that a receiver can verify messages from up to six vehicles within its range. Above this number, the messages could be lost because the receiver singularity is reached. However, they used an ECDSA signing and verification times of 7ms and 22ms respectively. Similarly, the works of [87] evaluate ECDSA's performance using signing and verification times of 1.52ms and 4.14ms respectively. Their results show that the average message delay of ECDSA is low when the number of vehicles within range of a given receiver is less than 22 and increases rapidly when the number of vehicles becomes larger than 22. To validate our analytical model results with these existing works, we use the same μ_S and μ_R parameter values of [89] and [87] in our analytical model. Our ECDSA curve changes with receiver queue position of singularity now occurring at N≈5 and N=20 for the parameter values used in [89] and [87] respectively. This indicates that the analytical model results obtained in this study are consistent with those found in the literature. In addition, our assessment of ECDSA's performance under a high traffic density scenario is more realistic than existing works because we use parameters reported in the latest vehicle's OBU. Our simulation and analytical results show that a receiver can verify ECDSA-based messages from up to 74 vehicles within its range, whereas others indicate inferior performance with a fewer number of vehicles that a given receiver can support.

3.4 Discussion

Enabling safety-critical applications in V2V requires frequent message exchange between vehicles. It is clear from the requirements of these applications and their use-cases that V2V message latency is a serious issue. In practice, the timeliness and freshness of dynamic data contained in a received message are essential for cooperative safety applications particularly in critical traffic safety situations where vehicles need to monitor continuously and more precisely step changes in each other's kinematics. For instance, in the case of longitudinal collision risk warning (LCWR), which is targeted at reducing the risk of collision, one important quality parameter for receiving vehicle to be concerned with when assessing the risk of collision to the sender, is the age of data contained in the received message, in particular, the

age of highly dynamic data. The receiving vehicle may take an erroneous decision if the received data does not correspond to the latest situation of the sender e.g., position, speed. The performance of such applications depends on the freshness as well as the accuracy of the information contained in the received messages.

The security functionalities applied to ensure the authenticity and integrity of received messages in V2V have an impact on the overall system performance in terms of processing time and processing capacity. Such impact may not be negligible for the operation of critical safety applications. The latency requirements of these applications should be met in order to provide safety measures in a timely and efficient manner. The outcome of both the simulation work and analytical modelling based on queuing theory demonstrates that ECDSA has performance issues in V2V scenarios with high traffic density. In these scenarios, vehicles receive thousands of safety messages from their neighbours. The results obtained indicate that the position of singularity at the receiver queue occurs for a value of N greater than 75. At this point, the message arrival rate is more than the processing rate, and therefore the performance of ECDSA becomes worse beyond this traffic density. With more messages received, the receiver queue grows and messages experience delays of than more the 100ms maximum tolerable latency of the safety applications that uses them. The more the messages are delayed or dropped occasionally during the verification stage, the wider the interval time between message updates to the relevant safety applications. Thus, the freshness of highly dynamic data of neighbourhood traffic is reduced, and consequently impacting the awareness between vehicles as well as the performance of critical safety applications. Therefore, the performance degradation of ECDSA security scheme in high traffic scenarios poses a safety risk.

The ITS standard recommends the use of a congestion management technique referred to as decentralised congestion control (DCC) [100] to throttle message generation frequency in a congested traffic state. In the case of congestion, the DCC mechanism causes vehicles to reduce their message generation time interval in order to reduce the number of messages sent to the network. Also, it requires vehicles to dynamically adjust their transmission power based on the vehicle density situation. If high traffic density is perceived, the vehicles should use a low transmit power to broadcast their messages in order to reduce the reception range and consequently the effective value of N. However, this approach is more concerned with

controlling congestion at the network level, i.e. on the radio channel. As previously seen from the results obtained, the receiver queue term approaches singularity before the network transmission queue system. In addition, the DCC mechanism may have severe consequences on cooperative safety applications that require an updated view of the surrounding environment. For example, safety applications like LCWR can issue a specific request to the cooperative awareness (CA) service in the facilities layer responsible for generating safety messages, to adjust the message generation interval based on the perception of the criticality of traffic safety situation around the vehicle. The application may request that the message generation interval should remain at its lowest i.e. 100ms or even lower. Such request overrides the DCC's rules and must be obeyed. Therefore, the use of DCC may not be effective in this situation. In addition, from a security perspective, a malicious node may refuse to cooperate and continue using its message generation interval or even broadcast messages at a higher rate.

Other studies in the literature proposed mechanisms to verify relevant messages in order to alleviate the load on the receiver and to reduce the latency experienced by delivered messages in high traffic conditions. Given the safety-relevance of many V2V applications, it seems unwise to leave it to chance to decide which messages get through and which do not. Consequently, there is a need for an adaptive mechanism that decides which messages go through and which do not. Such a mechanism should be able to decide the relevance of the message to the receiver based on, for example, the distance between the sender and receiver. Messages from vehicles closer to the receiving vehicle are more likely to influence the receiver's behaviour than messages sent by vehicles that are far away. Alternatively, the mechanism could drop messages based on threat level such that messages that pose a security threat to the receiving vehicle are allowed to get through. However, such a mechanism will add computational overhead to the receiving vehicle.

3.5 Summary

V2V communications have the potential to greatly increase safety on roads and improve the driving experience, but the safety messages exchanged among vehicle need to be protected. This is essential to prevent malicious actors from broadcasting messages with false information that could potentially lead to wrong decision-making. The ETSI ITS and IEEE ITS working groups require the use of ECDSA algorithms to ensure the authenticity and integrity of V2V messages. However, the ECDSA mechanisms come with overheads that affect the performance of the V2V communications, and that of the safety applications. In this chapter, we have used queuing theory and simulation to compare the performance of the standard, VPKI-based, message authenticity and integrity protection mechanism with that of an alternative scheme, TESLA, which uses symmetric-key cryptography combine with hash chains.

Perhaps the most significant observation from a security perspective arising from this study is that the VPKI-based security solution results in high message latency and dropped messages under conditions of high traffic density. In settings such as busy city centres, a vehicle is expected to receive up to 2000 messages per second from its neighbours configured with a half-duplex radio. With a full-duplex radio configuration, as indicated in previous studies, even more messages are expected to be received. In such conditions, the VPKI-based security solution incurs a message latency of more than 100ms, exceeding the fixed waiting time in TESLA. This exceeds the critical latency of the V2V messaging system, ultimately impacting the safety of ITS users. Moreover, an attacker can exploit these issues to carry out a denial-of-service attack. Regardless of traffic density, an attacker can simply increase its message transmission rate to overwhelm the receiver's resources, forcing it to drop messages.

The analytical and simulation results presented in this chapter indicate the need for an alternative to VPKI and demonstrate that schemes based on TESLA are viable candidates. The next chapters investigate how to make TESLA more effective in V2V by addressing its limitations.

Chapter 4: Commitment Key Distribution in V2V

The performance evaluation studies presented in the previous chapter indicate that the use of VPKI-based security schemes such as ECDSA to provide authenticity and integrity for V2V safety messages increases the message latency in moderate to high traffic density situations to a level greater than what is acceptable for time-critical safety applications, and that TESLA offers a potentially-viable alternative. However, one of the core requirements of TESLA-like schemes is the need to distribute authentication information (known as commitment keys, or simply commitments) that is used to verify the keys used to authenticate broadcast messages. This is a problematic issue in the V2V context due to the mobility of vehicles. In this chapter, two novel schemes are presented, which can be employed to manage the distribution of commitments to vehicles efficiently. In addition, the chapter provides details about the use of a realistic V2V simulation to demonstrate how these schemes would be implemented in practice. Finally, the comparative analysis and performance evaluation of two schemes using the analytical and simulation results is discussed. The performance of a related solution found in the literature is also analytically evaluated and compared with the two schemes.

4.1 Problem Statement

The distribution of commitment is an issue in V2V environment due to the large population of moving nodes and short-range communications, resulting in a highly dynamic network topology. Vehicles need to know the commitments of vehicles they are likely to receive messages from for message verification to take place. Suppose vehicle V_i broadcasts a message at time T and discloses the corresponding key at time T+δ, where δ is the key disclosure time interval. A receiving vehicle, V_j, wishing to make prompt use of the message will need to be in possession of the commitment for the key chain containing this key before T+δ. We say that a vehicle V_i is relevant to vehicle V_j at time T if knowledge of V_i's state at T could potentially influence J's behaviour at T+δ. For simplicity, we make the reasonable assumption that if V_i is relevant to V_j, then V_j is within the range of messages broadcast by V_i. Thus (if communication is reliable) a vehicle should receive all safety messages relevant to it, but not all safety messages received will be relevant. The commitment distribution problem

can be stated as follows: for any pair of vehicles, (V_i, V_j), if V_i is relevant to V_j at time T, then V_j must receive the commitment for the key chain in use by V_i at time T before T+δ.

Two main commitment distribution solutions are proposed to provide efficient ways for vehicles to obtain the commitments of their neighbours in a timely manner. The two solutions are distinguished primarily by the entity responsible for distribution. The first approach called V2X Application Server-centric (VAS-centric) is an application-level solution that makes use of a central server to send commitments to vehicles that are identified to be relevant to each other in a given time interval. The second approach referred to as vehicle-centric, is a distributed scheme in which each vehicle distributes its own commitment.

The performances of these distribution solutions are evaluated using the following criteria:

- **Timeliness of the distribution scheme** – represents the ability of the scheme to deliver the commitment of a safety message sender to the relevant vehicles before it is needed. When a vehicle receives a newly-disclosed key, it checks whether it has the sender's current commitment key. If it does, the vehicle validates the key as being part of the chain currently in use by the sender and subsequently verifies messages received during the period for which the key is valid. Consequently, if a commitment is not received at all, safety messages signed with keys that this commitment is required to verify cannot be used. If the commitment is delivered after a corresponding key, verification and processing of any safety messages signed with this key will, at best, be delayed and their usefulness decreased.
- **Distribution efficiency** – represents the ability of an implementation of the scheme to avoid distributing commitments unnecessarily. It can be measured as the proportion of received commitment messages that are subsequently used to validate keys of relevant safety messages.
- **Storage cost** – represents the amount of buffer space required on vehicles onboard units to store the commitment key messages.
- **Impact of the commitment messages on the delivery of safety messages** – this measures the potential impact on safety message delivery, which are increase in latency and message drop rates, as a result of simultaneous transmissions of both the commitment key messages and the safety messages on the same broadcast channel.

This applies to the distribution solutions that utilise the broadcast channel for delivery of commitment keys.

The above performance criteria will help to assess the viability of each distribution solution and to compare them with each other and with an alternative solution that exists in the literature. Below is the detailed description and analysis of these approaches.

4.2 V2X Application Server-Centric Solution

This solution involves extending the role of the V2X Application Server (VAS) introduced by 3GPP in its C-V2X architecture [101]. In this architecture, the VAS is a central, trusted entity that exists at the application layer and communicates with the vehicles on the user plane. As previously described in chapter 2, the VAS is responsible for exchanging additional road safety and traffic efficiency related information with user equipment (including vehicles, roadside units, and devices of pedestrians) supporting V2X applications. In this solution, the VAS first determines vehicles that are relevant to each other, and then forwards the commitments to each vehicle through unicast transmission via the eNB. The approach applies equally well to ITS that use cellular communications end-to-end and to ones that use a cellular (or fixed) network between the VAS and an edge-of-network gateway and a local wireless protocol between gateway and vehicle.

The VAS maintains a commitment key table, KT_{VAS}, relating vehicle IDs to current commitments such that $KT_{VAS}(V_i) = K_{0i}$. All vehicles are required to a) register with the VAS and b) send it the commitment of their latest key chain (updated whenever a key chain is recomputed). As part of the registration process, a shared key, $SK_{V_i}^{VAS}$ is agreed between each vehicle V_i and the VAS. In addition, the VAS assigns a pseudo-identity ID_{Vi} to each vehicle V_i. This key is used to generate and check HMACs to ensure the integrity and authenticity of messages exchanged between the VAS and the vehicle. In particular, the messages notifying the VAS of a new commitment and informing a vehicle of commitments of relevant vehicles are protected in this way as shown.

The VAS could adopt a reactive, on-demand distribution strategy whereby vehicles request commitment keys as and when they need them. Effectively, it is the vehicles that are

determining relevance in this case. The requesting vehicle would have to wait while the request travels from the network edge to the central VAS and for the commitment key to return before it can be used to verify safety messages being received. To avoid this delay, we instead propose a proactive distribution strategy in which the VAS predicts the needs of vehicles and delivers the commitment keys just in time for use. To enable this, the VAS also maintains a commitment distribution table *CDT*, such that *CDT(V_i, V_j)* is true if and only if V_j has been sent V_i*'s current* commitment, i.e. K_{0i}. The relevant entry in *CDT* is set when whenever the VAS sends out a commitment key, and unset whenever a commitment key is updated or becomes invalid for some reason.

The VAS executes a relevance prediction function, *rel*, such that *rel(V_i, V_j, ΔT)* returns true if and only if it is judged that safety messages from V_i would be useful to V_j for at least part of the interval between the current time and the current time i.e. $T+\Delta T$. The third argument thus determines how far ahead the relevance function is looking. Clearly, the VAS must possess sufficient knowledge of the vehicles' states and other factors in order to evaluate *rel*. If *rel* evaluates to true for a given vehicle pair *(V_i, V_j)* and the corresponding entry in *CDT* is false, then the VAS sends V_j V_i's commitment key and changes *CDT(V_i, V_j)* to true. On the side of the vehicle, each vehicle *i* locally maintains a table KT_i, to store current commitment keys of other vehicles received from the VAS and uses it to check whether the commitment key of a safety message sender is available or not in order to proceed with the message verification. Below is the description of the steps/phases involved in the VAS-centric solution:

System Initialization and Registration Phase

The VAS-centric solution requires some cryptographic parameters to be known by the vehicles for shared secret key generation, which is derived using Elliptic Curve Diffie Hellman (ECDH) as follows:

1. The VAS picks (i) a finite field $\mathbb{Z}_p^*$, where p is a large odd prime of at least 160 bits, $\mathbb{Z}_p^*$ = {0, 1, 2, . . . , p − 1}; (ii) an elliptic curve Ep (a, b) having the set of all points of the elliptic curve: $y^2 = x^3 + ax + b\ (mod\ p)$ such that a, b ∈ $\mathbb{Z}_p^*$ are constants satisfying the condition $4a^3 + 27b^2 \neq 0\ (mod\ p)$, together with a special point *O* called the point at infinity or zero point; and (iii) a base point G in Ep (a, b).
2. VAS select H a one-way hash function H: $(0,1)^* \rightarrow \mathbb{Z}_p^*$;

3. VAS chooses a random number SK_{VAS} as its private key and then compute its corresponding public key as $PK_{VAS} = SK_{VAS}.G$.
4. Then the VAS publishes the credentials $\{p, a, b, G, H, PK_{VAS}\}$ to all vehicles
5. V_i chooses a random secret SK_{Vi} as its private key and calculate its corresponding public key $PK_{Vi} = SK_{Vi}.G$.
6. V_i sends a registration request to the VAS with its unique ID_{Vi} and PK_{Vi}, $ID_{Vi} = H(SK_{Vi}||N)$, where N is a cryptographic nonce value.
7. VAS generates a unique long-term temporal credential TC_{Vi} for V_i. $TC_{Vi} = H(ID_{Vi}||SK_{VAS}||PK_{Vi}||RT)$, where RT is the registration time for V_i.
8. VAS stores <$ID_{Vi}||TC_{Vi}||PK_{Vi}$> in its commitment key table KT_{VAS} , which will be used to identify each vehicle and its corresponding commitment key.
9. The VAS and V_i derives a shared secret, s: V_i computes $s = SK_{Vi}.PK_{VAS}$ and VAS computes $s = SK_{VAS}.PK_{Vi}$
10. Both the VAS and V_i use s as input to a key derivation function (KDF) to generate a symmetric shared key K_{Vi}^{VAS} used for securing the commitment messages exchanged between VAS and V_i: $K_{Vi}^{VAS} = KDF(s)$

Commitment Generation and Transmission Phase

Each vehicle generates and sends commitments of its latest key chain to the VAS a follows:

1. V_i randomly chooses the last value of its key chain K_n and then repeatedly applies H to derive previous values: $K_i = H(K_{i+1})_{\forall i \in \{0,1,\ldots,n-1\}}$. K_{0i} serves as commitment to the entire key chain currently in use by V_i
2. V_i computes $H(K_{0i}||T_{0i}||T_s)$ using K_{Vi}^{VAS} where $T_{0i}$is the key chain expiry time and $T_s$is the current time stamp.
3. V_i sends the commitment message $\{ID_{Vi}, K_{0i}, T_{0i}, T_s, H(K_{0i}||T_{0i}||T_s)\}$ to the VAS

Commitment Add/Update

The VAS adds/updates vehicles' commitments in its KT_{VAS} table.

1. Once the VAS receives a commitment message $\{ID_{Vi}, K_{0i}, T_{0i}, T_s, HMAC(K_{0i}||T_{0i}||T_s)\}$ from V_i
2. It computes $H'(K_{0i}||T_{0i}||T_s)$ using $K_{Vi}^{VAS}$and compare it with $H(K_{0i}||T_{0i}||T_s)$.

3. If step 2 is correct, the VAS add/update K_{0i}, T_{0i} in KT_{VAS} for V_i entry. Otherwise, the VAS discard the commitment message $\{ID_{Vi}, K_{0i}, T_{0i}, T_s, \ H(K_{0i}||T_{0i}||T_s)\}$

Determination of Relevant Vehicles and Distribution of Commitment Messages

The VAS determines vehicles that are relevant to each other and then sends commitment keys to each vehicle.

1. VAS determines vehicles that are relevant to V_i
2. VAS computes $H(K_{0j}||T_{0j}||ID_{Vj}||T_s)$ using K_{Vi}^{VAS},where K_{0j} is the current commitment of a given vehicle V_j that is relevant to V_i, and ID_{Vj} is V_j's unique identity.
3. VAS send the commitment message $\{ID_{Vi}, ID_{Vj}, K_{0j}, T_{0j}, T_s, H(K_{0j}||T_{0j}||ID_{Vj}||T_s)\}$ to V_i.
4. VAS updates its CDT.

Commitment Reception by Vehicles

On the receipt of commitment key message sent by VAS, each vehicle verifies the authenticity of the message using the shared secrete key with VAS as follows:

1. V_i uses K_{Vi}^{VAS} and recomputes $H'(K_{0j}||T_{0j}||ID_{Vj}||T_s)$
2. V_i check if $H'(K_{0j}||T_{0j}||ID_{Vj}||T_s) = H(K_{0j}||T_{0j}||ID_{Vj}||T_s)$
3. If a matching is found in step 2, then V_i add ID_{Vj}, K_{0j}, T_{0j} in its KT_i table. Otherwise V_i discards the commitment message, and waits for the reception of the next commitment key message from VAS

Algorithm 4-1 describes the distribution procedure executed by the VAS. The algorithm executes periodically at intervals of δT^{VAS}, which is reasonable if it is running as a thread within a server that also has other tasks to perform. For each ordered pair of vehicles *(V_i, V_j)* registered with it, the VAS checks that V_j has not already been sent V_i*'s* commitment key and that V_i is, or is expected soon to be, relevant to V_j. If this is true, then the VAS sends V_i's commitment key to V_j and records this fact in *CDT*. Once all ordered pairs have been considered, the VAS waits for the next scheduled time to repeat the process.

Algorithm 4-1: VAS-Centric:

$TP = current\ time$
$while\ (true)$

$for\ each\ pair\ of\ vehicles\ (V_i, V_j)\ do$
$if\ CDT(V_i, V_j) = false\ and\ rel(V_i, V_j, \Delta T) = true$
$then$
$send\ KT_{VAS}(V_i) to\ V_j$
$set\ CDT(V_i, V_j) = true$
$end\ if$
$end\ for$
$TP = TP + \partial T^{VAS}\ wait\ until\ current\ time\ \geq TP$

$end\ while$

Below is the description of the particular relevance prediction function that is used in this work to assess the VAS centric commitment distribution concept.

4.2.1 A Relevance Prediction Function

(4.1) defines a concrete instance of the abstract relevance prediction function in order to illustrate and evaluate the concept. It is relatively simple, depending only on the straight-line distance between the two vehicles and the average vehicle speed in the locality. More elaborate, and potentially more accurate, forms can readily be envisaged that take into account factors such as relative velocities, projected closest approach distances, and road geometries.

$$rel(V_A, V_B, \Delta T) = rel(V_B, V_A, \Delta T) = true, if\ and\ only\ if\ d(V_A, V_B) \leq r[v(V_A\ or\ V_B), \Delta T]$$

$$r(v, \Delta T) = r_0 + r_1(v) + r_2(v, \Delta T) \qquad (4.1)$$

$$r_0 = a_0, \qquad r_1(v) = a_1.v, \qquad r_2(v, \Delta T) = a_2.v.\Delta T$$

where $d(V_A,V_B)$ is the distance between V_A and V_B, and $v(V_A) \approx v(V_B)$ is the average speed of vehicles in the vicinity of V_A and V_B, which is assumed to vary slowly with the position.

One can imagine three circles centred on V_A as illustrated in Fig. 4.2. A circle of radius r_0 encompasses all vehicles judged to be relevant to V_A when traffic is stationary. It seems

reasonable for the 'radius of relevance' to grow with vehicle speed, hence a circle of radius $r_0+r_1(v)$ encompasses all vehicles relevant to V_A at the current time (i.e. without extrapolating vehicle trajectories) when the local traffic speed, v, is finite. The largest circle, of radius $r_0+r_1(v)+r_2(v, \Delta T)$ adds vehicles that might become relevant in the next ΔT.

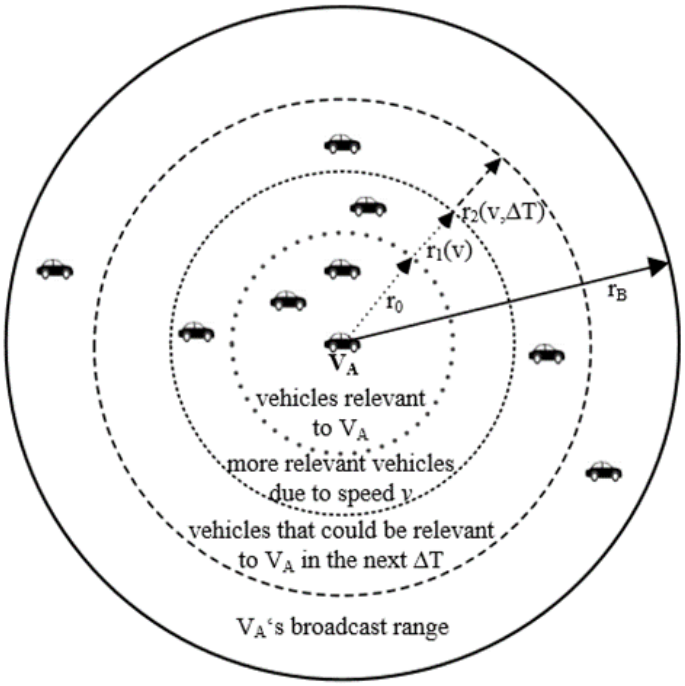


Figure 0.1: Radius of Relevance vs Broadcast Range

The above relevance prediction function is comparable to the studies in [102], as both share a common emphasis on evaluating the significance of vehicles in close proximity and how their messages are critical to safety applications. While the authors in [102] formulates and measure what is referred to as 'vehicle awareness' using heading information and selectively considers critical neighbours, the function $rel(V_A, V_B, \Delta T)$ determines the significance of vehicles based on straight-line distance and average vehicle speed in the locality. Both approaches underscore the importance of spatial relationships between vehicles and prioritize safety concerns, with [102] using a weight function for position error prioritisation and our work estimating relevance through considerations of proximity and local speed.

4.2.2 Commitment Key Distribution Frequency

Suppose we initialise *CDT* with all elements set to false and run algorithm 4-1. Vehicle V_A will receive an initial burst of messages informing it of the commitment keys of vehicles within its

radius of relevance. If $\rho(V_A)$ is the density of vehicles in the vicinity of V_A, then this burst will consist of $\rho(V_A).\pi r^2(V_A, \Delta T)$ messages. Subsequently, V_A will receive further messages as additional vehicles enter its radius of relevance. The rate will be proportional to the circumference of V_A's circle of relevance and the local vehicle speed. Thus, the total number of messages sent to V_A up to time T will be:

$$M^{VAS}(V_A, T, \Delta T) = \rho(V_A).\pi r^2(V_A, \Delta T) + \rho(V_A).2\pi r(V_A, \Delta T).v(V_A).T \quad (4.2)$$

as long as the number of commitments known by V_A is much less than the total number of vehicles in range of V_A. With this algorithm, the rate at which messages are sent to V_A will decline with time as it becomes more likely that vehicles encountered by V_A are already known to it. However, in a real implementation, V_A's capacity for storing commitments will be constrained. In consequence, a mechanism for 'discarding' commitments that have not been accessed recently will be needed, with the VAS being informed accordingly so it can update its *CDT.*

The radius of relevance $r(V_A, \Delta T)$ increases with ΔT, so does the $M^{VAS}(V_A, T, \Delta T)$ also increases. It is advantageous for a proposed mechanism or network protocol to ensure that it does not introduce high traffic load or signalling overhead in order to minimize congestion on the limited transmission resources. Thus, to minimize the number of messages sent/received, the value of ΔT is being kept as small as possible. A lower bound on ΔT is dictated by the time it takes the VAS to complete a loop of Algorithm 4-1. ΔT must therefore equal or exceed δT^{VAS}. We choose $\Delta T = \delta T^{VAS}$ and make δT^{VAS} as small as possible. The minimum value for δT^{VAS} will depend on the processing capacity of the VAS, the number of vehicles registered to it, and the time taken for a commitment key sent by the VAS to reach and be verified by the destination vehicle.

4.3 Vehicle-centric Solution

In this solution, vehicles are responsible for sending the commitment of their latest key chain to their neighbours using the same direct communication channel employed to broadcast safety messages. A VAS is still required, but its role is greatly reduced. Similar to the VAS-centric solution, vehicles are required to register with, and forward their current commitment keys to the VAS. However, in this case, the VAS only signs the commitment key and returns it

to the vehicle. This process can be done in advance, for example while the vehicles are parked, and then use the signed commitment later. Contrary to the VAS-centric case, an asymmetric digital signature mechanism is used to protect the integrity of the commitment message. Each vehicle holds the public key of the VAS, which can be pre-provisioned or obtained during the registration, and this is used to verify the commitment keys sent by its neighbours.

As in the VAS-centric case, it is possible to adopt a reactive, on-demand approach whereby on receipt of a safety message from an unfamiliar vehicle, a vehicle requests its commitment key. In this case, the request is made to the relevant vehicle (rather than the VAS), which provides its commitment key in response. This means that the reaction to the safety message is delayed by at least the request-response round trip time. As mentioned above, we assume that a vehicle is able to receive broadcast messages from all vehicles that are relevant to it. Consequently, the unfamiliar vehicle is detected and its commitment key is requested before it is relevant. The latency of the request-response exchange may therefore be acceptable provided the commitment key is received before the zone of relevance is entered and communication is reliable. Another disadvantage is that a malicious agent may exploit the on-demand reactive approach by sending multiple requests forcing vehicles to respond with their commitment keys. This could potentially cause congestion on the transmission channel, preventing vehicles from sending/receiving safety messages. In addition, since V2V communication is unreliable and vehicles may only be within range of each other for a short period especially in high traffic speed situations, the request/response message may be lost, and the received message from the unfamiliar vehicle would eventually have to be dropped.

Because of these disadvantages, a proactive approach is adopted in this work. Here, all vehicles broadcast their commitment keys repeatedly at intervals of δT^{veh}. In this case, commitment keys are received at approximately the same time as the first safety messages from a new neighbour. This makes the acquisition of commitment keys faster, but at additional overhead costs, which will be discussed later. Algorithm 4-2 describes the simple broadcast distribution solution executed by each vehicle in sharing its commitment key.

System Initialisation, Registration and Commitment Key Transmission Phase

In the vehicle-centric solution, vehicles also need to register with the VAS and they need the VAS's public key. Thus, Step 1 to 8 are the same with the VAS-centric case. Then each vehicle

V_i sends its commitment (K_0) to the VAS for signing. Note that this phase can be done offline or in advance, e.g. when the vehicles are parked. V_i can also send a set of commitments (K_{01}, K_{02}…K_{0n}) for its different key chains for the VAS to sign, and then use them later.

Commitment Signing

The VAS generates an ECDSA signature of the commitment or set of commitments and then sends this back to the individual vehicle through the secure channel as follows:

a. Randomly select $t \in [1, n-1]$, where n is the order of the elliptic curve group generated by the point G.
b. Compute the random point $t * G = (a_1, b_1)$ and $r = a_1 mod\ n$, if r=0 then go to step a
c. Calculate the hash value $SHA(K_0||T_0)$
d. Compute the signature proof $s = (t^{-1}SHA(K_0||T_0) + k * r)\ mod\ n$. If s = 0, then go to step a
e. VAS sends $\{K_0||T_0||(s,r)||T_s\}$ to V_i

Commitment Distribution and Verification

Each vehicle broadcasts its commitment message $\{K_0||T_0||(s,r)||T_s\}$ repeatedly at intervals of δT^{veh}. A receiver verifies the signature (s,r) in the commitment message using VAS public key as follows:

a. Compute $w = s^{-1}(mod\ n)$, where w is the modular multiplicative inverse of s
b. Calculate $u_1 = (SHA(K_0||T_0) * w)\ mod\ n$
c. Calculate $u_2 = (r * w)\ mod\ n$
d. Recover the random point $R' = u_1 * G + u_2 * PK_{VAS}$
e. Compute $r' = R' mod\ n$.
f. If $r' == r$, then $K_0||T_0$ are verified and stored in KT table; otherwise, they are discarded.

Algorithm 4-2: Vehicle-Centric:

Pre-conditions: vehicle V_i is registered with VAS, and has received the signed commitment of its latest key chain from the VAS K_{0i}.

$TP = current\ time$

$while\ (true)$

$i = get(K_{0i})$

$broadcast(i)$

$TP = TP + \partial T^{veh}$

$wait\ until\ current\ time\ \geq TP$

$end\ while$

As commitment keys are sent repeatedly at intervals of δT^{veh}, it is likely that vehicles will receive multiple copies of the same commitment key from the same sender. To address this issue and prevent the waste of a vehicle's processing power and time in verifying every received commitment message, each receiver checks its key table to determine whether it already has a verified copy of the sender's current commitment key. If it does not, and the newly received commitment key is verified successfully, the receiver adds it to its commitment key table. This approach also reduces the risk of potential denial-of-service attacks.

4.3.1 Commitment Key Distribution Frequency

The broadcast of commitment keys every δT^{veh} increases the chances of vehicles that are within broadcast range to have a copy of each other's commitment keys. However, in this solution, the rate at which messages are sent and received by each vehicle is greater than in the VAS-centric case. The rate of messages received by vehicle V_A per unit time can be expressed as:

$$\dot{M}^{veh}(V_A) = {(\rho(V_A).\pi r_B^2)}/{\delta T^{veh}} \quad (4.3)$$

which can be compared with (from(4.2))

$$\dot{M}^{VAS}(V_A, T, \Delta T) = \rho(V_A).2r(V_A, \Delta T).v(V_A) \quad (4.4)$$

The vehicle-centric messages include duplicate commitment keys that are received from the same vehicle while it is still in the broadcast range unless the vehicle generates a new key chain. The average number of these duplicate copies can be determined by estimating the average time spent in the broadcast range of the sending vehicle. Assuming a uniform distribution of vehicles within the broadcast range, and that vehicles are moving with an average speed of v, the average distance is given by:

$$L_{avg} = 2 * \frac{1}{r_B} \int_0^{r_B} \sqrt{(r_B^2 - x^2)}\ dx = \frac{\pi r_B}{2}$$

and the average time spent in the broadcast range is given as:

$$T_{avg} = {}^{\pi r_B}/_{2v}$$

The average number of duplicate copies of commitment key messages D_{ck} can then be expressed as:

$$D_{ck} = {}^{T_{avg}}/_{\delta T^{veh}} \qquad (4.5)$$

The choice of δT^{veh} has a direct effect on the number of messages received and resource consumption. A small value of δT^{veh} makes it more likely that commitment keys are received before they are needed but at the cost of an increased number of messages that may overwhelm the processing and buffer capacity of the vehicle's OBU. On the other hand, using a large δT^{veh} interval leads to fewer messages, but some vehicles might not receive some commitment keys in time, especially when vehicles are moving at high speed. Since the main purpose of the commitment key is to validate the key to be used by the receiver to authenticate safety messages, there is no advantage, therefore, in broadcasting the commitment key more frequently than the safety messages themselves. In fact, one simple strategy is to send them at approximately the same time. Furthermore, since the same key is used to authenticate all messages sent by a vehicle during a given TESLA time interval before the related messages are sent, there is no advantage in broadcasting the commitment key more than once per TESLA time interval. These considerations give a lower bound on δT^{veh} of the greater of the safety message broadcast interval and the TESLA time interval.

To minimize the number of commitment key message transmissions as well as the communications overhead, an adaptive mechanism can be introduced, such that vehicles

adjust their δT^{veh} according to the perceived traffic situation. At low speed, a vehicle will have a consistent set of neighbours for a long period of time, and most likely, repeat copies of commitment keys will be received. In this case, the vehicles should use large intervals to save resources and prevent potential buffer overflow. Conversely, in a high-speed environment, vehicles should apply small intervals to distribute commitment keys to rapidly changing neighbours. What matters is that at least one copy of the commitment key is received by the time the first relevant safety message needs to be authenticated. The worst-case involves two vehicles approaching each other head-on. The time available between the first possible receipt of a commitment key and the first possible receipt of a relevant safety message is $(r_B - r)/2v$, which gives an upper bound on δT^{veh}.

4.4 Literature Review of Commitment key Distribution in V2V

This section presents existing research studies found in the literature that also proposed different solutions to the commitment key distribution problem in V2V. For instance, in [25], the authors proposed a reactive approach. Here, when vehicle V_A receives a safety message from V_B whose commitment key it does not have, it sends V_B a commitment key request. V_A also sends a request to a nearby roadside unit (RSU) to obtain a special token computed using a Bloom filter (BF) that allows it to verify the commitment key received from V_B. This is described in more detail later where it is compared with VAS-centric and vehicle-centric schemes. Similarly, the work in [68] also proposed a reactive commitment key exchange method. When a vehicle V_A receives a message from an unknown vehicle V_B, V_A broadcasts its commitment key together with the list of vehicles for which V_A needs commitment keys. As V_B receives this broadcast message and finds its own identifier in V_A's list of identifiers for which a commitment key is requested, V_B also broadcasts its commitment key. In this way, other vehicles within the vicinity of V_A and V_B will obtain both V_A's and V_B's commitment keys. Such a vehicle will only need to broadcast its commitment key without requesting the commitment key of its neighbours. However, this approach may result in vehicles being overwhelmed with many copies of messages containing commitment keys that they already possess. Also, it is computationally expensive because the broadcast messages are verified using asymmetric cryptography. Moreover, the scheme is vulnerable to denial of service attack, as a malicious vehicle can simply use different identities to trigger other vehicles to

broadcast their commitment keys. The authors of [69] also adopted periodic broadcasts of commitment keys at a fixed time interval in their TESLA-based cooperative message verification scheme. However, they did not evaluate its performance nor did they provide the mean time interval that should be used for the periodic distribution.

Some of the above commitment key distribution methods can affect the operation of the underlying safety applications as some safety messages have to be either delayed or discarded due to a lack of the required commitment key. For instance, the approaches of both [25] and [68] share the same characteristics of requesting a commitment key and waiting until it is received. As no acknowledgment (ACK) / negative acknowledgment (NACK) mechanism is used in the V2V broadcast setting and so no retransmission of lost messages, when any of the request/response messages are lost, the buffered safety message cannot be verified and would have to be discarded after its elapsed time. A mechanism to determine when to discard buffered safety messages due to the non-availability of commitment key needs to be provided. In addition, the approaches of both [25] and [68] have some vulnerabilities that can be exploited by adversaries to launch further attacks. For example, one or more malicious agents could send multiple request messages forcing vehicles to respond with their commitment keys. This could cause congestion on the transmission channel preventing the transmission of conventional safety messages or leading to a potential denial of service attack. Moreover, [68] requires collaboration among the vehicles to maintain a consistent neighbour list. This cannot be easily achievable in a dynamic V2V environment.

4.5 Analytical Evaluation of Commitment Key Distribution Solutions

Using the performance criteria outlined in 4.1, this section compares and evaluates analytically the performances of VAS-centric and vehicle-centric commitment distribution solutions with each other and with the solution proposed in Bao et al [25]. Of the works described in the previous section, the work in Bao et al was chosen for comparison because it is basically a reactive version of our vehicle-centric solution, but with support from a roadside infrastructure. Moreover, to the best of our knowledge, it is the most recently published work in the literature that addresses the commitment key distribution problem in V2V at the time of writing this thesis. In addition, the authors present the analytical results of

their proposed solution, while other works describe their ideas without thorough analysis and evaluation.

4.5.1 Timeliness of the Distribution Scheme

To define this metric, we need to establish the criteria for determining the relevance of safety message to a given receiver and the latest time a commitment can be delivered if it is to be used to validate the key needed to verify a safety message. For relevance, we consider that the direction of movement, relative traffic speed, and separation distance between vehicles as factors to be used in defining relevant safety messages out of all received safety messages. The specific definition of relevance used for the evaluation will be discussed later. Concerning the latest time a commitment needs to be delivered, let us suppose that a safety message has associated with it an expiry time, i.e. a time beyond which the information contained in the message is no longer useful. For the message to be used, it must have been verified using the key used to sign it, and that key must have been validated using the appropriate commitment. Thus, for a commitment to be timely with respect to a relevant safety message received by a given vehicle, the commitment must be delivered to that vehicle, and the difference between the arrival time of the commitment and the expiry time of the message must be greater than the sum of the commitment verification, message verification and key validation times. The timeliness of a commitment is then the number of relevant messages for which it is timely divided by the number of relevant messages with which it is associated[1], and the timeliness of the distribution scheme is the total number of relevant messages for which a timely commitment exists divided by the total number of relevant messages. It is given as:

$$\beta = \frac{M^T}{M^R} = 1 - \frac{M^U}{M^R} \tag{4.6}$$

where M^R is the total number of relevant messages, M^T is the total number of relevant messages for which a timely commitment exists, and M^U is the total number of relevant messages for which the associated commitment is untimely.

The reasons for untimely commitment are either (a) the commitment was not sent at all or sent too late (simply referred to as unsent), (b) the commitment was dropped in transmission

[1] A commitment is said to be associated with a message if the key required to verify the message is in the commitment's key chain. A commitment can only be timely with respect to messages it is associated with.

due to the unreliability of the commitment delivery channel, or (c) delivery of the commitment takes too long, which is a function of the transmission and commitment verification delays. We can thus write:

$$M^U = M^{Un} + M^{Ud} + M^{Ul} \quad (4.7)$$

where the second index indicates the reason for untimeliness (a), (b) and (c).

The unsent (M^{Un}), reliability (M^{Ud}), and late delivery (M^{Ul}) components of the three distribution solutions are discussed below.

VAS-centric Approach

For the VAS-centric approach, which is predictive, the 'unsent' component of the timeliness depends on how accurate and successful the relevance function used by VAS is in anticipating the need for the relevant vehicles. For instance, If $rel(V_A, V_B, \Delta T)$ mistakenly thinks that V_A and V_B are not relevant to each other or would not be relevant in the next ΔT, then the VAS will not send out their commitments. The accuracy of the relevance function can better be assessed and evaluated with an implementation in a realistic V2V setting and will be investigated in the simulation section. The late delivery component of the timeliness depends on the time it takes for the VAS to loop over all registered vehicles and the transmission and processing times for an individual commitment message. The commitment message processing delay is negligible due to the use of symmetric cryptography in protecting the commitment messages. Similarly, the transmission time is negligible due to the use of a downlink channel for commitment delivery, which has a high message processing capacity. However, there is a potential latency issue in that if the VAS is serving many vehicles, the time taken to loop over all vehicles could result in some commitments being delivered late. But, since the VAS is aiming always to deliver a commitment just before it is first needed, the delivery time including the loop time is taken into account in the ΔT parameter and also the δT^{VAS}. Moreover, the VAS is assumed to have computational resources, such that each cycle will be completed on time and deliver all commitments before the next one. This potential latency issue is not a problem, and thus the late delivery component is negligible. The reliability component depends on how reliable the unicast channel between the VAS and the vehicles is. In general, communications reliability is often analysed and modelled in the literature in terms of factors that could cause packet losses. These include errors due to

propagation effects, errors due to a received power below the sensing power threshold, errors due to packet collisions, and errors due to half-duplex transmissions. The probability of a packet being lost is modelled following the approach of [103] [104], which is given as:

$$\alpha = 1 - (1 - BER)^L \quad (4.8)$$

where BER is the bit error rate probability and L is the packet length. Using BER = 10^{-4} as obtained from [103] [104], and a commitment message of size 384 bits for the VAS-centric case, the value of α is estimated to be 0.037674.

To get an estimate for M^R, let us suppose that the radius of relevance is set to 50m, such that all messages received by a given receiver from other vehicles within this radius are considered to be relevant to the receiver. This is similar to the concept used in [105], where the authors used a zone of relevance to limit the number of vehicles that would need to execute their proposed misbehaviour detection algorithm. Combining this together with the results obtained in chapter 3 under our ring scenario for r = 50m, a vehicle receives 840 messages in 100s, which is approximately 9 relevant messages per second. $M^{Ud} = M^R \times \alpha$, and M^R = 840, thus, the percentage of relevant messages that cannot be used due to lack of a timely commitment as a result of commitment message loss is estimated to be ≈ 4%.

Vehicle-centric Approach

In the vehicle-centric case, every vehicle broadcasts its commitment repeatedly at intervals of δT^{veh}, so the 'unsent' component is zero i.e. M^{Un} = 0. There is no advantage to a malicious vehicle in not sending out its commitment or delaying its transmission, as this will only result in its messages not being verified and eventually discarded by the receivers.

The late delivery component of timeliness depends on the transmission and commitment message processing delays. Using M/M/1 queuing theory to model the transmission and receiver queue systems, the average time T^{veh} a commitment message spent in the transmission and receiver queue systems (including the time spent being processed) is expressed similar to equation 3.2 in chapter 3. The communication range r_B that each vehicle broadcast its commitment is likely to be much larger than the radius of relevance. Thus, it is highly likely that a commitment associated with a relevant safety message would be received and processed before the message itself. In addition, the duplicate transmission of a

particular commitment means that a given vehicle may already have a copy of the sender's commitment in its table before it becomes relevant. The 'worst case' is at the start of a new chain. Even here, if we assume that commitments are sent with safety messages, the new commitment would be sent with the safety message signed with the first key of the new chain, and the key used to sign the safety message would be sent δ later. So there is a good chance that the commitment will be processed before the key arrives. Even if verification of the safety message is delayed a bit, the commitment would be timely as long as verification can occur before the safety message expires. Therefore, T^{veh} will not have an impact on the timeliness of a commitment associated with a relevant safety message. The percentage of relevant safety messages that cannot be used due to lack of a timely commitment because of latency issues is thus zero i.e. $M^{Ul} = 0$.

The reliability component of timeliness depends on how reliable the direct broadcast channel is between vehicles. However, the repeated transmission of a particular commitment every δT^{veh} increases the chances of a copy of it being successfully received. The average number of duplicate copies D_{ck} of a given commitment is estimated to be 628 (from 4.5), based on the average time vehicles spent within each other's communication range assuming an average traffic speed of 7.5m/s taken from the simulation runs in chapter 3. Using $M^{Ud} = M^R \times \alpha^{D_{ck}}$ with α equals 0.07096 in the vehicle-centric case, the percentage of relevant messages that cannot be used due to lack of a timely commitment because of commitment message loss issue is ≈ 0.

Bao et al Approach

For the Bao et al approach, which requests commitments when the need arises, the 'unsent' component of the timeliness depends on the reliable delivery of the message requesting the commitment. However, if this message is lost, then the safety message sender will not send its commitment as it is not aware that its commitment is needed. In this case, the receiver can take two possible actions: (a) to send another request message when it receives another safety message from the same sender for which it does not already have its commitment, or (b) to discards all messages from the sender for which it does not receive its commitment after sending the first request message. For case (a), a commitment will probably be received because of repeated requests. As the authors did not mention what action the receiver will

take in the event that no commitment was received after sending the first request message, we assumed case (a). In this case, the 'unsent' component of the timeliness is negligible, because vehicles start requesting for commitment upon receiving the first safety message from an unfamiliar sender i.e. before they become relevant to each other, and the communication range is much larger and radius of relevance.

In the Bao et al approach, a commitment could be untimely if its delivery takes too long due to the roundtrip communications delays. In addition, a received commitment may become untimely if the corresponding BF value needed to verify it was not delivered at all or was delivered late. This effectively means that the late delivery component of the timeliness has two sub-components: (a) the reliability affecting the delivery of both the request and response BF messages and (b) the two roundtrip communications delays plus the commitment and BF processing times. For the reliability issue, we assumed that the BF request message to a nearby RSU is sent only once. Thus, the number of relevant safety messages that cannot be used due to loss of BF request or response message can be expressed as : $M^R(1 - (1 - \alpha)^2)$, where α is the probability of loss on the unicast channel between the vehicle and the nearby RSU. For the latency issue, the sum of roundtrip communications delay can be modelled using queuing theory, which is expressed as $T^{Bao} = 2T_T + T_Q$, where T_T is the transmission delay, and T_Q is the BF processing delay. However, for the same reason that a commitment is requested at a far distance (i.e. communication range is much larger than the radius of relevance), it is highly likely that it will be delivered and processed before the vehicles become relevant to each other. Thus, T^{Bao} will not cause a commitment to be untimely. Consequently, the reliability affecting the delivery of BF is the main factor that can cause a commitment to be untimely. Using the above expression, together with an estimated value of α=0.09443 for Boa et al case, the percentage of relevant messages affected due to the late delivery component is ≈ 18%.

The reliability component of the timeliness in Bao et al approach depends on how reliable the unicast channel is between the vehicles. The percentage of relevant safety messages that cannot be used due to loss of commitment message is estimated to be ≈ 10%.

In summary, the timeliness of the VAS-centric approach relies on the accuracy of the relevance function. The unreliability of the delivery channel also contributes to the timeliness

of the VAS-centric approach as some proportion of commitment messages were estimated to be lost in transmission. However, in practice, this would not be an issue because of the use of eNB downlink channel for commitment delivery. The downlink channel uses high transmission power and advanced modulation and coding schemes (MSC) at the physical layer, which increases the channel's reliability compared to the direct PC5 and IEEE802.11p channels. Thus, the VAS-centric can deliver timely commitments provided its relevance function is accurate. The vehicle-centric approach shows high timeliness compared to VAS-centric and Bao et al approaches due to its proactive nature and message redundancy. The analysis shows that all the three components affecting timeliness have a negligible impact in the vehicle-centric case. In comparison, the Bao et al approach appears to have the worse timeliness. It shows a high number of relevant messages with untimely commitments compared to the other two approaches. Given that Bao et al approach is reactive and requires four consecutive messages, the major issue is the unreliability of the wireless channels causing untimely commitments.

4.5.2 Distribution efficiency

This metric can be defined as the total number of distinct commitment keys that were available to be used to validate keys of relevant messages (C^U) divided by the total number of commitment messages sent (C^T):

$$\omega = \frac{C^U}{C^T} = 1 - \frac{C^N}{C^T} \tag{4.9}$$

where C^N is the total number of commitments that were not used to validate keys of relevant safety messages. Thus, the factors that contribute to C^N include (a) reception of duplicate commitments, (b) reception of commitments that are not associated with relevant messages, and (c) reception of commitments that are not timely for any relevant messages. We can thus write:

$$C^U = C^T - C^{Nd} - C^{Na} - C^{Nu} \tag{4.10}$$

where C^{Nd}, C^{Na} C^{Nu}, corresponds to duplicate, not associated, and untimely commitments. These components of efficiency are discussed for the three distribution solutions.

VAS-centric approach

The VAS-centric approach is designed to have high distribution efficiency. The use of the CDT table, where the VAS keeps an updated record of commitments that were sent prevents the VAS from distributing duplicate copies. Thus, vehicles will not receive duplicate commitments, so C^{Nd} is negligible. The VAS-centric approach specifically takes relevance into account when predicting which commitments to distribute. As long as the relevance function is accurate, the commitments distributed by the VAS are for safety messages that are or will eventually be relevant to the vehicles. Thus, the component that causes inefficiency due to reception of commitments that are not associated with relevant messages is negligible . The VAS-centric approach tries to deliver commitments just in time before they are needed, as previously discussed. Hence, the commitments sent by VAS will be timely, and so the contribution of inefficiency that is due to the reception of untimely commitment for any relevant message is negligible.

Vehicle-centric approach

The transmission of a particular commitment every δT^{veh} interval results in vehicles receiving duplicate copies of it. Given that radius of relevance = 50m, δT^{veh} =100ms, and average speed = 7.5m/s as considered in the previous section, and using equation 5.5, the number of duplicate copies of a particular commitment that is associated with relevant messages is estimated to be ≈105. Thus, we can say that there is a 105% increase in duplicate reception of the commitment, i.e. C^{Nd} = 105%. In the vehicle-centric approach, vehicles broadcast their commitment over the communication range r_B , and as previously discussed, $r_B$is much larger than the relevance radius. This implies that vehicles will likely receive a high proportion of commitments that are not associated with messages relevant to them. To get an estimate of this, let us consider the results obtained in chapter 4 under both the distributed and ring scenarios. In that, a receiver receives approximately 178 messages/seconds. Assuming that both the safety messages and commitment messages have the same broadcast interval, we can say that the vehicle receives 178 commitments/second, of which 9 commitments/second are from senders within the radius of relevance. This implies that 1878% more commitments were received from vehicles that are not relevant to the receiver, and thus not associated with relevant messages, i.e. C^{Na} = 1878%. In vehicle-centric approach, commitments that are

associated with relevant messages are timely, as described under the timeliness metric. Thus, the component of efficiency due to the reception of an untimely commitment that is associated with any relevant message is negligible.

Bao et al approach

In this approach, vehicles acquire commitments upon request, so the duplicate component of efficiency is zero i.e. C^{Nd}= 0. Also, as request message is sent when the receiver overhears the first message from the unfamiliar sender, it is likely that the vehicles are yet to be relevant to each other. In this case, some commitments that are not associated with relevant messages will be received but will be less than those received in the vehicle-centric case. This can be estimated as the rate of safety messages received outside the radius of relevance multiplied by the probability of loss (due to the reactive nature of Bao et al approach). Using the same settings described above, it is estimated that about 183% more commitments will be received that are not associated with relevant messages C^{Na} = 183%. As described under the timeliness metric, in Bao et al approach, a received commitment can become untimely if the BF value needed to verify it was not delivered at all. Therefore, the component of efficiency due to the reception of an untimely commitment for any relevant safety message is the same as the late delivery component of the timelines metric. This implies that 18% of commitments received will be untimely, i.e. $C^{Nu} = 18\%$. .

In summary, the VAS-centric approach prioritises efficiency by limiting the distribution of commitments to vehicles that are identified to be in need only. Provided the relevance function is accurate, the approach would have a very high efficiency compared to the other two approaches. In comparison, the vehicle-centric approach prioritises timeliness and reliability over efficiency. For the parameter settings considered for evaluation, the approach shows that vehicles will receive a very high proportion of commitments that are unnecessary and not needed. This makes the distribution solution highly inefficient. For Bao et al approach, no duplicate copies are sent and the proportion of commitments not associated with relevant messages is far less than that in the vehicle-centric case, due to its reactive nature. However, the sequence of communications required in obtaining BF makes the efficiency of the Bao et al approach to be dependent on the reliability of the delivery channel. This makes the

distribution of Bao et al approach less efficient than VAS-centric but more efficient than vehicle-centric.

4.5.3 Storage Cost

This metric estimates the amount of buffer space required on the vehicle's onboard unit to store commitments sent by the distribution schemes. To estimate the storage space required, let's suppose that each commitment has associated with it a lifetime, i.e. the time at which the commitment is removed from the buffer when it expires. Thus, the storage space needed at any given time can be expressed as:

$$S_C = \lambda \times C_t \qquad (4.11)$$

where λ is the rate of arrival of new commitments and C_t is the average lifetime of the commitment.

For the VAS-centric approach that sent commitments only when a given pair of vehicles are identified to be in need of it, it is likely that vehicles will store few commitments, i.e. only those associated with messages relevant to the vehicles. Taken C_t =200s, commitment message size = 384 bits, and the arrival rate of commitment in the VAS-centric case λ^{VAS} to be 9 commitments/second, assuming each relevant message has an associated commitment as estimated above, S_C is calculated to be 86.4KB

For the vehicle-centric approach, in which vehicles broadcast their commitment repeatedly to all other vehicles in range, a given vehicle will receive a large number of commitments including duplicate copies and those that are not relevant to it. Given that λ^{veh} equals 178 commitments/second as estimated in 5.5.2, the amount of storage space required S_C is calculated to be 1709KB.

For Bao et al approach that is reactive, vehicles will store commitments that were successfully delivered and verified using the BF value. So lack of delivery of the corresponding BF value will reduce the number of commitment verified, and hence the storage space. Assuming a worse case situation that for every safety message received, a commitment, as well as the corresponding BF, are requested. In this case, the commitment arrival rate will be λ^{Bao} =178 messages/second x α, which gives 17 commitments/second, and the amount of storage space required S_C is calculated to be 166KB.

For the parameter values estimated above, it can be seen that the VAS-centric approach has low storage requirements on the vehicles compared to the other two distribution solutions. The VAS avoids sending redundant commitments and commitments that are not relevant to the vehicles. The vehicle-centric appears to have the highest buffer requirement on the vehicles OBU because it sends commitments proactively to all vehicles in range. Vehicles need to store commitments of other vehicles that are not relevant, which could potentially be discarded without being used. The Bao et al solution has a lower low storage requirement than vehicle-centric, but higher than the VAS-centric case. This is because vehicles also request commitments that are not relevant to them, but the arrival late is less than that of vehicle-centric because of the unreliability of the delivery channel. The commitments of those vehicles in which the corresponding BF value is lost in transmission cannot be verified and consequently stored.

4.5.4 Impact of commitment messages on the delivery of safety messages

This metric investigates the impact of commitment messages on the delivery of safety messages. For the VAS-centric approach that distributes commitments to vehicles via the downlink channel, the delivery of safety messages is not affected. In addition, commitment messages are signed with a symmetric key, and the rate at which vehicles receive commitment messages is low because of the efficiency of the distribution solution, i.e. avoids sending commitments unnecessarily. Thus, a given vehicle will have less commitment in its verification queue. Consequently, the increase in safety message latency due to verification of commitment messages is negligible.

In the vehicle-centric approach, commitments are sent using the same direct communication channel employed to broadcast safety messages. As such, a lot more messages are transmitted on the broadcast channel. These additional messages are protected using an asymmetric digital signature mechanism, i.e. they are larger in size and time-consuming to process. This makes them analogous to the case of signing safety messages using the ECDSA approach. Based on the results obtained for the ECDSA approach from the analytical works in chapter 3 (3.1.3), it was shown that ECDSA introduces high latency overhead to the safety messages at moderate to high traffic density. It indicates that the receiver queue system appears to be the bottleneck, approaching the position of singularity at a traffic density greater than 75 vehicles within range of a given receiver for the estimated values. Beyond this

point, the receiver queue grows up and the safety message delay increases rapidly. Therefore, when the commitment messages are signed with ECDSA and sent periodically on the broadcast channel, the latency that the safety messages will experience will certainly be worse than the latency of ECDSA safety messages due to an increase in message arrival rate. Some safety messages will get dropped when their lifetime elapsed due to a longer waiting time in the verification queue. This increase in safety message latency and message drop rate will be assessed and evaluated in the simulation section.

In Bao et al approach, the request/response messages are sent via the direct unicast channel between the vehicles and between the vehicles and the nearby RSU. However, the unicast channel shares the same frequency bandwidth as the direct broadcast channel. Thus, the request/response messages will still affect the delivery of the safety messages but will be less than the vehicle-centric case, due to the reactive nature of the Bao et al approach, and no redundant transmissions.

4.6 Simulation of Commitment Key Distribution Solutions

This section presents an evaluation of the performance of VAS-centric, vehicle-centric, and the Bao et al [25] commitment distribution schemes using a simulation representative of a real V2V environment.

The simulation is conducted following the procedure and settings used in Chapter 3 and described in 3.2.1. The scenario considered in this work is a city centre road network, the same used in Chapter 3 simulation, as illustrated in Fig. 3.10. Also, the security parameters including ECDSA signature and MAC processing times used are the same as those in Table *0-2*: Simulation Settings.

A simulated VAS implements algorithm 4-1, using the relevance prediction function (4.1) with parameter values listed in Table 4.1 below. Each time step, the average speed of traffic (v in (4.1)) in the vicinity of each vehicle is obtained from SUMO data. An arraylist and a HashMap with key-value pair were used to store vehicles records including vehicles IDs, updated position information, and current commitment keys. The VAS is running on a Linux virtual machine with the following specifications: processor Intel Core i7-8750H @2.20GHz-4.10GHz,

RAM 16GB. It is added as a module in the NS3 simulation tool to communicate with the vehicles via the LTE unicast module, which uses the uplink/downlink channels.

For the vehicle-centric case, vehicles broadcast both commitment and safety messages at regular intervals on the direct broadcast channel. Each safety message has associated with it a lifetime, which is set the moment it is generated at the sender. It is assumed that each vehicle already has a signed copy of its commitment key sent by VAS and that all vehicles have VAS's public key, PK_{VAS} to verify received commitment messages.

In the case of Bao et al, RSUs are deployed to cover the whole simulated area, with each RSU having a communication range of 600m. Each vehicle registers with an RSU within range and sends its latest commitments following the procedure described in [25]. Each RSU aggregates all commitments and generates the corresponding Bloom filter value. We adopt the same Bloom filter settings as used in [25] with k=7 i.e., seven cryptographic hash functions. Unicast mode is used for communication between RSUs and vehicles. It is assumed that each vehicle has the public key of the RSU PK_{RSU} within its range. This is used to verify the signature of BF messages received from the RSU. Also, we set a timer for every commitment request message made, such that when a reply (both commitment and BF reply messages) is not received within the timer window, another request message is sent.

Vehicles also record the arrival time of each commitment message received, and the time at which the commitment is added to each vehicle's KT table after is being verified, which is referred to as T_V. To determine the relevance of a message to a given vehicle, each vehicle computes what is referred to as time proximity (TP) every time step. The TP is defined as the minimum time a vehicle has available to react for safety reasons, e.g., to carry out an emergency stop. It is given as the separation distance between a pair of vehicles divided by the sum of their speeds. If TP is less than the value listed in Table 4.1, then the vehicles are said to be relevant to each other. The value of TP was chosen according to the driver reaction time, deceleration capabilities of the subject vehicle, and some margin to account for possible positioning error, as described in [106]. Table 4.1 lists other parameters used in this simulation.

Table 0-1: Simulation Parameters

Parameters	**Value**
Safety message lifetime	100ms
Commitment key generation interval	200s
a_0	50m
a_1	20s
a_2	10
ΔT	5s
Min. TP	10s
T_M (δ + MAC + Hash operation times)	12.002µs

4.6.1 Simulation Results

This section presents the results obtained from the simulations. The performances of VAS-centric, vehicle-centric and Bao et al distribution solutions are compared and analysed using the equations in section 4.5.

Timeliness of the distribution scheme: To collect information for this metric during each simulation run, each simulated vehicle carries out the steps as follows. On arrival of a safety message, the vehicle first calculates TP to the sender to determine the relevance of the received message. It uses the position and speed information reported in the message and its own information. If the calculated TP is less than min. TP, then the message is considered relevant and is counted as M^R. Otherwise, the message is not used. Next, the vehicle then lookup its *KT* table to check whether a timely commitment for the relevant message exists or not. If yes, then it checks that the difference between the commitment's T_V and the expiry time of the relevant message is greater than T_M (where T_M = δ + message verification + key validation times). If that is satisfied, then the message is counted as M^T. Otherwise, If no commitment is present or it exists but it is untimely, then the message is counted as M^U. The total M^R, $M^{T,}$ and M^U recorded are then aggregated to compute 4.6 for each distribution solution.

Fig. 4.2 shows the curves of timeliness as a function of number of vehicles for the three distribution schemes. It appears that the timeliness of the vehicle-centric solution

outperforms that of the VAS-centric and the Bao et al approaches in the range of traffic densities considered in the simulation. In the vehicle-centric case, vehicles receive timely commitments for all relevant messages received up until when there are around 60 vehicles within range. Beyond this point, the timeliness begins to decrease gradually. The reason for this is that as traffic density increases, more vehicles are in range of each other, therefore, the shared broadcast channel used for transmitting both commitment and safety messages becomes congested. Consequently, it starts to drop packets. This effectively reduces the commitment delivery rate, and thus the timeliness drops down to 0.93 with 100 vehicles within range. In addition, there is the contribution due to congestion at the receiver caused by commitment message verification latency.

In the VAS-centric case, the results indicate that the relevance function used by the VAS was able to predict the needs of the vehicles well and deliver timely commitments. Its timeliness decreases steadily with increasing number of vehicles. We notice that the reason for this decrease is due to VAS's processing capacity to cycle through all the computations. As traffic density increases, the number of vehicles served by VAS increases, and so does the time taken to loop over all vehicles in each cycle. This delays the start of the next cycle until the previous one is complete, and consequently results in some commitment being delivered late. Compared to the analytical work, an expectation for the late delivery component was negligible. Whereas as observed in the simulation result, the late delivery component contributes up to about 10%. The reason for this difference is due to the computational performance of the VAS. Under the analytical estimation, it is assumed that the VAS server is running on a very powerful machine. In the simulation, a machine with the specification described above was used for the VAS. To confirm this, the simulation runs were repeated using a machine with low specification (Intel i5-337U at 1.7GHz, 16GB RAM) than the current one. The result obtained (VAS-centric low spec) indicates worse performance compared with previous as shown in fig. 4.3. Therefore, deploying a processor with a higher processing speed will improve the timeliness performance of the VAS-centric approach.

The Bao et al solution has the worst timeliness compared to the VAS-centric and vehicle-centric approaches. With fewer than five vehicles in the simulated area, all commitments are received and processed on time. Beyond this point, the timeliness drops rapidly, due to the unreliability of the wireless channel affecting the delivery of the commitment and BF

messages, reaching around 0.55 with 100 vehicles in the simulated area. More messages carrying either the commitment or BF value are lost in transmission. Also, the verification time of the BF messages makes commitments to become untimely due to the use of ECDSA.

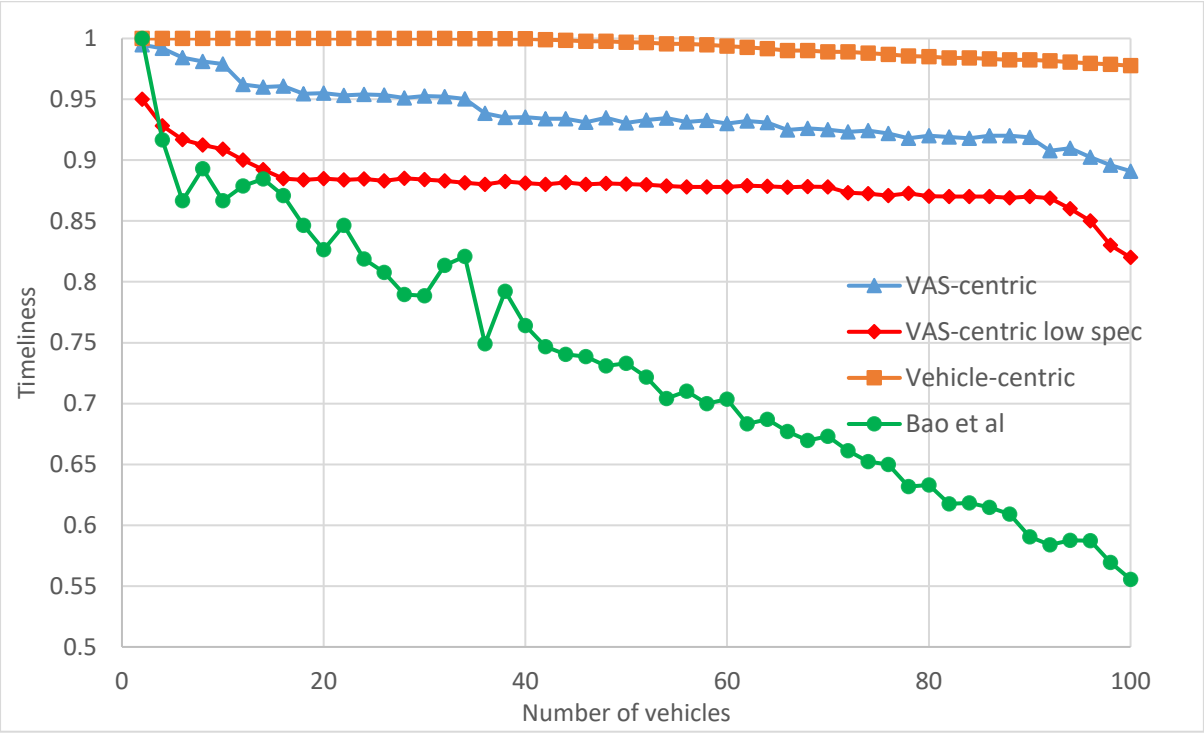


Figure 0.2: Timeliness against number of vehicles

Distribution Efficiency: To collect information for this metric during each simulation run, each simulated vehicle carries out the steps as follows. Upon reception of each commitment message, the vehicle counts it as C^T. Then for every safety message received, the vehicle determines the relevance of the message as described above. Next, the vehicle checks whether a timely commitment for it is available in its KT table. If the commitment is present and timely, then the vehicle marked the commitment as used, and count it as C^U. Other received commitments but never used are counted as C^N. The total C^T, $C^{U,}$ and C^N recorded are then aggregated to compute 4.9 for each distribution solution.

Fig. 4.4 shows the curves of efficiency as a function of number of vehicles in the simulated area for the three distribution schemes. The VAS-centric solution achieves high efficiency, appropriately predicting and sending out commitments that were used by the vehicles. It appears that the vehicles used nearly all the commitments sent by VAS when the traffic density is fewer than 40 vehicles in the simulated area. The efficiency decreases slightly with

increasing traffic density due to late delivery of some commitments, meaning that they were not used by the receiving vehicles. Nevertheless, the efficiency of the VAS-centric is still above 97% with 100 vehicles in the simulated area. The vehicle-centric solution shows lower distribution efficiency. Vehicles used fewer than 10% of the received commitments to validate keys of relevant messages in all the range of traffic densities considered. This means that the vast majority of the commitments received were duplicate copies and commitments not associated with relevant messages. For the Bao et al case, the distribution efficiency is high when the number of vehicles is below 30 because there is less congestion on the transmission channel. The BF messages requested from the RSUs are received successfully, and the verification of the BF value as well as the commitment messages are timely. But as the number of vehicles is above 30, the distribution efficiency starts to decrease due to congestion and unreliability of the channel. The number of BF requests increases, and when the BF request/response messages are loss in transmission or arrived late, the corresponding commitment becomes unused.

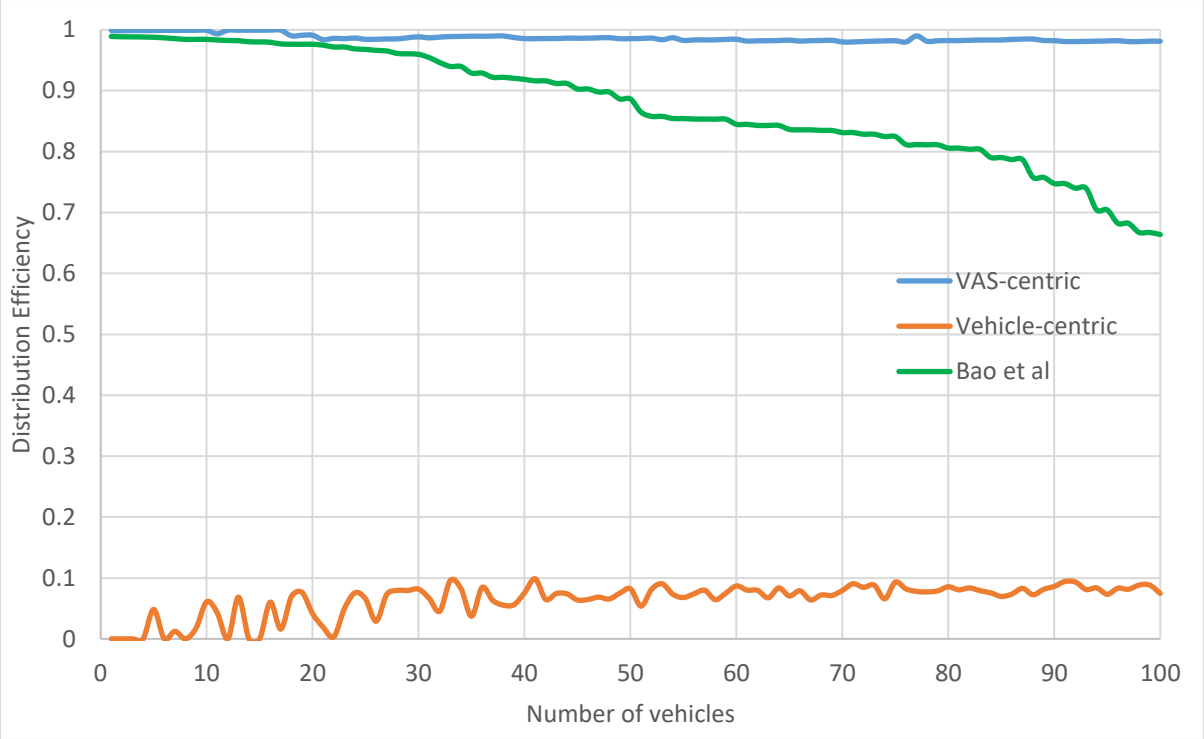


Figure 0.3: Distribution Efficiency against vehicles density

The impact of each distribution scheme on the performance of the safety message delivery and verification will be evaluated next.

Safety message latency: Simulations were conducted to evaluate the consequences of commitment message broadcast on the performance of safety message delivery and verification latency. The simulation reuses the ECDSA and TESLA processing parameters of chapter 3, Table 3.3. During simulation runs, each vehicle records the difference between the time when the verification of the safety message is complete and the safety message timestamp added by the sender. This end-to-end safety message latency includes time spent during scheduling, transmission, queue waiting, and processing times. Safety messages whose expiry time is exceeded are discarded, so if the end-to-end latency is greater than the message expiry time minus the time stamp, the message will not be verified and used.

Fig. 4.5 shows the curves of average end-to-end safety message latency as a function of traffic density for the three cases. In the VAS-centric case, in which only safety messages are broadcast on the direct channel, the average message latency is similar to the one obtained in Chapter 3 for the TESLA case. The approach has no impact on the performance of safety message verification latency. For the vehicle-centric case, in which commitment and safety messages are both sent on the broadcast channel, the average message latency increases with increasing traffic density. It can be seen that when there are 100 vehicles within range, the average message latency Is already approaching 100ms, which is the maximum tolerable message latency for time critical safety applications. This shows that the message latency is unacceptably high even at moderate densities. For the Bao et al solution, the average message latency is slightly more than that of VAS-centric as the number of vehicles increases. This is due to the processing of time-consuming BF messages received from RSUs, which are protected using ECDSA, making the safety messages stay longer in the verification queue. But as Bao et al solution is reactive i.e., the BF messages are only delivered on demand, the effect of ECDSA processing time on the safety message latency is small. A 5ms delay is observed when compared to the VAS-centric in the range of traffic density considered in the simulation.

These results highlight the effect of the vehicle-centric mechanism on the safety message latency compared to the VAS-centric and Bao et al commitment distribution mechanisms. Also, when compared with ECDSA latency results from Chapter 3, the message latency was around 65ms at a similar traffic density. This is means that the effect of the vehicle-centric approach on safety message latency is worse than that of the ECDSA case.

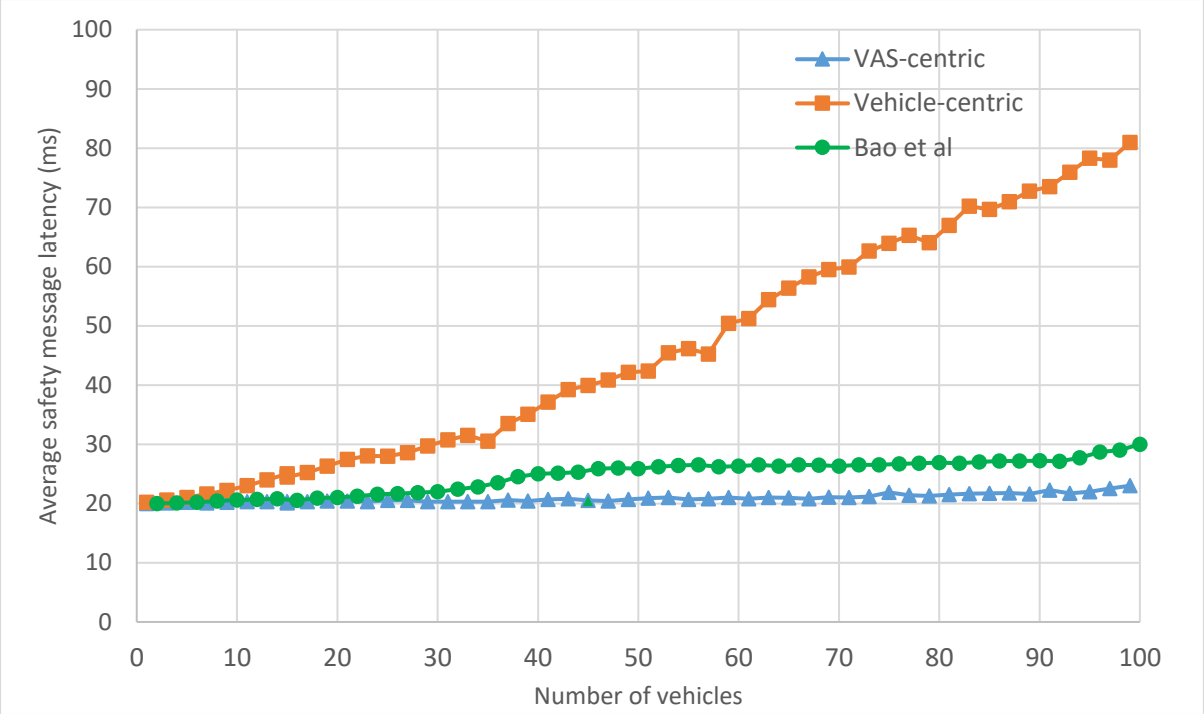


Figure 0.4: Average safety message latency against vehicle density

4.7 Security Analysis

This section presents the formal and informal security analysis of VAS-centric, vehicle-centric solutions and compares their security features with that of Bao et al. The formal security analysis is carried out using the widely accepted random oracle model to prove the semantic security of the commitment distribution schemes.

4.7.1 Random Oracle Model (ROM)

The Random oracle model (ROM) is a theoretical framework used in cryptography to analyse and design cryptographic protocols and algorithms [107]. In ROM, a “random oracle” is an idealised mathematical function that takes inputs and produces random and unpredictable outputs. A number of studies have applied ROM to analyse and prove the correctness of their different V2V message security schemes e.g., [108] [109] [110]. The same approached is followed to prove the security of the VAS-centric and vehicle-centric solutions as described below.

4.7.1.1 VAS-centric

Here, we want to prove the semantic security of the shared secret key between the VAS and the vehicles.

Participants: there are two participants: the VAS and the vehicle V_i. Let the instance t of vehicle V_i be $\prod_{Vi}^{t}$, and the instance of u of the VAS be expressed as $\prod_{VAS}^{u}$,. These instances are called oracles.

Accepted State: An instance $\prod_{.}^{t}$. will be in an accepted state, if it moves into an accepted state upon receiving the last expected commitment message. The session identification (sid) of $\prod_{.}^{t}$. for the current session is constructed by concatenating all commitment messages by $\prod_{.}^{t}$. in order.

Partnering: Two instances $\prod_{.}^{t1}$. and $\prod_{.}^{t2}$. are partnered to each other if the following three conditions hold: both $\prod_{.}^{t1}$. and $\prod_{.}^{t2}$. are in accepted state, both $\prod_{.}^{t1}$. and $\prod_{.}^{t2}$. are mutually authenticated and share the same *sid*, and $\prod_{.}^{t1}$. and $\prod_{.}^{t2}$. are mutual partners of each other.

Freshness: The instance $\prod_{Vi}^{t}$, or $\prod_{VAS}^{u}$, is considered fresh if the shared key K_{Vi}^{VAS} between Vi and VAS has not been revealed to an attacker *A*. It is assumed that *A* can modify, fabricate commitment messages exchanged between the VAS and the vehicles. thus, the attacker *A* will have access to the following oracle queries:

- $EXE(\prod_{.}^{t}, \prod_{.}^{u}.)$: the attacker executes this query to obtain the messages exchanged between two honest entities, which is modelled as an eavesdropping attack
- $SND(\prod_{.}^{t}, \prod_{.}^{u}.)$: this query models an active attack in which the attacker sends a commitment message *Msg* to a participant instance $\prod_{.}^{t}$,.
- $RVL(\prod_{.}^{t}, \prod_{.}^{u}.)$: This query models ability of the attacker A to force $\prod_{.}^{t}$, and/or its partner to reveal the shared key K_{Vi}^{VAS}.
- $TEST(\prod_{.}^{t}, \prod_{.}^{u}.)$: this query models the semantic security of the shared key following the indistinguishability in the random oracle model. A coin c is flipped, whose value is only known to the attacker and is used to decide the output of *TEST()* query. When the attacker executes *TEST()* and K_{Vi}^{VAS} is fresh, the instance $\prod_{.}^{t}$, returns the correct shared key if c=1 or a random number in the same domain if c=0; otherwise it outputs a null value.

Semantic Security of the shared key: In the random oracle model, the attacker is challenged in an experiment to distinguish between an instance's real shared key and a random key. The attacker can send several queries to either $\prod_{Vi}^{t}$, or $\prod_{VAS}^{u}$, instances. The output must be consistent with respect to the random bit c. In the end, the attacker returns a guessed bit c' and wins the game if c==c'. let ω denotes an event that the attacker can win the game, the advantage of *A* in breaking the semantic security of the symmetric key K_{Vi}^{VAS} is given as $Adv^{VAS} = |2.\Pr[\omega] - 1|$. We say that the VAS-centric scheme is secure if $Adv^{VAS} \leq \varepsilon$, for negligible value of $\varepsilon > 0$.

Recall that the VAS shares the credentials $\{p, a, b, G, H, PK_{VAS}\}$ to all vehicles. Thus, all nodes including the attacker knows the cryptographic hash function $H(.)$. The $H(.)$ is modelled in the random oracle as *HASH*.

Definition 1: A one-way collision resistant hash function $H: \{0,1\}^* \rightarrow \{0,1\}^l$ is a deterministic algorithm that takes an input as a binary string $s \in \{0,1\}^*$of arbitrary length and returns a binary string $H(s) \in \{0,1\}^l$ of fixed length *l* as output. Let Adv_A^{HASH} denotes the attacker's advantage in finding collision. $Adv_A^{HASH}(t) = Pr[(s_1, s_2) \leftarrow_R A: s_1 \neq s_2 \; with \; H(s_1) = H(s_1)]$, where $\Pr[.]$ represents the probability, and $(s_1, s_2) \leftarrow_R A$ means the attacker chooses (s_1, s_2) at random. By an (τ, t) attacker *A* attacking the collision resistance of $H(.)$, it means that the runtime of A is at most t and that $Adv_A^{HASH}(t) \leq \tau$.

Definition 2: Let $G \in E_p(a, b)$ be a base point in an elliptic curve $E_p(a, b)$. The elliptic curve computational Diffie-Hellman (ECCDH) problem is to find abG when given $aG$and bG, where $a, b \in \mathbb{Z}_p^*$. The attacker A has negligible advantage in solving ECCDH problem.

Theorem 1: Let the attacker *A* performs a polynomial number of oracle-queries t times against the VAS-centric scheme in random oracle model, then we have $Adv^{VAS} \leq \frac{q_h^2}{|HASH|} + 2Adv^{ECCDH}(t)$, where q_h is the number of *HASH* queries, $|HASH|$ is the range of space of the hash function H(.), and $Adv^{ECCDH}(t)$ is the advantage of the attacker in breaking the elliptic curve discrete logarithm problem.

Proof: To proof this theorem, we define a sequence of four games G_i for i=0 to 3. Let R_i represent the event that the attacker guesses the bit *c* correctly in game G_i. We start from an initial game, which reflects the real VAS-centric scheme. Then we modify it a few times until

we end up with the final game that reflects an attack against an ideal scheme, where an attacker A has a negligible advantage of winning. As the advantage between any two consecutive games is negligible, we can conclude that A has a negligible advantage of winning the game against the real VAS scheme.

Game G_0: This game corresponds to the real attack by A against the VAS-centric approach in random oracle model. The attacker begins by choosing a bit *c* at random. Thus, we have $Adv^{VAS} = |2.\Pr[R_0] - 1|$

Game G_1: In this end, the attacker begins with querying $EXE(\prod_{.}^{t}, \prod_{.}^{u}.)$ oracle to obtain the K_{Vi}^{VAS}. It then queries $TEST(\prod_{.}^{t}, \prod_{.}^{u}.)$ to test whether is able to derive the actual symmetric key or just a random number. For the attacker to compute K_{Vi}^{VAS} shared between VAS and the ith vehicle, the attacker must compute the shared secret $s = SK_{Vi}.PK_{VAS}$, $s = SK_{VAS}.PK_{Vi}$, using either the VAS's private key or the i^{th} vehicle's private key. It is computationally infeasible to derive SK_{VAS}or SK_{Vi} from PK_{VAS} or PK_{Vi} due to difficulty of solving the elliptic curve discrete logarithm problem. Also, the attacker has no knowledge of the KDF used at either ends. Thus, $\Pr[R_1] = \Pr[R_0]$

Game G_2: This game models an active attack by which the attacker creates a bogus commitment message appearing as a genuine one. G_2 is derived by adding the simulations of $SND(\prod_{.}^{t}, \prod_{.}^{u}.)$ and HASH queries. The attacker queries the HASH oracle several times to find collision. As each commitment message $\{ID_{Vi}, K_{0i}, T_{0i}, T_s, H(K_{0i}||T_{0i}||T_s)\}$ contains the vehicle's identity and a timestamp, no collision will occur when the attacker queries $SND(\prod_{.}^{t}, \prod_{.}^{u}.)$ oracle. According to the birthday paradox, the outcome of this game can be expressed as $|\Pr[R_1] - \Pr[R_2]| \leq \frac{q_h^2}{2|HASH|}$

Game G_3: Similar to G_2, this game is modelled as an active attack. $Adv^{ECCDH}(t)$is the advantage of the attacker in the experiment such that the attacker need to distinguish between the shared secret s and a random number. This becomes: $Pr[R_2] - \Pr[R_3]| \leq Adv^{ECDDH}(t)$

In this game, all the random oracles are simulated. The attacker is therefore only left to guess the bit c for winning the game after *TEST()* query. Thus, $\Pr[R_3] = 1/2$. From G_0, we have $\frac{1}{2}.Adv^{VAS} = \left|2.\Pr[R_0] - \frac{1}{2}\right| = \left|\Pr[R_1] - \frac{1}{2}\right|$. Applying the triangular inequality, we obtain

$\left|\Pr[R_1] - \frac{1}{2}\right| = |\Pr[R_1] - \Pr[R_3]| \leq |\Pr[R_1] - \Pr[R_2]| + |\Pr[R_2] - \Pr[R_3]| \leq \frac{q_h^2}{2|HASH|} +$ $Adv^{ECCDH}(t)$. Finally, we have $Adv^{VAS} \leq \frac{q_h^2}{2|HASH|} + 2.Adv^{ECCDH}(t)$, by taking the sum of all the games. By this, there is no more advantage to the attacker A to guess the shared key K_{Vi}^{VAS}, and $\Pr[R_3] = \frac{1}{2}$. Therefore, we proved that VAS-centric scheme is secure when the range of the hash function is large and the shared key K_{Vi}^{VAS} is secure under the assumption that the ECDL problem in *G* is unbreakable in the random oracle model.

4.7.1.2 Vehicle-centric

Here, we show that the signature used to secure the commitments cannot be forged under the random oracle model.

Let *A* the attacker performs an existential forgery, running in polynomial time against the vehicle-centric scheme in the random oracle model within a time bound T and with a probability of success µ. Assuming the attacker *A* can create a valid signature $\{g_1, H, g_2\}$ on the commitment message M. if $\{g_1, H, g_2\}$ can be simulated without knowledge of SK_{VAS}, with an indistinguishable distribution probability, then there is another machine that has control over the machine obtained from *A* replacing interaction with the VAS by simulation and compute valid signature $\{g_1, H, g_2\}$ such that $H \neq H'$ in expected time $T' \geq 120686QT/\partial$, where Q is the maximum number of queries that A can ask the random oracle, and R represent the maximum number of queries in which *A* can ask VAS.

Theorem: the vehicle-centric scheme can resist chosen message attack under the random oracle model.

Proof: Suppose that an instance of ECDLP is given and A could forge $\{V_i, M_i, s_i\}$, where $M_i = \{K_0||T_0||T_s\}$ signed commitment message of V_i sent by the VAS. A challenger Ĉ is set that can be able to solve the ECDLP problem with a non-negligible probability by running A as a subroutine with a probability that cannot be ignored.

Setup: the algorithm uses a secure parameter n as input. Ĉ selects a random number ***s*** as the system private key and then compute the corresponding public key $P_{pub} =$ s. G. Then Ĉ sends $\{p, H, E, G, P_{pub}\}$ to the attacker.

H_1*Hash Query:* When the attacker makes H_1query with commitment message M_i, Ĉ checks whether $\langle M_i, \tau_{H_1} \rangle$ is in the list L_{H1} or not. If it exists, then Ĉ sends $\tau_{H_1} = H_1(M_i)$ to the attacker. Otherwise, Ĉ chooses a number $\tau_{H_1} \in \mathbb{Z}_p^*$ at random, adds $\langle M_i, \tau_{H_1} \rangle$ into L_{H1} list, and then sends $\tau_{H_1} = H_1(M_i)$ to the attacker.

H_2*Hash Query:* Ĉ creates a list L_{H2}, and *A* makes a query with commitment message M_i. Then Ĉ checks to see whether the entities $s, H \in \mathbb{Z}_p^*$are in L_{H2}or not. If false, then Ĉ chooses a number $\tau_{H_2} \in \mathbb{Z}_p^*$ at random, adds the $\langle TC_{Vi}, PK_{VAS} \rangle$ into the list L_{H2} and send A, τ_{H_2}. Otherwise, Ĉ directly sends $\tau_{H_2}$to *A*.

Sign Query: Upon receiving A's query on commitment message M_i, Ĉ generate random numbers $\mu_1, \mu_2 \in \mathbb{Z}_p^*$, a random point G in curve E, and computes $TC_{Vi} = \frac{sG - \mu_1 . PK_{VAS}}{\mu_2}$. It then adds $\langle M_i, \mu_1 \rangle$ into L_{H1} and $\langle TC_{Vi}, \mu_2 \rangle$ into L_{H2}. Following the Forking Lemma theorem [111], Ĉ replies the attacker with the same random entities, and it gets two valid signatures $g = w\mu_1 + r\mu_2$ and $g' = w\mu_1 + r\mu_2'$ that can be used to compute x as follows:

$$s = \frac{g.\mu_2' - \mu_1 g'}{\mu_1(\mu_2' - \mu_2)} mod\ q$$

As a result, Ĉ breaks the ECDLP within the expected time less than $120686QT/\partial$, if $\partial \geq \frac{10(R+1)(R+Q)}{q}$. However, this contradicts the difficulty of solving the ECDLP problem. Therefore, the signature of the VAS on the commitment messages is secure against forgery under the alternatively chosen message attack in the random oracle model.

4.7.2 Informal Security Analysis

Here we discuss the informal security analysis of the distribution solutions. We assumed that the VAS has high computational and memory resources and cannot be compromised.

Unforgeability of the Commitment: the VAS centric solution is secure against forgery of the commitment of a given vehicle. Assuming an attacker captures one of the elements K_{ji} where $i < n$, of key chain currently in used by V_j and wants to forge K_{j0}. This is not possible because it requires the attacker to break the cryptographic hash function HASH, which is modelled in the random oracle model. Also, the attacker needs to defeat the one-way property of the hash chain, which is computationally infeasible.

Bogus Commitment Key Messages: An attacker may attempt to distribute bogus commitment key messages to the vehicles so that verification of received safety messages will fail. However, in the VAS- and vehicle-centric cases, the messages are protected against forgery and modification by MACs and signatures respectively. In the VAS-centric case, the receiving vehicle will check that the commitment key message includes a MAC computed using K_{Vi}^{VAS}. As this key is not known to the attacker, it will not be able to generate a valid MAC, and verification of the bogus commitment key message will fail. For the vehicle-centric case, each commitment key message is signed by the VAS using SK_{VAS} and verified by the receiving vehicle using PK_{VAS}. An attacker cannot generate a valid signature, because he has no knowledge of SK_{VAS}, so a bogus message will fail verification. In the Bao et al solution, an attacker can send bogus commitment keys to vehicles. However, their verification will fail, since the bogus commitment key is not among the elements used in constructing the Bloom filter value by the RSU.

Denial of Service (DoS) Attack: An attacker may try to overwhelm vehicles' resources with frequent commitment message updates. In the VAS-centric case, commitments are sent to the VAS for distribution to relevant vehicles. The VAS can monitor update frequencies based on individual vehicles' key chain lifetime. If the VAS repeatedly receives new commitment keys before the current key chain in use by a given vehicle expires, then the VAS can become suspicious and apply appropriate sanctions to the misbehaving vehicle. In contrast, the vehicle-centric solution is vulnerable to DoS attack because an attacker can broadcast its commitment key repeatedly to cause congestion on the communication channel or to overwhelm the resources of the receiving vehicles with a frequent signature verification process. This is analogous to the vulnerability described in chapter 3 when the safety messages are protected with the ECDSA approach. Similarly, the Bao et al solution is also vulnerable because an attacker can repeatedly send its commitment key causing receiving vehicles to also repeatedly request BF from RSU for verification. This can lead to high communication overhead on the channel and also exhaust the vehicles' resources in verifying the RSU's signature for each received Bloom filter value.

Replay Attack: An attacker may capture and replay a legitimate vehicle's commitment key update messages to the VAS. However, there is no advantage to the attacker in replaying a commitment key message, rather such action would only increase the chances of message

delivery. Also, it is easier for the attacker to generate a new fictitious message rather than to wait, capture and repeat an existing one. Moreover, it is relatively easy to provide a defence mechanism to filter out repeat messages using either a timestamp or a sequence number.

Impersonation Attack: In VAS-centric case, an attacker can pretend to be the VAS, persuading vehicles to register with it rather than the real VAS. However, this can be defeated if the authentic VAS's certificates and other credentials are obtained by the vehicles during registrations or pre-installed on the OBU's. The certificate can then be used to verify the VAS's identity. For the vehicle-centric case, an attacker can impersonate a given vehicle V_A, and send out a commitment key using V_A's identity. However, its verification will fail because the attacker has no knowledge of SK_{VAS}. It is possible to impersonate both the RSU and a given vehicle in Bao el at case. An attacker can pretend to be an RSU, sending out BF value to vehicles instead of the legitimate RSU. However, this can be defeated if the vehicles have the legitimate RSUs public certificates, but the authors did not mention how the vehicles would obtain these certificates. The RSU's certificates can be attached along with each BF message or pre-provisioned to the vehicles. The former increases the BF message length, which leads to additional communication overhead, while the latter requires vehicles to obtain the certificates of all RSUs that they will likely encounter on the road. This means that vehicles will need to store as many public certificates as possible, which could result in high storage overhead. An attacker can impersonate a vehicle and send out its valid commitment key to a new vehicle as commitment keys are not signed when broadcasted. The commitment key will be verified with the BF value obtained from the RSU.

4.8 Summary

This chapter addresses the challenge of distributing commitments to vehicles in TESLA-like security schemes. We propose two techniques—vehicle-centric and VAS-centric. In the former, vehicles broadcast commitments periodically, while in the latter, a central server (VAS) selectively distributes commitments. We compare these methods to each other and to the Bao et al solution [25] found in the literature, through theoretical analysis, and simulations that represents a real V2V environment.

Table 4-5 provides a summary of the chapter's findings. For the VAS-centric approach, over 98% of commitments delivered are utilised by the vehicles. This results in a lower buffer requirement (86.4KB) than the vehicle-centric and Bao et al solutions. An important feature of the VAS-centric solution is that its operation does not delay the delivery of safety messages. This is because it uses the downlink channel for commitment distribution, whereas the safety message use the sidelink channel, and because it uses a symmetric MAC to secure commitment messages, which has low computational cost, rather than an asymmetric signature. The average delay of 21ms is largely due to TESLA's authentication delay. However, the timeliness is lower than that of the vehicle-centric approach, implying that some messages may experience delays in verification due to untimely commitments.

On the other hand, the vehicle-centric approach prioritizes timeliness and reliability over distribution efficiency. Due to its proactive nature, nearly all commitments arrive before they are needed, but only 6% of them are actually utilised. The latter leads to a 10x increase in storage requirements compared to the VAS-centric approach. One major drawback of the vehicle-centric approach is that it increases the safety message latency by an average of 45ms, due to its use of the sidelink channel and of ECDSA to protect the commitment messages.

The Bao et al solution performs better than the vehicle-centric approach but less than the VAS-centric approach on most criteria, and worst on timeliness. Its reactive nature makes it avoid distributing commitments unnecessarily, with fewer than 12% of commitments received ending up unused. As the Bao et al solution utilises the unicast channels for commitments distribution, safety messages experience an average delay of 25ms due to the use of ECDSA to protect the BF messages sent by RSUs. This delay is half of what the safety messages experienced in the vehicle-centric approach, and about a 20% increase compared

to the VAS-centric approach. However, the approach faces some practical challenges, relying on RSUs for commitment verification and demanding widespread RSU deployment. But, RSU coverage is limited, especially outside urban areas. Thus, a backup mechanism is needed for areas not covered by RSUs coverage. Also, Multi-RSU scenarios are not considered, requiring mechanisms for smooth handovers and coordination between vehicles connected to different RSUs.

Table 0-2: Comparison of Distribution Schemes

Metric	VAS-centric	Vehicle-centric	Bao et al
Timeliness (%)	93.75	99.33	73.24
Distribution Efficiency (%)	98.68	6.40	87.62
Impact of distribution scheme on safety message latency (ms)	20.87	45.45	25.01
Storage cost (KB)	86.4	1709	166

The formal and informal security analysis conducted indicates that both the VAS-centric and vehicle-centric approaches are secure and robust over the range of attacks described. For the VAS-centric approach, commitments distributed by the VAS are resistant to forgery and collision attacks due to the use of well-designed MACs, making it computationally infeasible for an attacker to generate valid commitment messages without knowledge of the secret key shared between the VAS and individual vehicles. It is infeasible for the attacker to know this shared secret key because of the hardness of solving ECCDHP in polynomial time. For the vehicle-centric approach, the VAS's signature on the commitments guarantees the authenticity and integrity of the commitments, provides non-repudiation, and ensures unforgeability. However, a computation-based DoS attack is possible in the vehicle-centric case if a malicious agent increases the rate of commitment distribution. Similarly, the Bao et al solution is vulnerable to malicious agents exploiting the on-demand reactive approach.

Thus, the VAS-centric solution is the most promising from both the performance and security points of view, and should be the choice for use in practice.

Chapter 5: Analysis of Approaches for Reducing TESLA's Authentication Delay

The performance evaluation presented in Chapter 3 indicates that TESLA offers a viable alternative to VPKI-based ECDSA schemes for providing authenticity and integrity protection for V2V safety messages. However, the authentication delay that is inherent to the TESLA approach gives rise to a minimum message latency that is likely to exceed that tolerable by future safety applications. This chapter analyses potential ways to eliminate or reduce this authentication delay.

5.1 Problem Statement

Future V2V safety applications designed to support advanced safety services through fine-grained coordination between vehicles based on periodic exchange of V2V messages have more stringent performance requirements, most notably low message latency, than the first generation set of safety applications (sometimes referred to as day-one safety applications). Cooperative sensing, cooperative collision avoidance and convoy management (platooning) are typical use cases, some of which have a maximum tolerable end-to-end message latency of around 20ms and the frequency of messages as high as 100Hz [45]. Such strict time constraints cannot all be achievable by standard TESLA due to the delay in disclosure of the message verification keys. The minimum latency to receive the message verification key (or otherwise the minimum TESLA authentication delay) alone was estimated to be 12ms, as indicated in Chapter 3. This value contributes up to 50% of the overall end-to-end safety message delay, which was found to be approximately 22ms in the same chapter. So when the end-to-end message latency exceeds the safety message lifetime, the message cannot be used. This would effectively affect the performance of the underlying safety applications. The problem we are addressing here is finding one or more mechanisms to eliminate or reduce the authentication delay in TESLA, in order to improve the verification latency performance of TESLA-like security schemes in V2V and to make TESLA-like security scheme cope with the stringent latency requirements of some of the future safety V2V use-cases.

5.2 Prompt Message Verification Model

5.2.1 Outline of model

This section considers a simplified version of TESLA in which the authentication delay is removed entirely. This approach involves the use of a fresh key to sign each message, and the key being broadcast with the message. As in standard TESLA, keys are generated using a one-way hash-chain technique and verified using the commitment key of the chain. The benefits of this approach are: 1) message verification can take place immediately, 2) it frees-up space as there is no longer a need to buffer received messages temporarily, which in turn eliminates the risk of so-called memory-based denial of service (DoS) attacks, and 3) there is no need for synchronisation between senders and receivers. We will refer to this approach as the prompt verification (PV) model. In the PV model, we also assumed that vehicles used different key chains for the different major safety message types. For example, a sender signs a safety message of type CAM (messages that contain vehicle's status/kinematic information i.e. position, speed, etc.) with K^i_n, where K^i is an element of key chain n. Then, when the same vehicle is to send a DENM message type (i.e. event-driven messages that conveys warnings etc.), it signs the message with K^j_m, where K^j belongs to key chain m. This idea of using different key chains for different message types is introduced in order to limit the attacker's ability in exploiting the vulnerability in the PV's model, as will be discussed below.

5.2.2 Vulnerability of the PV Model

The benefits of this prompt verification approach come at the cost of a vulnerability that can be exploited by attackers. Suppose a malicious entity receives a legitimate message, extracts the key and immediately uses it to construct a fake message of its own choice, which it then broadcasts. In isolation, receivers would be unable to distinguish this from a genuine message without using contextual information. This is because the fake message will appear to be sent by the sender of the legitimate message, as it is protected with the sender's valid key and contains its identity.

Vehicles receiving both the legitimate message and the fake message would be able to tell the difference based on the order of arrival: the genuine message will always arrive first. This is because a) the total message path from sender to attacker to receiver cannot be shorter

than the direct path from sender to receiver, and b) the attacker needs time to process the original message and then construct and send the fake one. However, because of the unreliability and finite range of wireless communications, it is possible that some vehicles will only receive the fake message, which they would be unable to distinguish from a genuine one by cryptographic means.

Fig. 5.1 shows a typical scenario in which such a situation can happen. V_2 and V_3 receive both the original message ($M_{Si} | K^i_{nS}$) from V_S that is signed with K^i_{nS} and the fake message ($M'_{Si} | K^i_{nS}$) from the malicious vehicle also signed with the same K^i_{nS}, whereas V_1 receives only the fake message ($M'_{Si} | K^i_{nS}$). Thus, V_2 and V_3 can distinguish between M_{Si} and M'_{Si} based on their arrival times and the fact that each key K^i_{nS} can only be used once. On the other hand, V_1 cannot because M'_{Si} is the only message it receives that is signed with K^i_{nS}. So V_1 would assumed that ($M'_{Si} | K^i_{nS}$) is a genuine message sent by V_S.

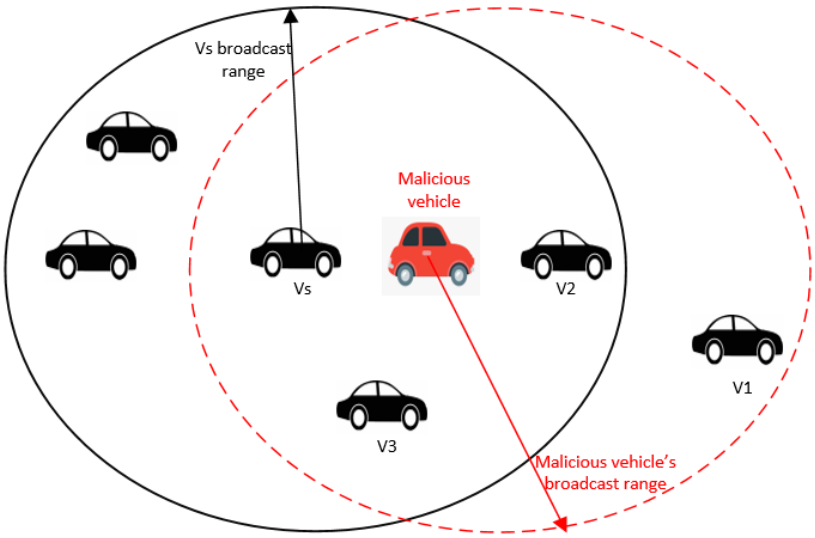


Figure 0.1: Scenario illustrating exploitation of PV model vulnerability

5.2.3 Evaluation scenario

5.2.3.1 Threat model

The basic threat model considered here is that a malicious agent wishes to influence the behaviour of a particular target vehicle, e.g. to make it divert from its route, crash, obstruct traffic, etc. It positions its vehicle appropriately relative to the target vehicle (e.g. following it

at a fixed distance) so that the target remains within range of its own messages for an extended period of time. It then selects one or more legitimate vehicles to impersonate whose messages it can receive, but that it is judged that the target vehicle cannot. There are advantages in choosing vehicles travelling with a similar velocity to itself and the target, so that the favourable configuration is maintained. Thus, a suitable reference scenario is one in which traffic is travelling in lanes on a moderately-busy main road (e.g. a highway or city centre dual-carriageway), with the malicious vehicle positioned in front of (or behind) the target and matching speed with it. The vehicles to impersonate are then selected from among those a suitable distance behind (or in front of) the attacker's vehicle. To execute the attack, the malicious agent captures messages sent by the legitimate vehicles, extracts the keys and immediately uses them to construct fake messages designed to influence the target vehicle's behaviour, which it then broadcasts.

The type and content of the fake messages constructed by the attacker depend on the attacker's aim and what he is trying to achieve. However, the attacker will try as much as possible to make the fake messages consistent with the genuine ones by taking into account the type, format of the messages intended to reuse their keys and other traffic messages (those previously received by the target vehicle) in order to make the fake messages more convincing and avoid detection. So with the separate key chain for different message types employed in the PV model, the attacker would have to construct a message type that is consistent with that key chain, which limits its ability in exploiting the vulnerability. For example, If the attacker wants to announce a false event about a hazardous condition on the road, he would have to wait to receive a safety message of DENM type (not necessarily conveying a similar true event) in order for its fake message to be consistent with the key chain used for DENM type. Similarly, if the attacker wants to broadcast false kinematic information about the vehicle he wishes to impersonate in order to mislead the target vehicle to take incorrect decisions or fall into dangerous situations, the attacker needs to capture the legitimate vehicle's CAM message and reuse the CAM message type key chain.

Depending on the target behaviour the attacker wants to induce, it can pick a different vehicle to impersonate for each fake message, or use the disclosed keys of the same vehicle to broadcast a series of consistent fake messages. The latter is appropriate e.g. when the malicious agent is trying to create a false movement path or to implicate a particular vehicle.

However, the attacker can only use each captured key once, so in this case it needs to remain in the legitimate sender's broadcast range to continue to receive its keys in order to send a series of consistent fake messages.

Variations on this scenario include the following:

- The attacker can impersonate other ITS-enabled entities e.g. RSU, roadside pedestrians, rather than simply vehicles, and reuse their keys.
- The attacker does not necessarily need to use a real vehicle, RSU, etc., in order to pretend to be one. Instead, the attacker could set up, for example, a laptop configured to act as an ITS node.
- There could be multiple adversaries acting independently of each other, or else colluding to achieve a common purpose. They could perhaps agree to impersonate the same vehicle and disseminate a stream of fake messages. In addition, the colluding adversaries could focus on a single target vehicle or vehicles within a given area of interest. For example, adversaries acting in a coordinated way could falsely report an event such as an accident or stopped vehicle on the road in order to convince other surrounding vehicles.

In this work, we focus on the scenario with a single adversary using a normal vehicle to exploit the vulnerability in the PV model. We consider the following four attack cases:

a. **Single impersonated vehicle, single fake message (SS)**: the malicious user picks a vehicle to impersonate and sends a single fake massage using its key.
b. **Single impersonated vehicle, multiple fake messages (SM)**: the malicious user picks a vehicle to impersonate and uses a sequence of its disclosed keys to generate a series of consistent fake messages. For example, when the attacker is trying to create a false movement path (trajectory).
c. **Multiple impersonated vehicles, with each apparently sending a single fake message (MS)**: the malicious user selects multiple vehicles to impersonate and uses their keys to send correlated fake messages, one apparently from each impersonated vehicle.

d. **Multiple impersonated vehicles, with each apparently sending a sequence of fake messages (I)**: the malicious user selects multiple vehicles to impersonate and uses their key sequences to send correlated fake message sequences (MM).

In all cases, the aim is to influence the behaviour of a single target vehicle and to avoid being implicated in the attack. In addition, we consider that the fake messages constructed by each malicious vehicle are of CAM type, containing false kinematic information (position, speed, acceleration, etc.) about the vehicle being impersonated. We focus on CAM messages because they are sent more frequently than DENM messages, so the attacker has more opportunities to send fake CAMs than fake DENMs. More importantly, many safety use-cases (e.g. longitudinal collision warning, forward collision warning, lane change assistance, etc.) rely on the processing of incoming vehicles' status update information, particularly position, speed and acceleration. This information is used to construct traffic views, assess traffic situations, evaluate potential collision risks and make decisions. Thus, a fake CAM message would potentially affect the operation and services of many safety use-cases, making it more attractive to adversaries than DENM. Moreover, some of the event-driven safety applications also require some information in the recent CAMs sent by the vehicle reporting the event to support their decision-making processes. For example, in the case of emergency electronic brake light warning, the facility layer entity responsible for processing the received DENM also takes into account the recent CAM sent by the vehicle reporting the warning in order to assess its relevance to the receiving vehicle before issuing it to the driver.

Note that malicious agents are not necessarily restricted by the same equipment limitations as a normal vehicle. For example, they can increase their transmission power in order to extend their broadcast range.

5.2.3.2 Vehicle Safety System and its Interaction with the Driver

In this section, we provide an overview of V2V messaging system, because it will be the basis for our analysis later on. As described in chapter 2, figure 2.2, the processing of received safety messages follows a layered model. In V2V, the message flow is basically arranged in a 4-layer stack, but for simplicity, we combined the two bottom: lowest layer (access and networking and transport), middle layer (facilities) and the applications layer at the top of the stack, comprising of one or more safety applications. Also, our description focuses on

safety/security-related decisions made at each layer. At the lowest layer, incoming messages undergo security checks based on cryptographic means, and other radio-related services e.g. received signal quality checks. Messages that successfully pass the security checks and whose signal strength is above a pre-defined threshold are passed up to the middle layer, while those not verified or their signal strength is below the required threshold get dropped. At the middle layer, the message content (position, time, and speed) is interpreted for plausibility/consistency and relevance checks are made, if required. In addition, the middle layer has access to the vehicle's own status data measured by the onboard sensor devices. This information is used for constructing outgoing messages and for safety application usage. The received message content and the vehicle's own status data are then passed to one or more safety applications at the upper level for evaluation of traffic safety situations, estimating collision risk and decision-making. If a potential danger is perceived (e.g. a collision risk), the safety application immediately communicates a warning to the driver in form of visual or haptic information or an audible signal in order for the driver to take appropriate action. The warning also includes a recommendation on actions to be taken as the perceived danger dictates. It continues until the perceived danger has been eliminated. For example in longitudinal collision risk warning (LCRW) [112], the safety application monitors and processes the relative distances and speeds of vehicles that are longitudinally aligned to estimate potential longitudinal collision risk between them. Upon detecting a collision risk, the LCRW sets its priority level to "1" (i.e. moves to warning mode) and then presents an informative message (e.g. estimated time to collision (TTC), distance to the predicted collision, etc.) to the driver with a recommendation on the actions to take, e.g., brake or change lane. If appropriate driver action has not been taken and TTC continues to decrease, the LCRW changes its priority level to "0" (change to a pre-crash phase) and may request (if applicable) one or more autonomous system in the vehicle such as an automatic braking system, to react directly to the situation.

In this work, we consider a manned vehicle and assume that the driver is in charge of controlling it (i.e. the vehicle does not act autonomously at all). Furthermore, we consider the following steps concerning the interactions between the 3-layer safety system on incoming messages and between the safety application and the driver:

1. Cryptographic authentication checks and tests based on low-latency detection methods (to be described later in 6.2.4.2) take place at the lowest layer, and if the message passes them, then it is forwarded up to the middle layer.
2. At the middle layer, the message content is checked for consistency with the laws of physics and with the receiver's understanding of the situation.
3. Messages passed to the safety application are used to update the receiver's awareness about the current traffic state and for situation assessment.
4. When the need to alert the driver is established, the safety application displays warning information to the driver with recommendations on actions to take
5. The driver reacts based on the safety application's recommendations and its understanding of the situation to avoid the perceived safety risk. This is similar to the argument presented in [113], in which the authors mentioned that the information presented to the driver in V2V environment is expected to improve the driver's situational awareness and its strategic decision. Thus, the driver's behaviour changes according to the information provided by the safety application.

Also, for the purpose of the PV model, we make the following assumptions concerning the behaviour of a receiving vehicle:

- When two messages are received that are apparently from the same sender and signed with the same key, the lowest layer drops the second message. It then flags subsequent messages from that sender as suspicious before passing them up the stack and alerts the middle layer of potential attacker activity involving that vehicle.
- At the middle layer, a sender's trajectory is flagged as suspicious if any two consecutive messages are found to be inconsistent. This recognises that it is possible to have a mixture of genuine and fake messages in the trajectory of a given sender without a clash occurring. Also, it is more likely that a completely fake trajectory is inconsistent than a completely genuine trajectory. Moreover, the longer a fake trajectory is, the more likely it is for a detectable inconsistency to appear in it.
- Messages or trajectories tagged as suspicious are processed with caution by the safety applications, and the driver is warned to exercise caution about decisions made based on them.

5.2.4 Analysis of the Vulnerability

This section examines the additional risk of impersonation attacks due to the use of PV model rather than standard TESLA. Note that the transmission of valid messages with false content is possible without impersonation, irrespective of the authentication method used. Malicious users can simply use valid credentials that they have acquired legitimately to broadcast messages containing false information. Several research works in the literature have acknowledged and investigated this security concern in the context of V2V [114] [115]. The main advantages to the attacker in impersonating innocent vehicles rather than using its own credentials are that:

- it helps it conceal its identity and avoid revocation of the credentials it is using in the event that a detection mechanism is put in place;
- there may be attacks in which it is important that the fake message appears to come from a specific vehicle or type of vehicle, e.g. an ambulance, or a vehicle belonging to a specific collection like a platoon
- it provides the attacker with the opportunity to intentionally implicate or destroy the reputation of a given legitimate vehicle

In order to assess the relative risk of PV's vulnerability to impersonation attacks, we consider in turn the following risk components for each of the cases, SS, SM, MS, and MM, outlined in section 5.2.3.1:

- **Feasibility:** this describes how complicated it is to carry out the attack.
- **Ease of Detection:** Can the attack easily be detected either by the driver of the target vehicle, the vehicle's ITS system, or other vehicles within the vicinity?
- **Success likelihood:** Given that an impersonation attack is carried out, what is the probability that it will be successful, i.e., that the target vehicle's behaviour will be influenced as intended by the attacker? This depends partially on the Feasibility and Ease of Detection. For example an attack that is easily detected by the driver or system is unlikely to succeed even if it is highly feasible.
- **Impact:** Given that an impersonation attack succeeds, how serious are the consequences likely to be for the target vehicle and other vehicles/road users within

the affected area. Impacts can range from minor inconvenience through disruption and delays to potential damage to vehicles and injury or loss of life.

5.2.4.1 Feasibility of the attack

Here we analyse how easily the different cases, SS, SM, MS Id MM, are to carry out.

In the SS case, the attacker simply stations itself at an appropriate distance from the target vehicle, and waits to receive a CAM message from a vehicle it judges to be out of range of the target. It then extracts the key from the message and uses it to construct the fake status update message, which it then broadcasts. To increase chances of success, the attacker may want to send further fake messages using keys of any other vehicles that are in a favourable position relative to itself and the target vehicle.

In the SM case, the attacker needs to send a series of consistent fake messages appearing to be from the same vehicle. This means that the selected vehicle the attacker is impersonating must remain within its communication range and outside the target's nominal communication range for the period of time required to generate a sufficiently long series of fake CAMs. If the selected vehicle goes out of the attacker's communication range e.g. the vehicle stops or speeds up, then the attacker will have to look for another vehicle to start afresh and look for a new vehicle to impersonate. Thus, this case is more difficult than SS. Clearly, the SM difficulty increases with the length of the message sequence that needs to be generated.

Suppose we write the SS feasibility as $F_{SS} = kP_I$, where P_I is the probability of finding a suitable vehicle to impersonate, then the SM feasibility can be written as $F_{SM} = kP_IP_T$, where P_T is the probability that an impersonated vehicle will remain suitable for long enough to complete a fake trajectory.

In the case of MS, the attacker sends a correlated set of single fake messages from multiple vehicles. Relative to the SS case, the feasibility depends on the probability of the attacker receiving CAMs from a sufficient number of vehicles that are out of range of the target within the relevant time window. This depends on the traffic density. Thus $F_{MS} = kP_I^N$, where N is the number of vehicles to be impersonated.

In the MM case, in which the attacker wants to create multiple fake trajectories using multiple impersonated vehicles, the attacker needs to continue receiving the CAM messages from several vehicles that are out of range of the target over an extended period of time. Thus $F_{MM} = kP_I^N P_T^N$.

It is easy for the attacker to exploit the PV vulnerability to perpetrate SS attacks. MS are more difficult because of the need to impersonate multiple vehicles, with the difficulty increasing with number of vehicles. Similarly, SM attacks are more difficult than SS due to the need for the impersonated vehicle to remain in range of the attacker and out of range of the target for an extended period of time. Whether or not SM is more difficult than MS depends on the relative sizes of P_I^N Id P_T. MM has the least feasibility of the four cases, as it is depressed by factors due to both number of impersonated vehicles and the length of fake trajectory.

5.2.4.2 Ease of Detection

We now look at possible ways by which the target vehicle could detect an attack. These can be divided into two categories:

1. Low-latency methods that use fields from the message envelope or header (timestamp, signature, key, etc.). If the message content is used at all, it is just treated as string and is not interpreted. The methods are applied at the lowest layer of the ITS stack, and can be used to reject a message at this layers or to tag it as suspicious before passing it to the facilities layer.
2. Model-based methods, applied within the facilities and application layers. These interpret and make use of the message content, which in the case of CAM messages concerns vehicle position, velocity, etc., and assess its consistency with or deviation from a model of the vehicle's environment.

Low latency methods:

- **Reception of two messages signed with the same key**: Although the attacker selects a vehicle to impersonate that is outside the expected reception range of the target, it is still possible that the target will receive a proportion of the messages it sends. As described above, the genuine message will always arrive first, and so the second message can immediately be discarded. The target will then be alert to attacker

activity, and subsequent (single) messages apparently from the same source can be tagged as suspicious. Clearly, the more fake messages sent to the target, the greater the probability that at least one will clash with receipt of a legitimate message.

- **Reception of message without commitment**. The commitment distribution mechanisms proposed in Chapter 5 are designed to ensure that commitments are received before any messages from relevant vehicles that use them. If the vehicle-centric approach is used, CAM and commitment messages from a legitimate vehicle should be received with equal probability. If significantly more CAM messages are received, this could be a sign that they are being sent by a different, nearer vehicle, because the proportion of messages received is a function of distance between sender and receiver. A comparison of commitment and CAM message reception could also be useful in the VAS-centric case, but it would need to be applied at the application layer.

Model-based methods:

- compare status update messages with previous ones received from the same vehicle. For instance, the target vehicle can compare the position information received in the current status update message with previous position information apparently from the same vehicle. If it notices any unreasonable movement or abrupt change in the position information, then the target may conclude that one or more of the CAM messages were fake. Further information or assumptions would be needed to decide which messages were fake and which genuine.
- compare status update messages received from different vehicles within the same vicinity. For each received message, the target vehicle can compare the message content with information received from apparently neighbouring vehicles. For example, it can check that the speeds of vehicles that claim to be in the same vicinity are consistent. If a small proportion of the vehicles display behaviour different from the majority, then it might plausibly be assumed that they are being impersonated. However, a large-scale MS attack could result in the fake vehicles being in the majority. Note also, that 'different' does not necessarily mean 'inconsistent'. In a multi-lane road traffic, vehicles in different lanes will often be moving at different

speeds. Even on roads with a single lane in each direction, vehicles may move at different speeds when traffic is light.

The ETSI standard also proposed the use of model-based detection techniques to validate the correctness of message content exchanged between vehicles. In ETSI TS 102 731 [116], the standard specifies a data plausibility checking called Validate Data Plausibility (VDP) to be part of the communication security services supported by an ITS station. Its main function is to compare information such as geographic position, time-of-day, and vehicle speed and direction received in an incoming ITS message with recently received data from available sources to validate whether the newly received information can be trusted on the basis of its plausibility. Similarly, the standard defines a facilities layer element called LDM (Local Dynamic Map) [117] for plausibility checking of incoming information. The LDM maintains a record of status information about surrounding traffic (including highly dynamic data i.e. CAM messages received from neighbouring vehicles) and provides this information to all safety applications that require it. Among the functional requirements of the LDM is to perform a basic plausibility checking on all incoming data before updating its record. For example, upon receiving a CAM status update, it establishes that the received information is consistent with information received previously from the same vehicle. Such services can be used to greatly reduce the range of false messages accepted as true by a receiver, regardless of whether they come from an attacker exploiting the PV vulnerability or abusing trust, or indeed generated unintentionally due to malfunctioning of on-board sensors. Even if an attacker can craft a message that is sufficiently plausible to fool the VDP service, the requirement for plausibility is likely to reduce the potential impact of the attack.

Aside from the ETSI standard specifications, several researchers in this domain have also investigated the concept of message consistency checking as a way of detecting messages with false content due to malicious behaviour or faulty nodes. Their works analyse and interpret different characteristics such as mobility information, application semantics, driving state, packet frequencies, etc. to detect inconsistencies or untrustworthy participants. For example, in [118], the authors propose a false message detection system that checks the consistency between the estimated and reported driving state (i.e., velocity, acceleration, brake status, steering angle) of a given sender. To detect a false message, the receiver utilises the sender's previous Driving state and the overall traffic flow (i.e., velocity deviation of and

inter-vehicular distances with nearby vehicles) to predict its current driving state. It then checks the consistency between the predicted driving state and the reported driving state, the consistency of the subsequent actions (i.e., driving states in the next few seconds) of the sender and its nearby vehicles. Similarly, the works of [119] utilises the message-sending-time in the message from the sender and message transmission time to estimate the location of the sender and then check whether it is the same as its reported location. However, the sender can send false message-sending-time to make it consistent with its reported false location. Also, this method assumes that the vehicle velocity does not change, so it cannot cope with the velocity change or traffic flow rate change.

The use of low-latency or model-based methods or a combination of both can be effective in detecting the different attack cases. It is likely that SS and MS attacks, which require only a single fake message from one or more impersonated vehicles, will be designed to provoke an immediate response, such as swerving or emergency braking. Consequently, rapid detection is required if the message is to be discarded or the emergency response over-ridden in time. The multi-vehicle consistency-checking method should be effective against SS attacks if it can be carried out sufficiently rapidly. This means use of low-latency methods or else extremely efficient, model-based one SM and MM attacks require one or more fake trajectories to be built up, giving more time for model-based techniques also to be effective.

5.2.4.3 Success likelihood

For an attack to succeed, it must avoid detection and rejection at all three stack levels and cause the driver to take the action intended by the attacker. We now consider likelihood of success, level by level, for each of the four cases in turn.

In the SS case, the success likelihood at level 1 is the likelihood that the single fake message will not clash with receipt of the legitimate message that is signed with the same key. If the attacker sets up the attack as described above, then the likelihood of the attack avoiding the message clash is high. At level 2, it is likely that the impersonated vehicle is out of range of the target, so it is unlikely the target has a recent status update from it. So if the attacker positioned the fake vehicle such that it avoid violating consistency e.g. ‘vehicle appears out of no where’, then it is likely for the fake message to be passed to the applications layer. When the single fake message contains, for example, a ‘slow vehicle’ type indicating that the fake

vehicle is 10m ahead of the target, the safety application will assess the situation. If it decides there is a potential safety risk, it displays this information to the driver and recommends him or her to take urgent action. As the driver receives no suspicious warning about the processed single fake message, it is likely for the driver to respond e.g., by braking suddenly, especially when moving at high speed and the visibility is low. Thus, the success likelihood of this attack is high.

Level 1 and 2 success likelihood of MS is similar to that of SS case, but in this case, the correlated set of single fake messages that are passed to the applications layer untagged will all be indicating the same situation. A typical example the attacker can use is the pre-crash sensing warning scenario. As described in ETSI TR 102 638 [44], each vehicle that detects an unavoidable collision (e.g., obstacles on road) should indicate a 'pre-crash' state in its CAM status update. So, the correlated set of single fake CAMs can all be indicating a false 'pre-crash' on the same lane the vehicle target is moving. So it is likely the driver becomes convinced since no corresponding warning alert is issued about the correlated messages processed. Consequently, the driver gets misled into performing incorrect actions, such as lane change, reducing speed, or even a complete stop.

In the SM case, the success likelihood at level 1 is the likelihood that no message in a fake sequence will clash with the receipt of a legitimate message signed with the same key. So, if the attacker sets up the attack as described above, then the likelihood of the attack avoiding the clashes is high. The success likelihood at level 2 is the likelihood of not receiving a mixture of fake and genuine messages in the sender's trajectory. So if the attacker gets the right traffic condition, then the likelihood of receiving only the sequence of fake messages is high. As the attacker is constructing a series of consistent fake messages, it is unlikely to detect inconsistencies in it, so the success likelihood at level 2 is high. Clearly, the longer the fake trajectory, the greater the chances of messages clashing, receiving mixed messages, and detecting inconsistencies in the sender's trajectory, which results in tagging subsequent messages as suspicious. For the proportion of fake messages that were passed to the safety applications untagged, they are likely to influence the target's behaviour using the scenario below. In LCRW, the use-case assesses CAM updates of all vehicles within the virtual safety shield of a given vehicle (a virtual area surrounding each vehicle such that when another vehicle enters it, extra particular attention is paid to avoid collision). If the attacker

strategically places the faked vehicle inside the safety shield of the target vehicle, the target's LCRW will closely monitor and process the sequence of fake status updates with target's own status updates. Eventually, it will be deduced that the two vehicles are in a critical safety situation, which results in issuing a false collision risk warning alert to the driver to start appropriate action to avoid the collision.

The more messages tagged as suspicious in the fake trajectory, the more the driver becomes aware of attack activity, and the less likely the SM attack becomes successful. Thus, the probability of the attack success is medium.

The success likelihood of MM is similar to that of SM, but in this case, the probability of success reduces. This is because the more fake messages sent to the target from multiple impersonated vehicles, the greater the probability of tagging messages as suspicious. For the proportion of fake messages that were passed to the safety applications untagged, they are likely to influence the target's behaviour using the scenario below. For example, in intersection collision risk warning (ICRW) [120], the use-case monitor and evaluate received CAM status updates of vehicles whose trajectories may cross at an intersection area. It then issues risk warning to drivers when a collision risk is perceived in order to take appropriate action e.g. apply brakes to stop or pay extra attention, to avoid the imminent intersection collision. So, the attacker can cause the target's ICRW to falsely issue a collision warning to the driver, and consequently lead to an unnecessary collision avoidance action.

SS and MS cases have a high success probability as a single fake message has a higher chance of passing the lowest and middle layer tests than a trajectory. This is followed by SM with a medium success probability, because of the likelihood of the driver being aware of the attack. MM has the least success likelihood of the four cases.

5.2.4.4 Impact

We now look at the potential consequences of the different attack cases. We categorise the potential consequences as follows:

- **High**: danger to human lives, cause injuries, physical damages to vehicles and other roadside infrastructure
- **Medium**: widespread traffic disruption, small hazards e.g. vehicles switching lanes

- **Low**: minor inconveniences e.g., slow down in traffic flow, delays to journeys.

In the SS case, where the attack indicates a sudden appearance of a 'slow vehicle' in front of the target, the driver may respond immediately e.g. by applying brakes to reduce its speed rapidly. This will plausibly cause a crash with following vehicles, which endangers human lives and physical damages to vehicles. Thus, the impact of this attack is potentially high. The MS case involves a sudden appearance of a pre-crash state indicated by vehicles in the same vicinity. It is plausible that the target and other vehicles around it start performing unnecessary lane switching or merging to avoid the affected lane, leading to disruption in traffic flow. Thus, the impact of this attack is medium.

In the SM case, the LCRW in the target vehicle monitors the fake trajectory the moment the faked vehicles enter the safety shield and informs the driver of potential danger in advance. The driver will plausibly not take urgent action as he is already aware ahead of time and can anticipate what might happen. The driver could slow down or be prepared to stop, and therefore, the attack has Iow impact. The MM attack case will yield similar impact events caused by the SM case. This is because as the target vehicle is approaching the intersection, the driver will be alerted earlier by the ICRW about vehicles that are likely to arrive at a similar time and other high speeding vehicles approaching the same intersection. It is plausible that the driver will look out for such vehicles, slow down and be prepared to stop and let the high-speed vehicles go pass.

SS has the most serious safety implications of the four cases, due to the sudden appearance of a false situation that is presented to the driver of the target vehicle, which in turn triggers unnecessary urgent actions. Followed by the MS, which will disrupt the traffic flow and possibly potential congestion. SM and MM attack cases have the least safety implications, as the sequences of fake updates are mostly monitored by the safety applications, which evaluate the situation and alert drivers on time, making them react in a safer way.

5.2.4.5 Risk Assessment

Here we combine the above risk components in order to evaluate the overall risk associated with the PV vulnerability to impersonation for the different attack cases. The attack feasibility

and ease of detection components contribute to the attack success likelihood. Then, the risk is defined as a function of the success likelihood and the impact, similar to the approach used in [121]. Table 6.1. summarises the analysis of the different attack cases based on the four risk components. To determine the overall risk rating for the different combinations of success likelihood and impact scales, we adopt a risk assessment approach described in [122], which is shown in Table 6.2. The authors use this approach to assess the risk of potential attacks within vehicles' onboard communication systems. It uses a five-point scale of risk, starting from 'noted' having the least risk level to 'critical' being the highest risk level.

Table 0-1: Summary of the Risk Components Assessments

Attack case	**Feasibility**	**Ease of detection**	**Success likelihood**	**Impact**	**Risk**
SS	High	Low	High	High	Critical
SM	Medium	Medium	Medium	Low	Low
MS	Medium	Low High	Medium	High	
MM	Low	Medium	Low	Low	Noted

Table 0-2: Risk Assessment [96]

Risk	**Success likelihood**			
		Low	**Medium**	**High**
Impact	**Low**	Noted	Low	Medium
	Medium	Low	Medium	High
	High	Medium	High	Critical

Using the information in both tables, it can be seen that the risk associated with the SS case is critical, as the attack is likely to succeed and its impact threatens the safety of vehicles and passengers onboard, which is contrary to the primary objectives of ITS systems. For MS case, its risk is high since the attack is likely to be successful but its consequences have no critical implications to road users and their properties. The risk posed by the SM case is low, since it

is unlikely to succeed and has minor consequences. The MM has the least risk of the four cases, and therefore, the attack is not significant with no safety implications.

5.2.5 Remarks on the PV Model

Although the PV model has several advantages (e.g., faster verification of received messages, eliminating the risk of memory-based DoS attack, etc.) over the standard TESLA, the vulnerability of PV to impersonation would enable a malicious agent to achieve a variety of ends with the fake messages. It is relatively easy to carry out a variety of attacks given the right traffic conditions. The probability of attack success is assessed to be high for some attacks cases (SS and MS) because it is difficult for the target vehicle to distinguish between genuine and fake messages under some circumstances. Consequently, attacks exploiting the vulnerability will affect the performance of safety applications since they rely on the periodic status update information exchanged between vehicles in their decision-making. It is plausible that the exploitation of PV's vulnerability to impersonation attacks could cause danger to human lives or damage to vehicles. Thus, the impact is potentially high. Based on the analysis of the SS attack case, the associated risk is assessed to be significant. Thus, a mechanism to mitigate PV's vulnerability is necessary.

5.3 Methods to Mitigate Prompt Verification's Vulnerability

The risk analysis discussed in the previous section shows that the PV's vulnerability to impersonation attacks is significant. The additional risk associated with some attack cases exploiting this vulnerability is high, and thus needs to be treated. In this section, we propose different mechanisms aimed at reducing some of the risk components outlined in 5.2.4, and assess their effectiveness in mitigating the vulnerability of PV. The proposed mitigation mechanisms should:

- Have low computational cost, such that the overall message latency is low compared to the TESLA's authentication delay
- Have low demand on the vehicle's on-board processing resources
- Have low communication overhead in terms of additional information that needs to be communicated to support its operation
- Not open another attacking opportunity to malicious agents

- Be practical to implement without major modification to the existing V2V messaging system

5.3.1 Reducing the Attack's Success likelihood

Below, we discuss some potential ways of reducing the success likelihood of the attack.

5.3.1.1 Use of VAS-centric Commitment Key Distribution

Recall that in both standard TESLA and the PV model, vehicles require the commitment of the key chain currently in use by a sender to verify the keys used to authenticate the messages received from it. This means that the target vehicle will not process fake messages unless it already has the impersonated vehicle's current commitment. As described in Chapter 5, the VAS-centric approach distributes commitments only to vehicles that are likely to be relevant to each other in the near future. This means that, if the VAS-centric approach is used, the target will only process fake messages if the impersonated vehicle has been relevant to it in the lifetime of its key chain. When the impersonated vehicle is relevant to the target, it is highly likely that the target vehicle will receive the genuine messages, because the message range is greater than the radius of relevance. In this case the fake messages will be discarded as discussed above in the PV model. This means that the target is only susceptible to attack if the impersonated vehicle has been relevant to it within the lifetime of the current key chain, but has since moved out of reception range of the target. Thus, the probability of success is greatly reduced.

Reducing the lifetime of the key chain will increase the effectiveness of the VAS-centric approach in reducing the success likelihood of the attack. However, it also means that commitments need to be generated and distributed more frequently, leading to increased message traffic between vehicles and VAS and a higher computational load on the VAS.

5.3.1.2 TESLA as a back-up Approach

Here we consider the use of standard TESLA in conjunction with the PV model for message authentication. In this case, each outgoing message is signed with two separate keys (i.e. a TESLA key and a PV key) belonging to different key chains. As in the standard TESLA approach, the TESLA key is kept and disclosed later, while the PV key is broadcast together with the message. So, an attacker impersonating a legitimate vehicle and reusing its PV key has no

knowledge of the corresponding TESLA key at the time it constructs its fake message. Thus, fake messages will certainly fail verification based on TESLA. Henceforth, this combined model will be referred to as PV+TESLA.

On arrival of a message, the receiver decides whether to verify it using PV, which can be done straightaway, but is less reliable, or using TESLA, which incurs a delay, or possibly both. It makes sense to use PV when some degree of trust in the apparent sender has already been established, or when it is likely that an attempted impersonation can be detected by other means.

For example, a receiver could use the TESLA approach in the following situations:

- receiving a message for the first time from an unknown sender
- receiving a message from a known sender that has not been heard from for a while.
- after detection of a suspicious event, e.g. receiving two messages apparently from the same sender with the same key.
- Randomly or when a set time has elapsed or a set number of messages has been received, since the last use of TESLA.

If the receiver uses the TESLA key to verify the first message in the sender's trajectory, then it is likely that the trajectory is a genuine one. It can then switch to verifying subsequent messages based on the PV key. To maintain trust in the sender's trajectory, the receiver should still perform a consistency check on messages appearing from the same sender. For example, if it is plausible that the sender could have reached its current position from its previously-reported position, then the receiver can continue to trust the messages based on PV. Suppose the receiver later notices an unreasonable step change in the movement of the sender, it may conclude that one or more of the messages were fake. Consequently, the receiver can fall back to using TESLA, and subsequently switch back to PV after several genuine messages have been confirmed. For a sender that was not heard from for a while and then reappears again, that looks like the signature of the 'classic' PV vulnerability attack. This is because the attacker is impersonating a vehicle in which the target has its mobility record in the past, but is currently out of the target's range. In this case, the receiver can use the TESLA key to verify the first message in the new trajectory of the sender. Also, when there is a message clash apparently from the same sender, the second message gets dropped as

described above. The TESLA key can be used to verify the next message appearing from the same sender so that if its verification succeeds, the message is passed up to the facilities layer for consistency checks with previous messages. At this point, there are at least two verified points on the sender's trajectory, so the receiver can use a consistency test to reject conflicting points on the sender's trajectory. The receiver can also randomly use TELSA to verify some messages from an already trusted trajectory. This would allow the receiver to have more confidence in an existing trusted trajectory.

The PV+TESLA approach serves as a means of establishing trust in the vehicle's trajectory. Also, it enables a receiver to distinguish conflicting trajectories relating to the same sender and subsequently reject inconsistent points. This can effectively reduce the success likelihood of the attacks.

Messages verified using TESLA incur the authentication delay, but since more messages are likely verified with the PV key than TESLA key, the average message delay should be better than using the TESLA approach alone. Previous research works have proposed approaches similar to the TESLA backup approach in order to reduce TESLA's authentication delay. In the works of [89] [123] and [124], the authors combine a prediction approach with standard TESLA. Each message is signed with a TESLA key and sent together with pre-computed information referred to as a "prediction outcome" that is used to predict the sender's future position in advance. With the prediction outcome known in advance, a receiver uses this information to reconstruct the sender's position and then compares it with the position information received in the following message. If the predicted position matches the reported position, then the message is verified, otherwise, it is verified later with the TESLA key. Their simulation results shows that the overall message verification delay is around 90% less relative to pure TESLA. However, the drawback of their scheme is that when the pre-computed information (i.e., preceding message) is lost in transmission, then the following message must be verified using the TESLA approach. This means that the more messages lost in transmission, the more TESLA is used for message verification.

The effectiveness of the above scheme depends solely on the accuracy of the mobility prediction model used to construct the prediction outcome. If a vehicle is unable to predict its own future state accurately based on its previous state, then validation of its prediction outcome will fail, and hence the corresponding message from it would have to be verified

with TESLA. This means that the less accurate the mobility prediction algorithm is, the higher the proportion of messages that need to be verified with TESLA. In addition, the generation and reconstruction of the prediction outcome itself incur some processing delay, which increases the overall message delay. Moreover, the size of the message is larger than the standard TESLA message due to the addition of the prediction outcome and other necessary information for its reconstruction, referred to as off-path nodes. Similar to standard TESLA, vehicles are required to buffer every prediction outcome received for the verification of the following message, in addition to storing the message itself if TESLA is to be used for verification. This increases the storage overhead and still opens the possibility of memory-based DoS attack that is inherent in standard TESLA. In our approach, there is no need for an additional algorithm and thus no extra computation is needed at the sender. Also, the performance of our approach does not depend on transmission losses, which is unpredictable. Moreover, messages are buffered only when TESLA will be used for verification. To increase the proportion of PV-based verification in our approach, a consistency check mechanism is needed. This will enable the receiver to maintain the use of PV on an already trusted trajectory.

5.3.2 Ease of Detection

Below, we discuss some potential ways of making it easier to detect fake messages.

5.3.2.1 Collaborative Defence

One way to reduce the risk arising from PV's vulnerability is to make it easier for the target vehicle to detect fake messages. If vehicles cooperate, they can share additional information that could benefit them all, and thus such information can be used to identify the fake messages. Although it may be possible for an attacker to position itself such that an intended receiver receives only the fake message and not the true one, it is likely that some legitimate vehicles in the vicinity of the attacker, including the impersonated vehicle will receive both messages and thus know that an attack activity is taking place. They could then broadcast a warning to alert other vehicles within range. There is no guarantee that the warning would reach all vulnerable vehicles, but the effective range of the warning could be increased by allowing a single rebroadcast.

It seems likely that this approach would be effective unless the vehicle density is very low. However, it has the following disadvantages:

- it increases the volume of message traffic
- A receiver has to wait before acting on an incoming message in case a warning is received, which increases message verification delay, whereas the whole point of the PV approach is to avoid delay.
- Neighbouring vehicle should not be relied on to send warnings as some may selfishly fail to do so.
- Malicious vehicles could send false warning messages.

Previous research works have investigated the use of collaboration among vehicles to detect false messages. For instance, in [125], a cooperative information exchange mechanism was proposed to detect vehicles sending false messages in V2V. The solution relies on a set of consistency checking rules obtained based on statistical techniques in order to determine whether a vehicle's behaviour is plausible or not. It extends the safety message with three fields: flow, average speed and density, which all vehicles must compute and transmit regularly. Each vehicle then uses this information to model the traffic around it. Hypothesis testing is employed to decide whether a received message is correct or not. If a given receiver detects an inconsistency in the sender's estimated flow value when compared with the overall flow of all surrounding neighbours, then a report is sent out to other vehicles informing them that the sender is fake and the type of attack being launched by it. Similarly, the authors in [126] propose a cooperative mechanism for position verification based on sensor measurement information exchanged among vehicles. In this scheme, each vehicle verifies its direct neighbour's position reported in the received message by comparing it with the position measured directly from the onboard sensor devices. It then assigns one of the three possible states to each of its neighbours: verified (advertised position corresponds to the true geographic position of the sender), faulty (advertised position does not correspond to the true position, tagged as an attacker) and unverifiable (information collected so far is not enough to determine the correctness of the reported position). The resulting neighbour states are then exchanged among the vehicles, which are then compared and cross-checked to detect vehicles that advertise false position information.

The above collaborative approaches require vehicles to carry out some computations e.g. estimation of flow values in [125], consistency checks with sensor measurements in [126], before exchanging information used for detecting the false messages. This introduces extra demand on the vehicle's on-board processing resources. Compared to our approach, a vehicle simply needs to send out a warning report (containing the false message) once it receives two messages signed with the same key from the same sender.

Although the collaborative defence approach has a high potential to improve the detection of fake messages, since vehicles close to the attacker will certainly be able to distinguish between genuine and fake messages and can assist others to do so, but the overhead associated with the approach is high compared to other mitigation techniques described above. The additional messages that need to be exchanged between vehicles are likely to overwhelm the transmission resources, especially in high traffic situations. Moreover, sending warning messages represents a new attacking opportunity that could be exploited by a malicious agent, and consequently results in a DoS attack.

5.3.3 Remarks on Mitigation Methods

The mitigation mechanisms described above are likely to be effective in reducing the risk of PV's vulnerability to impersonation attacks. All the three approaches discussed have less overhead compared to similar schemes proposed in the literature. However, the drawbacks associated with the collaborative defence approach could outweigh its benefits. Thus, the preferred models to adopt to deal with the vulnerability is a combination of PV with (a) VAS-centric commitment key distribution and (b) TESLA as a backup approach. In case (a), the combination seems plausible as it limits the window of opportunity to the attacker quite considerably, making it more difficult for the attacker to get the right traffic condition. Moreover, a commitment distribution scheme is needed, regardless of whether it is the standard TESLA or the PV model. In case (b), the combination could also be highly effective since it enables vehicles to establish trust on senders' trajectories, which in turn reduces the attack success likelihood.

5.4 Summary

In this chapter, the authentication delay that is inherent to the standard TESLA approach has been identified as a problem that hinders the effectiveness of TESLA-like security schemes in V2V, as it increases the message latency significantly, thereby affecting the performance of safety applications, particularly those having stringent message latency requirements. To address this issue, a modified version of TESLA called the PV model has been proposed and analysed. In the PV model, the verification key is broadcast with the message, removing the authentication delay entirely. Compared to the standard TESLA approach, the PV model provides several benefits, including faster verification of received messages, preventing the possibility of a memory-based DoS attack as there is no longer a need to temporarily store received messages and eliminate the need for synchronisation between senders and receivers. However, these benefits come at the cost of a vulnerability that can be exploited by malicious users. An attacker node with V2V capability can extract the key from a legitimate vehicle's message, and use it immediately to construct an arbitrary fake messages purporting to come from that vehicle. Vehicles receiving only the fake message are unable to distinguish it from a genuine one by cryptographic means, as the fake message is protected with the legitimate vehicle's valid key and contains its identity. But those receiving two messages using the same key will be aware of an attack activity as a given key is only used once in the PV model. The malicious users have some limitations in exploiting the PV's vulnerability. A new key must be extracted for every fake message to send. Also, the window of opportunity to use a given key is fairly short; if it delays using it, then the key would be invalidated by a later key, as keys are being used in sequential order.

To analyse this vulnerability thoroughly and examine its significance in the V2V domain, we developed a threat model consisting of different attack scenarios with the attacker constructing different fake messages designed to influence the behaviour of a given target vehicle e.g., cause it make incorrect decisions or fall into dangerous situations. Then we conducted a risk analysis based on four risk components i.e., feasibility, ease of detection, success likelihood and impact, in order to assess the additional risk associated with PV's vulnerability to impersonation attacks. Some attack scenarios e.g., SS and MS can easily be perpetrated and have a high probability of success given the right traffic conditions. The

attacks could endanger human lives and cause physical damage to vehicles and other infrastructure on the road. Based on this analysis, we conclude that the risk arising from this vulnerability is significant. Consequently, if the PV model is to be used, it needs to be integrated with a mitigation mechanism. We proposed and discussed several ways to reduce the risk of the PV's vulnerability. Among them, we found out that a combination of the PV model with the use of a VAS-centric commitment distribution scheme could be effective, as it makes the attack more difficult to be carried out. Alternatively, integrating the PV model with TESLA as a backup approach enables receivers to establish trust in the trajectory of senders by verifying first few points in it with TESLA keys, which reduces the success likelihood of the attack. For the proposed PV+TESLA model to be effective, a consistency checking mechanism is required, both to reduce the risk and also to minimise the effective authentication delay by increasing the use of PV relative to TESLA. In the next chapter, we will explore one such consistency checking technique that makes use of received signal strength (RSS) measurements.

Chapter 6: An RSSI-based Message Consistency Checking Scheme

The analysis of the PV model presented in Chapter 5 indicates that one of the ways to mitigate the risk of PV's vulnerability is by using TESLA as a backup. For this combination to work effectively, a criterion is needed for when to use PV alone and when to use TESLA. This should be based partially on the existence of evidence that an attack exploiting PV's vulnerability is in progress. A clear signature of such an attack is that the position information reported in a CAM is different from the actual position of the sender.

This can be achieved by comparing position information reported in the message with other sources of position information, including physical features of the transmission signals. This chapter proposes utilising the strength and variability of received signal strength indicator (RSSI) measurements to estimate the separation between sender and receiver. This can then be compared with the separation calculated from the receiver's known position and the claimed sender position reported in the message. The performance of this RSSI-based distance estimation scheme is evaluated through simulation work.

6.1 Background on Received Signal Strength Indicator (RSSI)

The received signal strength indicator (RSSI) is a commonly used measure of the power of a received radio signal. It is basically the ratio of the power measured at two different points, e.g.at the sender and the receiver, expressed in dB, i.e.

$$RSSI = 10 \log_{10} \frac{P}{P_0}$$

In the case of a non-directional broadcast signal through a uniform medium, the so-called log-distance path loss model (LDPLM) is widely used to estimate the RSSI at a receiver, as described in [127] [128]:

$$RSSI \approx A - 10B \log_{10} \frac{d}{d_0} \qquad (6.1)$$

where d is the distance from the transmitter, d_0 is a reference distance that is usually taken to be 1 meter, and A and B are positive constants. A depends on the transmitter and receiver characteristics, and B, the path loss exponent, depends on the transmission medium.

Fig. 6.1 shows how RSSI decreases with distance according to equation 6.1. The values of B are taken from [129] and are based on real measurements at 5.9GHz taken in different V2V environments: campus which can be interpreted as urban with low density (B=1.66), rural (B=1.89) and urban (B=2.56). A value of 32dBm has been used for A, which is not dependent on the environment. These values are generally consistent with those reported by other studies, e.g., the works of [130].

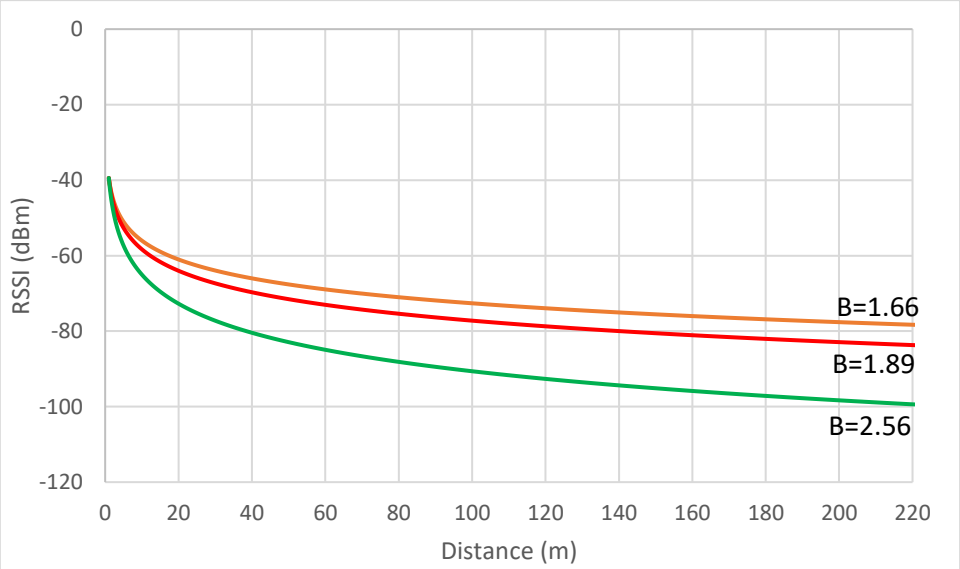


Figure 0.1: RSSI variation with distance

The expression for RSSI in (6.1) is a monotonically decreasing function of d and can readily be inverted to obtain an estimate of d given a measurement of RSSI provided A and B are known. Taking d_0 to be 1m:

$$d = 10^{\left(\frac{A-RSSI}{10B}\right)} \tag{6.2}$$

Assuming (6.1) is accurate, A and B can be obtained using RSSI and d from two trusted messages, then (6.2) can be used to predict d for subsequent, untrusted messages. The estimated value from this formula can be compared with the distance between the known position of the receiver and the claimed position of the sender as a consistency check.

There are complications that make the use of the pure LDPLM approach for distance estimate difficult to use in practice. Firstly, the LDPLM only really applies to propagation in free space. For example, one correction that is frequently applied is to allow for interference between

the radio waves travelling directly from sender to receiver and those reaching the receiver after reflection from the road surface. Even if the LDPLM is a good approximation at long distances, the presence of static and moving obstacles such as buildings, foliage and vehicles not only tends to attenuate the signal, but also introduces a considerable variation of RSSI due to absorption, reflection, refraction, and multi-path interference. Indeed, a more general form of LDPLM adds a Gaussian random variable with a mean value of 0 to the right-hand side of (6.1) to take such effects into account. This may be interpreted as a margin of error on the expected RSSI value at a given distance of ±σ, the standard deviation of the random term. This can be translated to an uncertainty on the estimated distance between sender and receiver.

Fig. 6.2 shows a scatter plot of values of σ and B extracted from published real measurements in the 5.9GHz frequency band obtained in different V2V environments (see [127] [130] [131]). The red dots correspond to NLoS condition in urban, suburban, highway, and rural areas, and the blue dots are the LoS condition in the same location.

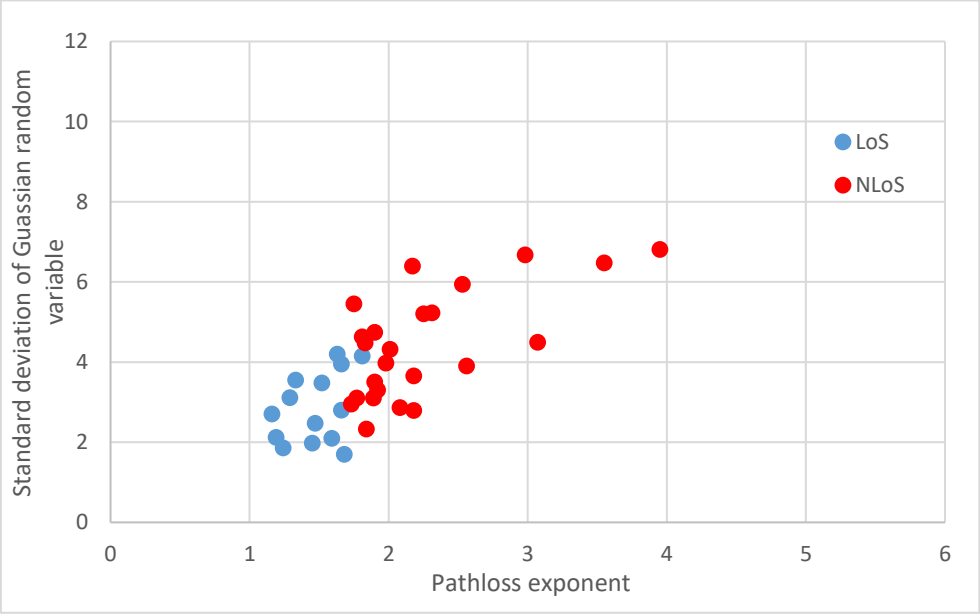


Figure 0.2: Correlation of standard deviation of Gaussian random variable with pathloss exponent

A degree of correlation between σ and B is evident as shown in the figure. Both σ and B tend to be smaller in situations where the link between sender and receiver has a clear path, and

larger when it is obstructed by buildings or other vehicles, which induce additional propagation losses.

We will now discuss some existing research studies that use RSSI-based techniques before presenting our method for evaluating the consistency of actual and claimed vehicle locations.

6.2 Literature Review of RSSI-based Techniques

RSSI-based techniques have low computational cost and require no extra hardware. Consequently, several existing research projects have proposed RSSI-based solutions to issues in V2V. However, in order to cope with the lack of accurate representation of the sender-receiver link in the propagation models including the LDPLM outlined above, some research works use the ratio of RSSI measurements taken from multiple receivers. Others compare similarities in the RSSI patterns. The most common applications of these RSSI-based techniques are Sybil node detection and localisation of vehicles. Works from these two categories are described below:

6.2.1 Sybil Node Detection

RSSI-comparison techniques have been proposed as a means of detecting fake vehicles fabricated by malicious agents (so-called Sybil nodes). The assumption behind this approach is that the Sybil nodes transmit messages that are actually sent by the same physical node, and so they share similar signal characteristics with each other and with genuine messages from that node. Thus, if some nodes have very similar RSSI distributions or time sequences, they are identified as Sybil nodes. For example, the works of [129] and [132] record successive RSSI values to obtain RSSI time sequences apparently corresponding to different vehicles. The sequences corresponding to Sybil nodes fabricated by the same vehicle will be identical (or at least very similar). A malicious node may regularly vary the transmit power of all Sybil nodes created by it in order to avoid being detected by the RSSI-based techniques that look for similarity in signal characteristics. To address this problem, the study in [133] proposes a power control Sybil attack detection (PCISAD) method, which find Sybil nodes by observing abrupt changes in the mean of the RSSI time sequence received within a very short period using a linear support vector machine classifier.

6.2.2 Localisation of Vehicles

There are several schemes that use RSSI to estimate the location of vehicles. For example, the authors of [134] describe a cooperative approach, whereby neighbouring vehicles collaborate to localise a target vehicle. Each vehicle estimates its distance to the target vehicle using the LDPLM formula and then sends the estimated distance and its current location to a chosen vehicle called the observer. The observer processes the aggregated information and advertises the target vehicle's approximate actual location. The study in [135] describe an RSSI-based localization mechanism that use nearby stationary roadside units (RSU) to estimate the location of a target vehicle. Each RSU measures the RSSI values of the vehicle being localised and uses it to estimate its distance. The distance estimations are shared between the RSUs, which then compute the actual position of the target vehicle using triangulation.

Schemes like these, are cooperative in nature, meaning that they are dependent on nearby vehicles or RSUs to send distance estimates and provide correct information in order for them function. The fewer vehicles or RSUs in the vicinity, the less effective they become and vice-versa. Moreover, they add traffic to the network due to the need to send distance estimates, which increases bandwidth consumption, and incurs a latency penalty as the observer must wait to receive distance estimates from other nodes.

Below, we discuss our proposed method to address the challenges of using RSSI-based approach for distance estimation.

6.3 The RMCCS Method

As described in 6.1, in principle, the inverted LDPLM formula, (6.2), can be used to estimate the true distance between two vehicles given the RSSI value measured by the receiver. In practice, the following complications make this difficult:

- The path loss exponent that is appropriate to the environment in which the vehicles are situated must be known in advance or measured in some way;
- RSSI measurements at a given separation exhibit considerable variability in some environments;

- Even within a given environment, the path loss exponent and RSSI variability can change abruptly. For example, in a city centre, propagation conditions can change from “heavily obstructed” to “line of sight” and back again when the receiving or transmitting vehicle passes a road junction.

In this section, we describe an RSSI-based Message Consistency Checking Scheme (RMCCS) method, which is our proposal to cope with these complications and provide distance estimates without any underlying assumption about the sender-receiver location or the condition of the link between them e.g. whether there is a clear line-of-sight (LoS) path or the link is obstructed by nearby objects.

The RMCCS method is based on the LDPLM formula with Gaussian noise, but with the additional assumption of a relationship linking the path loss exponent to the standard deviation of the Gaussian variable. If these two parameters are correlated, we can use measurements of RSSI variability alongside its mean value to obtain estimates of distance and the associated uncertainty that could be used to assess the likely truth of a reported position and give a measure of confidence on this assessment.

Suppose that B and σ are functions of a common hidden variable, γ, that characterises the nature of the obstacles on or near the path between them, for example,

$$B = \gamma B_0, and\ \sigma = k(\gamma - \gamma_0)$$

$$B = \left(\frac{\sigma}{k} + \gamma_0\right) B_0 \qquad (6.3)$$

where γ = 1 corresponds to LOS conditions, k is a constant of proportionality, and $\gamma_0 \leq 1$ allows for the possibility of variation in RSSI even in LOS conditions. Given measurements of RSSI and σ, the distance between sender and receiver, can be estimated as:

$$\hat{d} = 10^{\left[\frac{A-RSSI}{10.\left(\frac{\sigma}{k}+\gamma_0\right)B_0}\right]} \qquad (6.4)$$

where the hat symbol, ^, means an estimate of a variable, (i.e. $\hat{d}$ is an estimate of d), and the uncertainty on this value as:

$$\bar{\sigma}_d = \hat{d}.\frac{(10^{\Gamma}-10^{-\Gamma})}{2} \text{ where, } \Gamma = \frac{\sigma}{10B_0\left(\frac{\sigma}{k}+\gamma_0\right)} \qquad (6.5)$$

Scientists often use the ratio of the discrepancy between a measured value and the theoretical prediction to the standard deviation as a measure of agreement between the two. Similarly, engineers use this ratio as a measure of whether a manufacturing process is sufficiently precise. If data is normally distributed, then 68% of measurements will be within 1 standard deviation of the mean, 95% within 2 standard deviations and 99.7% within 3 standard deviations. In the same way, we can use, $\frac{|\hat{d}-d_r|}{\bar{\sigma}_d}$ where d_r is the distance based on the position of the sender as reported in the message, as a measure of the inconsistency of the reported position and the measured signal strength and variation. The inconsistency threshold beyond which a vehicle is suspected of lying about its position would then be expressed in terms of a multiple of the standard deviation ('n sigma'). Raising the threshold will reduce false positives (i.e. the number of truthful vehicles judged to be lying) and true negatives, but increase false negatives and true positives. Choosing the threshold value to use involves minimising some function of these quantities.

Note that, due to the logarithmic dependence of RSSI on distance in (1), if σ is independent of distance, then $\bar{\sigma}_d$ increases linearly with distance. Thus, a given discrepancy $\Delta d = |\hat{d} - d_r|$ may be regarded as inconsistent for small $\hat{d}$ and consistent for large $\hat{d}$.

To compute $\hat{d}$ and $\bar{\sigma}_d$, the receiving vehicle will need to extract estimates of the mean RSSI and the corresponding standard deviation from the noisy RSSI signal. This can be achieved using standard signal processing techniques such as Kalman and Savistzky-Golay filtering algorithms, as will now be discussed:

Kalman Filter: This is a mathematical algorithm that provides estimates of unknown variables given inaccurate and uncertain measurements of a dynamic system observed over time. Also, it provides a prediction of the future system state based on past estimations [136]. The Kalman filter is one of the most important developments of linear estimation theory, and has demonstrated its usefulness in various applications such as data smoothing (i.e., removing randomness in data measurements), target tracking, control systems, robotic motion, etc. The Kalman filter operates by repeating two main steps: prediction and update, to minimise the mean square error of the estimated variables. The efficiency of the Kalman filter is due to its low computational requirements, well-designed recursive properties, and suitability for real-time implementation. Using a Kalman filter, the noisy RSSI measurements can be

processed to obtain a mean estimation of the RSSI and corresponding standard deviation. Existing research works that apply this approach include [137] [138]. Others such as [139] [140] use it to improve relative positioning between vehicles.

Savistzky-Golay filter: This is another method for data smoothing based on local least-squares polynomial approximations [141]. Typically, the filter is applied to a series of digital data points to reduce noise from the high and low frequency components in the data, without deforming the signal. To achieve this, the subsets of consecutive data points are fitted using a low order polynomial with the linear least square method, and the convolution of all the polynomial is then computed. This filter is widely used in filtering noisy signals, especially in the fields of signal processing and image processing. As described in [142], the Savitzky—Golay filter can be applied to any consecutive data when the points of the data are at a fixed and uniform interval along the chosen abscissa, and the curves formed by graphing the points must be continuous and more or less smooth. This makes the filter suitable to smooth out the noisy RSSI measurements, obtain a continuous mean and the corresponding standard deviation. Previous research works have also used this filter in the context of V2V. For example, the studies in [143] applied it to smooth RSSI signals, which are then used to generate unique hashing keys called link fingerprints, between any pair of communicating vehicles. Similarly, in their RSSI-based Sybil detection method, the authors of [144] use the same filter to remove noise in the RSSI measurements.

6.4 Evaluation of RMCCS

In this section, we assess the validity and effectiveness of the RMCSS approach using RSSI measurement data obtained from a simulation framework that takes into account different propagation effects in the context of V2V.

6.4.1 Generation of RSSI data

It is not practical to gather data to evaluate our method by conducting field trials with real vehicles and/or radio equipment in realistic settings. Instead, we use the GEMV^2 simulation software [145], a location-specific geometry-based propagation model that combines deterministic and stochastics channel properties. It distinguishes V2V link into three types: line-of-sight (LoS), non-line-of-sight due to vehicles (NLoSv), and NLoS due to buildings or

foliage (NLoSb). Since each link type posses different channel characteristics, the propagation mechanisms such as scattering, diffraction, and reflection are computed separately. The presence of vehicles, buildings, and foliage in the location is modelled using the geographical descriptors of the objects i.e., outlines of buildings, foliage, and vehicles on the road. In order to achieve improved accuracy, GEMV^2 defines two propagation mechanism for each link type: (a) large-scale signal variations, modelling distance-dependent attenuation and large-scale fading, and (b) small-scale signal variations, which models variations due to obstructions caused by surrounding objects. The large-scale signal variations are computed deterministically whereas the small-scale signal variations are computed stochastically. The developers of GEMV^2 have validated it against measurements performed in urban, suburban, highway, and open space conditions. Research studies e.g., [146] [147] have used GEMV^2 to obtain RSSI measurements in order to assess the impact of propagation characteristics on V2V communication performance in different locations.

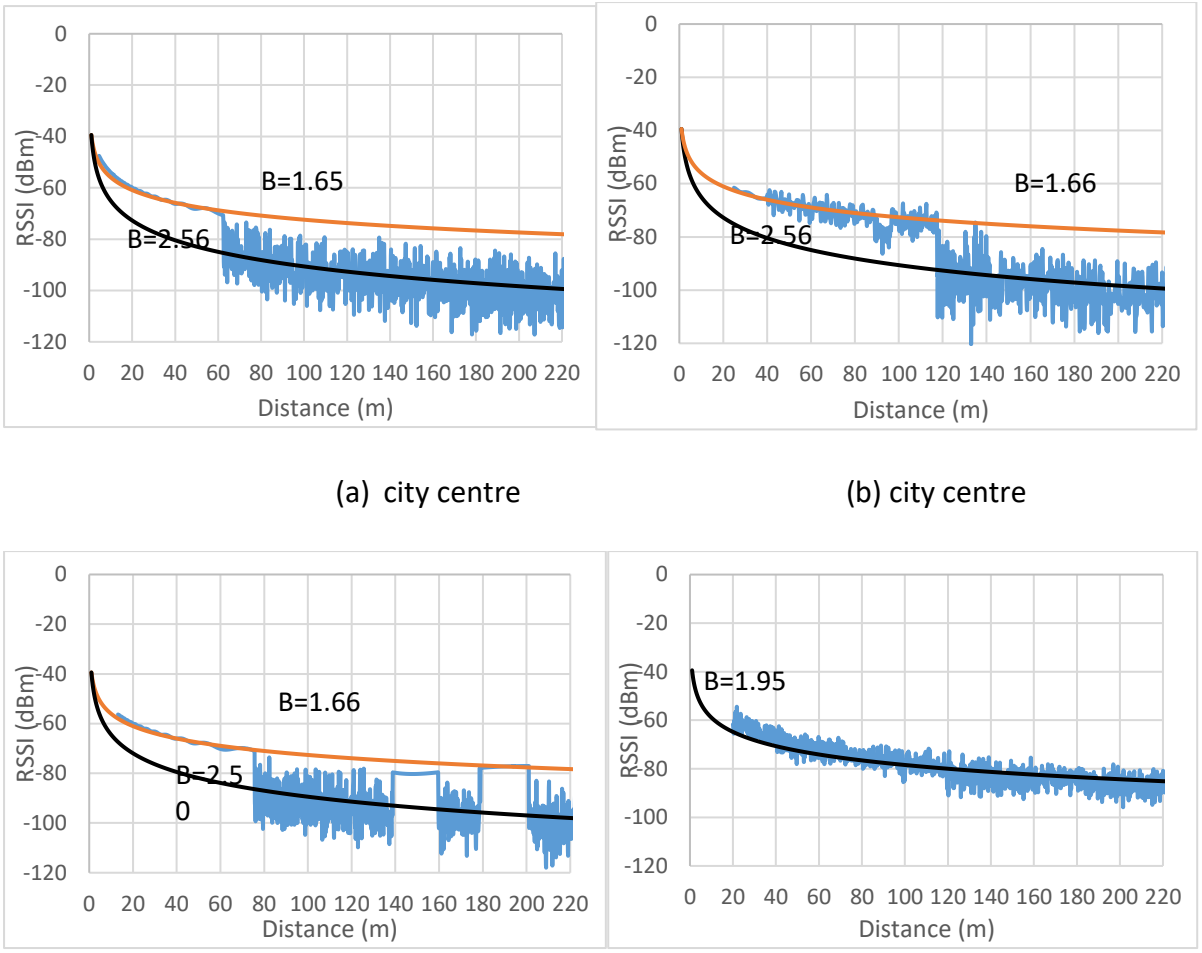
To generate data for the evaluation we used models of real locations taken from Open Street Map (OSM) that include representations of building geometry, foliage, and road networks. In particular, we selected locations in and around Newcastle, UK, that represents distinct types of environment. The locations are (a) a city centre area with many tall buildings, (b) an inter-city highway, and (c) a suburban area where buildings are of moderate size and density. We then used SUMO, which is a widely used road traffic simulation tool, to generate traces of vehicles' trajectories in these locations with varying range of vehicle densities as shown in Table 6.1. The traces are then converted into floating car data (FCD) format and used as input to GEMV^2, which runs the simulation. During each run, GEMV^2 determines the link type among all the transmitter-receiver pairs for the current time step and then calculates the RSSI values. Other parameters used in the simulation are shown in Table 6.1.

Table 0-1: Simulation Settings

Parameters	**Value**
Vehicle density (vehicles/km^2)	City centre: 2-300 Suburban: 2-200 Highway: 2-100
Message frequency	10Hz
Operating frequency	5.9GHz
SUMO simulation time	3600s
Channel Bandwidth	10MHz
Vehicle antenna type	Omni-directional
Energy detection threshold	-77dBm
Transmission power	32dBm

6.4.2 Determining the parameter values

The RSSI data generated from each of the three locations were plotted against the distance between sending and receiving vehicles. A total of about 100 simulation runs were conducted. Fig. 6.3 presents examples of such plots generated in the following scenarios: (a) city centre with traffic density = 300 vehicles/km^2, (b) city centre with traffic density = 250 vehicles/km^2, (c) highway with traffic density = 100 vehicles/km^2, and (d) suburban with traffic density = 200 vehicles/km^2. It is apparent that the plot can be divided into distinct segments, which were found to correspond to LoS conditions (characterised by absence of noise-like variability), obstruction by traffic, obstruction by buildings, etc. The curves overlaid on Fig 6.3 were generated with B values for different conditions reported in real measurements conducted in [129] [130] [131]: B= 1.65, 1.66 corresponds to LoS condition in both city centre and highway environments, B=1.95 corresponds to NLoS condition in suburban environment, B=2.50 corresponds to NLoS condition in highway environment, and B= 2.56 corresponds to NLoS condition in city centre environment. This is to establish a link between the simple theoretical propagation model with more complex results from the simulation work. In addition, it serves as means to check whether the data generated from the simulations is in line with the results obtained based on real measurements and other similar simulation studies.



(a) city centre (b) city centre

(c) highway environment (d) suburban

Figure 0.3: RSSI against distance for different scenarios

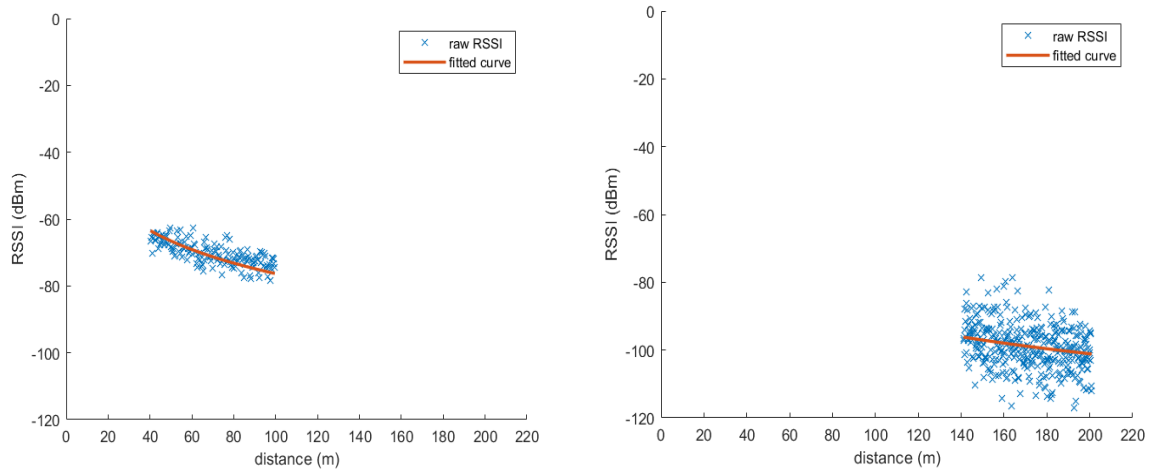


Figure 0.4: Example of curve fitting using (7.1) on RSSI trace plots

Each RSSI trace was divided into segments by eye. Curves of the form in equation (6.1) were fitted independently to each segment as illustrated in Fig. 6.4, to obtain values for B, with A being held fixed at a value determined from typical vehicle characteristics, and d_0 =1m. The root mean square deviation of RSSI points from the fitted curves was then calculated to obtain σ values for each segment. It may be seen from Fig. 6.5 that the segments appear to be distributed about a straight line in (B, σ) space. This looks quite similar to the earlier scattered plot in Fig. 6.2, a straight line of positive slope can also be fitted in it. However, there is more variations in Fig. 6.2. than in Fig. 6.5. The standard deviation of the random variable is greater for the real measurements than it is for the simulation results. This is possibly due to methodological differences (e.g. hardware setup) as well as the assumptions made in the real measurement studies. In addition, the presence of other static objects e.g. lamp post, street signs, traffic lights, etc., in the real locations obstructs the signals and causes multipath effects, which could have impact on σ. Not all of these objects are accounted for in the simulation framework due to their large numbers, varying shapes and sizes, which increases the computational complexity. Moreover, information on some of these objects is currently not available in the geographical databases. Given all the uncertainties involved and the limited precision of the geographical database, the simulation results and real measurements published in the literature are in very good agreement with each other. We, therefore, adopted the parameterisation of equation (6.3) with B_0 =1.4 being the least value of B for any segment, and k=3.89 and γ_0 =1.00 being determined from a least-squares fit through the points of Fig 6.5.

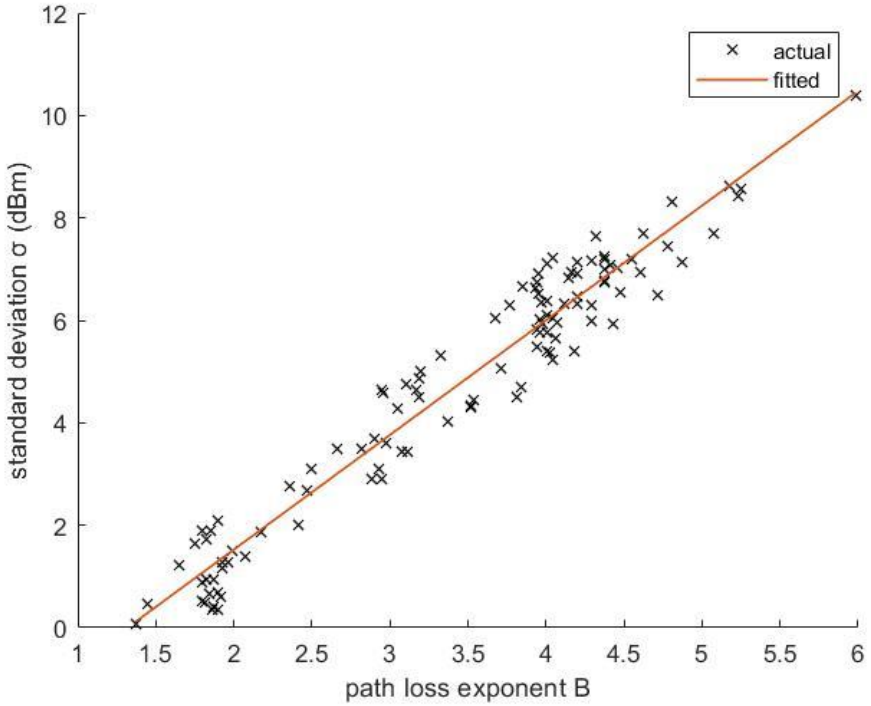


Figure 0.5: Least Square fit of (B, σ)

From equation (6.5) we see that the ratio of the uncertainty on the distance estimate $\bar{\sigma}_d$ to the distance estimate itself $\hat{d}$ is dependent on σ. For B_0=1.4 and using k=3.89 and γ_0=1.00 from the least squares fit, the ratio is about 0.14 for σ=1dBm, 0.38 for σ=5dBm and 0.49 for σ=10dBm. If we consider a 3-sigma criterion for consistency, then for σ=1dBm, the discrepancy between claimed distance and true distance would need to be greater than 42% of the true distance to be judged to be lying. For σ=3.75dBm, the required discrepancy is about the same size as the distance itself. As the main threat comes from vehicles claiming to be closer than they really are, then the proposed technique is only useful for σ<3dBm. Reducing the inconsistency criterion extends the applicable σ range, however, albeit at the cost of increased false positives.

6.4.3 Applying the filtering algorithms

To use equation (6.4) to estimate the separation distance between the receiver and a given sender, and equation (6.5) to estimate the uncertainty on this value, a receiving vehicle must extract mean RSSI values and the corresponding standard deviations from a 'noisy' sequence of RSSI measurements. Furthermore, these values must be updated continuously. The two filtering algorithms described above were tried for this purpose. The filtering algorithms were

reset at the boundaries between segments of the RSSI plots, which were detected as a rapid change in the rate of change of the mean RSSI. The standard deviation was obtained from the covariance matrix in the case of the Kalman filter and a moving window standard deviation in the case of Savistzky-Golay filter. Fig. 6.6 shows a sample RSSI trace overlaid with the mean RSSI, mean RS–I + σ, and mean RSSI - σ curves extracted using the Savistzky-Golay filter. As may be seen the algorithms are reasonably effective at tracking the mean RSSI value and the corresponding standard deviation boundary curves covering majority of the RSSI variability. The mean RSSI and σ will serve as input to the RMCCS method for the distance estimation.

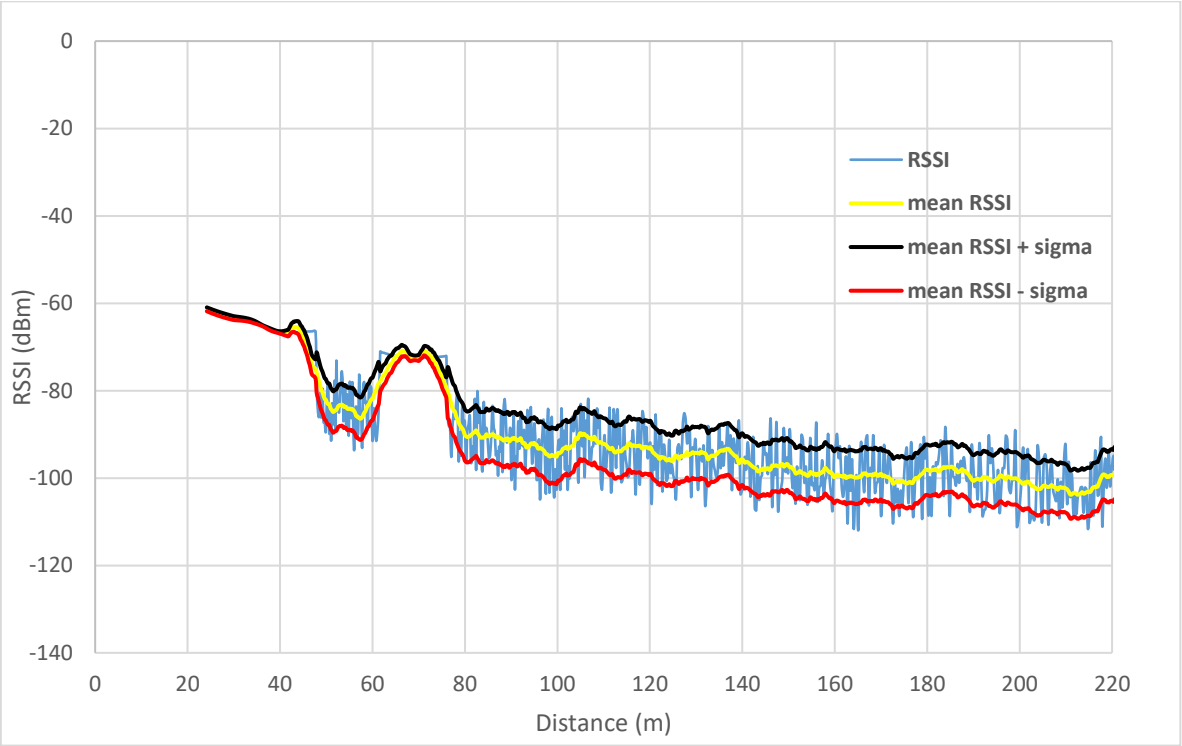


Figure 0.6: Mean RSSI and standard deviation data generated from the filtering algorithm

6.4.4 Evaluating the results

In this section, we evaluate the performance of the RMCCS using some of the RSSI traces. Fig. 6.7 shows the estimated distance against the true distance for one sample trace. At a distance of less than 50m, the estimated distance is similar to the true distance. However, as the separation distance increases, the margin of error between them also increases. The

estimation error, defined as $\frac{|\hat{d}-d_r|}{d_r}$, was computed and found to be less than 25% everywhere and below about 12% for separation distances less than 50m. The overall average estimation error was found to be 7.5% for distances up to 250m. This result is similar to those obtained in the other sample traces.

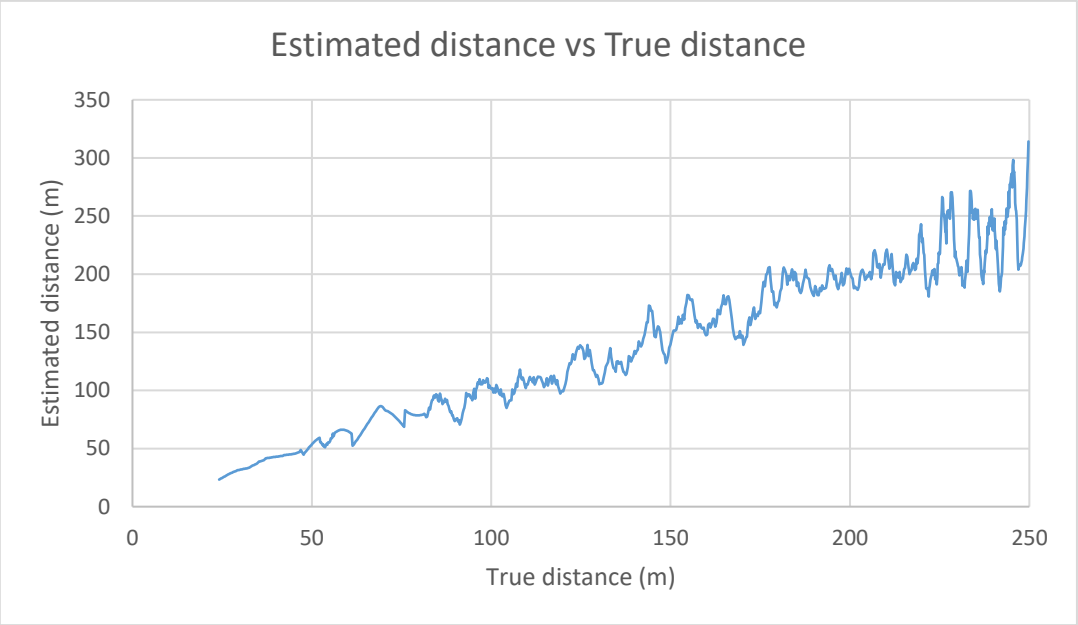


Figure 0.7: Estimated distance vs True distance

To assess the probability of true negatives, TN, (and false positives, FP) for different inconsistency criteria, we calculated the proportion of data points in the sample trace for which the absolute difference between the true and estimated distance exceeds various multiples of $\bar{\sigma}_d$. To assess the probability of true positives, TP, (and false negatives, FN), we used threat scenario in which a static malicious vehicle simulates a Sybil vehicle following the target vehicle at various fixed distances. TP is calculated as the proportion of data points in the sample trace for which the difference between the (constant) reported distance and the estimated distance exceeds various multiples of $\bar{\sigma}_d$. The results are shown for various following distances and inconsistency criteria in Figure 6.8. To get an overall assessment of TP for a given inconsistency criterion, we took the average over the various following distances up to 250m. Because it is reasonable to suppose that detecting fictitious vehicles that are faraway is less important than detecting ones that are nearby, we also calculated the averages over following distances up to 100m. Having obtained TN and TP values for a range of inconsistency criteria we calculated accuracy values, which is expressed as:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} = \frac{TP+TN}{2} \quad (6.6)$$

The results are shown in Table 6.2. As can be seen, an inconsistency criterion of $\frac{|\hat{d}-d_r|}{\bar{\sigma}_d} > 1$ appears to give the best accuracy of approximately 90% for distances up to 100m and about 83% for longer distance up to 250m.

Table 0-2: TP, TN and Accuracy values for the evaluation scenario of RMCCS method for three inconsistency criteria: |d̄ - dr|/ σ̄d > N

Metric	**Distance(m)**	**N = 1**	**N = 2**	**N = 3**
TN	up to 250m	0.9551	0.9708	0.9809
TP	up to 250	0.7195	0.4344	0.2051
	up to 100	0.8441	0.4834	0.1119
Accuracy	up to 250	0.8373	0.7026	0.59303
	up to 100	0.8996	0.7271	0.54641

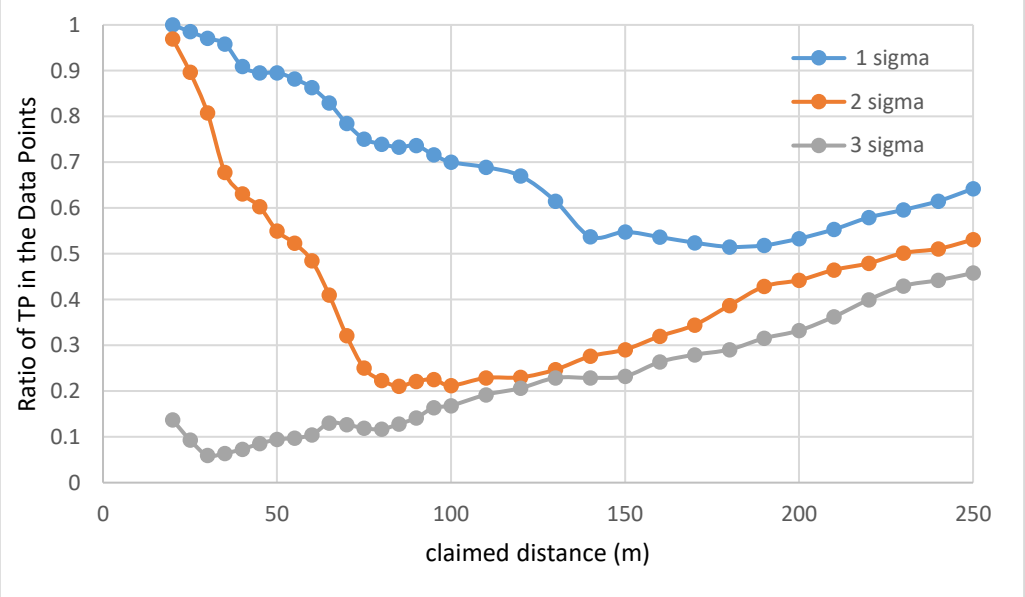


Figure 0.8: True Positives in the Data Points for the Evaluation Scenario

6.5 Discussion

In this section, we summarise the main features of RMCSS and its advantages over other RSSI-based distance estimation methods. We then discuss how it would support the PV+TESLA

model proposed in Chapter 5. Finally, we highlight some further investigations that can be carried out to confirm the reliability of the RMCSS method.

6.5.1 Summary of RMCCS Method

By proposing a linear relationship between the path loss exponent and the standard deviation of the noise component in the LDPLM model, the RMCCS method enables a receiving vehicle to independently estimate the distance to a given message sender and the uncertainty associated with this estimate. This is achieved based on the strength and variability of RSSI measurements without prior knowledge of environmental conditions such as the current traffic density, building density and foliage present, between and/or around the sender and the receiver. The assumption of this linear relationship is justified by empirical evidence obtained from a realistic simulation conducted in different environments under varying conditions, and also from data extracted from real measurements reported in the literature. The estimated distance and associated uncertainty provide a means to judge whether the sender is reporting false position information in its CAM message. We use the ratio of the magnitude of the difference between reported and estimated distances to the uncertainty on the estimate as a measure of inconsistency. The sender is judged to be reporting false position information if the inconsistency is greater than a threshold value. Lowering the threshold tends to increase true positives, but reduce true negatives. The threshold can be varied to obtain an optimal value that maximises accuracy (which is proportional to the sum of true positives and true negatives).

Our RMCCS method does not need the support of neighbouring vehicles or any nearby infrastructure to function. In contrast, the related works described in 6.2 including the works of [134] require collaboration among neighbouring vehicles to estimate the distance of a target vehicle whereas in RMCCS the estimation algorithm is purely local. The accuracy of [134] degrades as the number of vehicles reporting their individual estimated distances to the target decreases and also depends on the correctness of the reported information. Such approaches are unreliable because vehicles may fail to collaborate and messages to the observer from its neighbours may be lost in transmission. Furthermore, the same, fixed path loss exponent is used by all collaborating vehicles, whereas, as we have clearly seen, its value depends on the obstacles on or near the transmission path between any given sender and receiver. In contrast, RMCCS is able to extract a dynamic value for the path loss exponent

from the RSSI data using the linear relationship. Similarly, in [135], cooperation is needed, but in this case, among nearby RSUs to share the distance estimates. Again, a fixed path loss exponent is used to estimate the distance to the target vehicle. A further disadvantage is that it is unrealistic to assume that RSUs will be available in all locations. Collaborative methods also add message traffic, which increases bandwidth usage, and introduces delays that could affect safety applications performance.

In terms of evaluation, the previous works mostly assessed their RSSI-based distance estimation methods using simulators such as NS-2 that employ simple statistical propagation models. Furthermore, they are applied to all links between communicating pairs indiscriminately, without taking into account specific link conditions i.e. whether a link has a clear LoS path, or it is obstructed by nearby static and/or moving objects. Such models are unable to accurately represent link-level V2V communication and provide the expected RSSI measurements. In contrast, our RMCCS method was evaluated using GEMV^2, which accounts for RSSI variation due to obstructions from surrounding objects that are proven to affect the signal propagation. This makes GEMV^2 a more accurate and realistic testbed for signal power measurements in V2V. Moreover, studies in [146] [147] shows a significant difference in received power when comparing the performance of GEMV^2 and the propagation models built into NS-2. This indicates that performance estimates obtained using NS-2 are questionable and that when the previous work is evaluated with a more realistic simulation environment, it is highly likely the performance will reduce.

6.5.2 Use of RMCCS in the PV+TESLA Model

The results obtained in 6.4.4. show that the performance of the RMCCS method is promising. Within the separation distance of less than 100m between the sender and the receiver, the accuracy of RMCCS is at 90% level, which indicates that it could be adequate to form part of the PV+TESLA model proposed in Chapter 5. In this model, PV's vulnerability to impersonation attacks is mitigated by using TESLA to verify selected messages. As described in 5.3.1.2, one of the factors that a decision function/algorithm in the PV+TESLA model can employ to decide whether or not to apply TESLA on a given received message is when a suspicious event is detected. The RMCCS method fits into this category of the PV+TESLA model's decision-making as its operation targets a signature of the PV's vulnerability. Part of the signature of an attack exploiting the PV's vulnerability is that the actual position of the vehicle sending a message is

different from the position reported in the message. So, the RMCCS method can give evidence of a suspicious event that can serve as input to the decision function, if the estimated distance is found to be inconsistent with the separation distance computed from the sender's position claimed in the message. Thus, the measure of inconsistency can act as one trigger for invoking TESLA on messages found to be suspicious, which reduces the risk of processing false information in the safety applications. In addition, the consistency evidence provided by the RMCCS method further confirms the genuineness of messages verified with PV and increases the confidence of vehicles on processing the message content. This means that TESLA will be invoke less often, which in turn reduces the effective authentication delay. Therefore, the use of RMCCS method would greatly improve the success of the PV+TESLA model. In the next section, we integrate PV+TESLA model with the RMCCS and evaluate their performance.

6.6 Integration of PV+TESLA and RMCCS Algorithm

In this sub-section, we illustrate the integration of the PV+TESLA model with the RMCCS technique. Fig. 6.9 presents a Unified Modelling Language (UML) diagram depicting the sequence of activities/events in each layer of the ITS stack and the message flow between them. As safety messages arrives at the access layer, the mean RSSI and standard deviation required for RMCCS computation are extracted at this layer and attached to the received message. As mentioned in section 5.2.3.2, the network and transport layer handle cryptographic operations; thus, it is assumed that both PV and TESLA verifications are performed in this layer. Meanwhile, RMCCS computations and plausibility checks are done at the facilities layer, where message content is interpreted. Evidence from both the RMCCS scheme and a plausibility checking function is used to decide whether a message verified with PV should be accepted and forwarded to the application layer or wait for its corresponding TESLA key to verify it, after which it is then forwarded to the application layer.

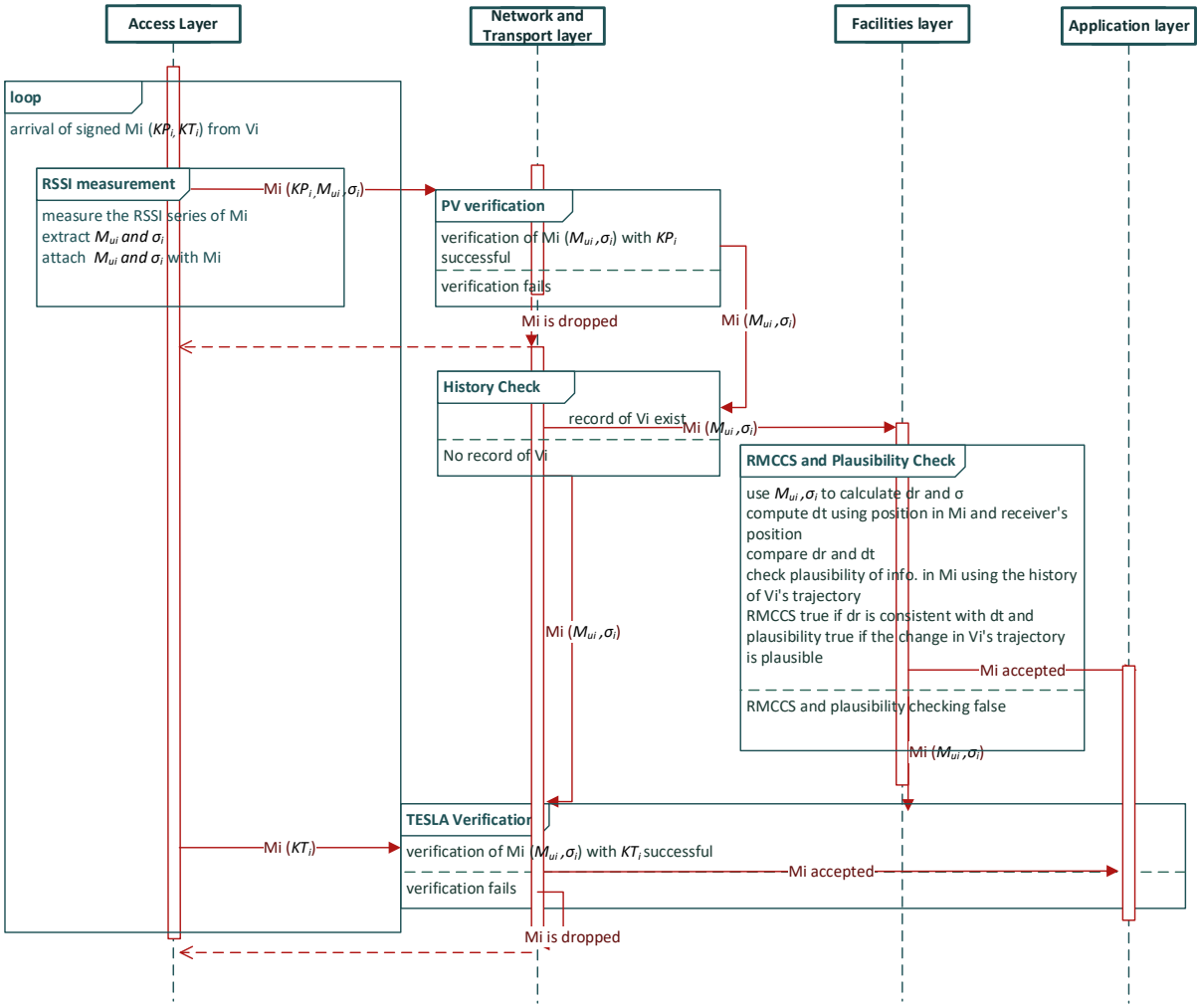


Figure 0.9: Integration of PV+TESLA and RMCCS Technique

Algorithm 6-1 demonstrates an implementation of the PV+TESLA and RMCCS integration.

Algorithm 6-1

parameters: *signed msg_i with KP_i (PV key) and KT_i (TESLA Key) from V_j (sending vehicle),M_{ui} (mean RSSI value extracted from RSSI measurements made during message reception), σ_i (standard deviation of RSSI measurements made during message reception), $\hat{d}$ (estimated distance of V_j from receiving vehicle), $\bar{\sigma}_d$ (uncertainty on $\hat{d}$) and d_r (V_j's distance from receiving vehicle calculated using receiving vehicle's known position and V_j's position reported in msg_i)*

Begin

On arrival of signed msg_i from V_j
Physical layer processing
1 measure the RSSI series of msg_i
2 extract M_{ui} and σ_i
3 attach M_{ui} and σ_i with msg_i and pass them up to the network and transport layer
Network and Transport layer processing
On receipt of signed msg_i from physical layer

4 verify msg_i with KP_i
5 if verification of msg_i fails
6 then drop msg_i and exit Network and Transport layer processing
7 If no record of V_j exists in the receiving vehicle
8 then
9 store msg_i and wait for the arrival of KT_i
10 on arrival of KT_i, verify msg_i using KT_i
11 if verification of msg_i is successful
12 then forward msg_i to the application layer
13 else drop msg_i
14 exit network and transport layer processing
16 forward msg_i to Facilities layer

Facilities layer processing
17 execute decision function (RMCCS_check and plausibility_check) on msg_i
18 calculate $\hat{d}$ distance of V_j and $\bar{\sigma}_d$ uncertainty in $\hat{d}$
19 if $\hat{d}$ is consistent with calculated distance d_r in msg_i
20 then RMCCS_check returns true
21 else RMCCS_check returns false
22 check plausibility of (msg_i) with V_j's trajectory history
23 if information in msg_i is consistent with V_j's trajectory history
24 then plausibility_check returns true
25 else plausibility_check returns false
26 If RMCCS_check is true and plausibility_check is true
27 then forward msg_i to application layer
28 exit facilities layer processing
29 else return to 9, exit facilities layer

End

6.6.1 Evaluation and Discussion

A simulation was conducted to evaluate and assess the performance of the PV+TESLA model together with RMCCS technique. Algorithm 6-1 was implemented in Python and then added to the security module described in 3.2.3. Similar to the simulations conducted in previous chapters, SUMO is used to generate random trips and the mobility traces of vehicles for each trip are then used as input to NS3, which simulates message transmission between vehicles. Each vehicle sends broadcast messages every 100ms. Each message is signed with two separate keys (PV and TESLA keys). The security module handles each incoming message as described in the algorithm above, from RMCCS computation to plausibility check, as well as cryptographic verifications stages. For the plausibility check, we adopt the misbehaviour detection techniques used in the work of [148] [149]. It uses the speed check and distance

moved verifier to determine whether the position claimed in the current message is plausible. The speed check decides plausibility based on how the reported velocity relates to the velocity implied by the position and time differences between the current and the previous message from the same sender, and the reported speed in the current message. If the deviation is less than a threshold value, Ω, it implies that the change in position is plausible. The distance moved verifier checks whether the vehicle moved a minimum distance relative to its previous position and speed information, and if this distance is too small, the new position is considered not plausible. The values of Ω and that of the minimum distance are obtained from [148] [149]. A table is set up for each vehicle to keep track of the most recent message and corresponding TESLA key from a given sender. A total of 58 experiments were generated during the simulation, with each run having more than 200 vehicles. In each experiment, the vehicle that received the highest number of messages is selected and its output is used for evaluation. The following parameters: the total number of messages received, messages verified using PV, messages verified using TESLA and the total message verification time, were recorded at the end of each trip. Table 6-3 shows the results obtained.

The percentage of messages verified with PV is computed by dividing number of messages that passed PV (including RMCCS and plausibility check) verification by the total number of messages received. In all the experiments, it can be seen that between 50% to 60% of messages received by the receiver vehicle were verified with PV. This means that the RMCCS method and the plausibility checking function were able to confirm the correctness of these messages. The separation distances computed from the senders' positions reported in these messages and the receiver's known position are consistent with the distance estimated using the RMCCS function. On the other hand, about 40% of total messages received were selected to wait for verification with their corresponding TESLA key in order to avoid the risk of PV's vulnerability. This includes messages received for the first time from an unknown sender and those where the RMCCS function found inconsistencies between the estimated distance and the true distance as well as those in which the change in position does not appear to be plausible.

Table 0-3: PV+TESLA and RMCCS Results

Trips	Messages passed PV verification	Messages initially suspicious and had to wait for TESLA verification	Messages passed TESLA verification	Total number of messages received	percentage of messages verified with PV
1	13080	12333	12333	25413	51.470
2	3783	3672	3672	7455	50.744
3	16084	15524	15524	31608	50.886
4	17945	17243	17243	35188	50.997
5	21214	20579	20579	41793	50.760
6	24411	23795	23795	48206	50.639
7	26813	26156	26156	52969	50.620
8	28637	27932	27932	56569	50.623
9	2816	1838	1838	4654	60.507
10	3556	2518	2518	6074	58.545
11	32310	31719	31719	64029	50.462
12	8120	7226	7226	15346	52.913
13	35050	34464	34464	69514	50.421
14	34687	33956	33956	68643	50.532
15	37170	36477	36477	73647	50.470
16	40264	39478	39478	79742	50.493
17	47522	46827	46827	94349	50.368
18	13080	12276	12276	25356	51.585
19	17718	16967	16967	34685	51.083
20	12492	12196	12196	24688	50.599
21	7953	7602	7602	15555	51.128
22	10777	10384	10384	21161	50.929
23	9378	8814	8814	18192	51.550
24	15198	14642	14642	29840	50.932
25	18447	17920	17920	36367	50.725
26	5329	5053	5053	10382	51.329
27	14151	13672	13672	27823	50.861
28	2248	2132	2132	4380	51.324
29	51539	50889	50889	102428	50.317
30	48175	47501	47501	95676	50.352
31	54107	53387	53387	107494	50.335
32	8375	7750	7750	16125	51.938
33	9739	9084	9084	18823	51.740
34	6581	6477	6477	13058	50.398
35	9739	9084	9084	18823	51.740
36	1340	1181	1181	2521	53.154
37	9578	9004	9004	18582	51.545
38	13168	12240	12240	25408	51.826
39	10088	9083	9083	19171	52.621
40	1536	1377	1377	2913	52.729

41	8375	7750	7750	16125	51.938
42	1536	1377	1377	2913	52.729
43	16485	15781	15781	32266	51.091
44	16507	15792	15792	32299	51.107
45	9739	9084	9084	18823	51.740
46	2035	1876	1876	3911	52.033
47	14817	14354	14354	29171	50.794
48	18422	17827	17827	36249	50.821
49	10108	9477	9477	19585	51.611
50	16930	16530	16530	33460	50.598
51	19170	18457	18457	37627	50.947
52	52038	51738	51738	103776	50.145
53	54107	53788	53788	107895	50.148
54	6124	6043	6043	12167	50.333
55	9227	9107	9107	18334	50.327
56	11685	11464	11464	23149	50.477
57	15001	14772	14772	29773	50.385
58	16084	15524	15524	31608	50.886

Next, we compare our PV+TESLA approach with an approach found in [62] [63] [64] and [65]. These works adopt a prediction-based approach (PBA) to reduce or eliminate TESLA's fixed minimum latency issue in order to achieved timely verification of messages in V2V. The PV+TESLA and PBA approaches share similar properties, which makes them comparable. First, they both rely on symmetric cryptographic functions (hashes and MAC) and the standard TESLA. Second, in both PBA and our PV+TESLA approach, some messages are verified immediately when certain conditions are met, while others are subject to standard TESLA verification. Third, they are both standalone schemes such that each receiver vehicle operates independently without participation of other vehicles or nearby RSUs for its decision making. Recall that in PBA schemes, vehicles predict their own future position information given their past trajectory and then generate what is referred to as the prediction outcome, which is sent in advance as part of every message broadcast. A given message M_i carries the prediction outcome of message M_{i+1} which is then buffered. Upon arrival of M_{i+1}, the receiver reconstructs the prediction outcome using the position information reported in the message and then compares it with the prediction outcome stored in the memory. If the two match, the receiver accepts M_{i+1} as genuine and verifies it immediately. Otherwise, M_{i+1} waits for TESLA verification when the corresponding key is disclosed. If a message is lost or dropped in transmission, the prediction outcome of the following message is also lost, hence the

following message cannot be immediately verified. This means that the effectiveness of PBA in achieving instant message verification depends on the successful reception of two consecutive messages.

We conducted simulations in NS3 to implement PV+TESLA, PBA, and the standard TESLA under various traffic scenarios. To ensure comparability with the results obtained by the authors of PBA, we adopted similar traffic scenarios and simulation settings. These settings include a traffic density ranging from 2 to 100 vehicles within communication range, a TESLA authentication delay set to 100ms, and a 6Mbps channel bandwidth.

In the case of PV+TESLA, the size of a safety message is set at 80 Bytes. For PBA, the message size is 160 Bytes, as each message includes the prediction outcome and additional information called off-path nodes. These off-path nodes are used in reconstructing the prediction outcome at the receiver side. For the standard TESLA, the message size remains consistent with that used in Chapter 3.

To assess their performances, we measured the average message verification delay for each approach under different traffic scenarios during the experiments. The message verification time was determined by measuring the interval between a message arriving and the completion of its verification. This interval includes:

- For PV+TESLA: MAC processing times, RMCCS (Reactive MAC with Checkpoint and Storage) processing times, and plausibility check function processing times. Additionally, TESLA's authentication delay is considered for messages that the receiver decides to verify with the TESLA key.
- For PBA: Prediction outcome reconstruction time, MAC processing time, and TESLA's authentication delay for messages in which the corresponding prediction outcome is lost or no match is found between the buffered prediction outcome and the reconstructed prediction.
- For standard TESLA: MAC processing time and TESLA's authentication delay.

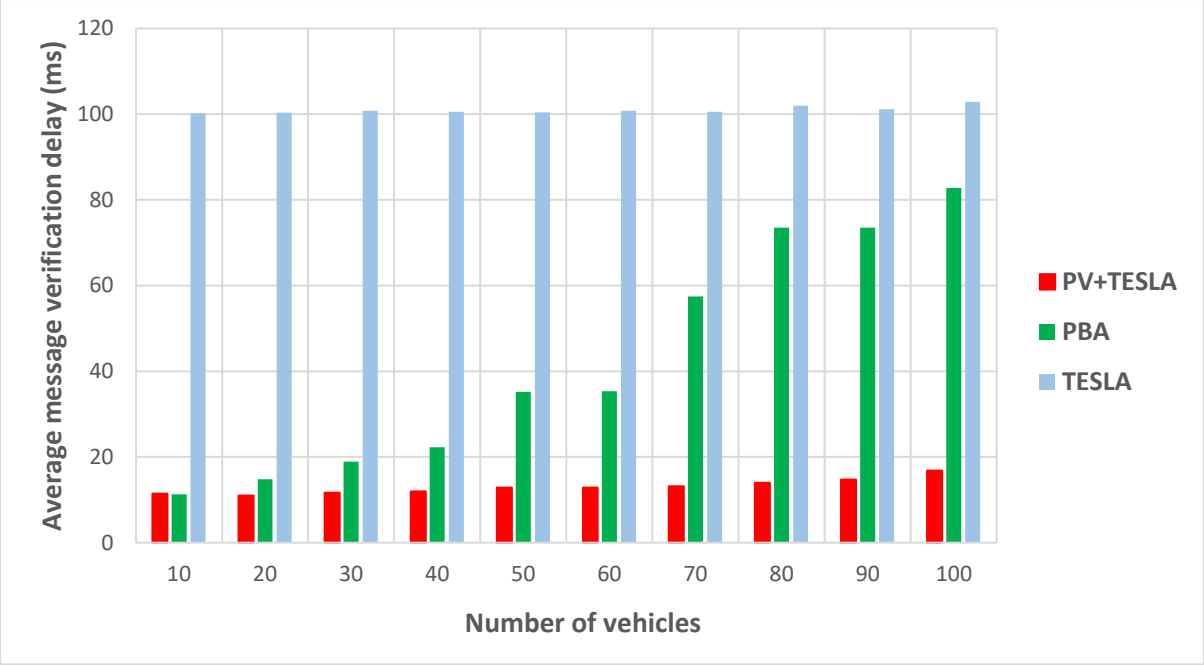


Figure 0.10: PV+TESLA vs PBA

Figure 6.10 illustrates the average message verification delay against the number of vehicles for PV+TESLA, PBA, and TESLA cases. As expected, the standard TESLA maintains a consistent average message verification delay within the examined range of traffic, primarily due to the fixed latency of 100ms. In the case of PBA, a lower number of senders within the communication range of the receiver results in a low average message verification delay. This is attributed to reduced congestion on the transmission channel, allowing messages to be more likely received and subsequently verified based on previously broadcast prediction outcomes. However, with an increasing number of senders, channel contention and transmission interferences grow, leading to higher message loss or drop rates. As more messages are lost in transmission (implying the loss of prediction outcomes), TESLA is invoked more frequently to verify successfully received messages, thereby causing an increase in the average message verification delay. This delay almost approaches that of the standard TESLA, as indicated in Fig 6.10. Notably, when the prediction outcome is lost in transmission, the subsequent message cannot be verified immediately and must be verified with standard TESLA. Thus, the performance of PBA depends on the reliability of the transmission channel, which is unpredictable.

On the other hand, PV+TESLA maintains an average message verification delay of less than 20ms throughout the range of traffic scenarios examined. The average message delay

increases slightly with increasing traffic density because when the receiver decides to verify a given message with standard TESLA, the message is stored until the reception of the TESLA key from the same sender. This contributes to the increased average message verification delay. Importantly, PV+TESLA remains independent of the transmission channel condition, whether there is a high message loss rate or not, as the verification of a given message does not depend on the successful reception of the preceding message from the same sender. Arriving messages are processed and verified individually without relying on the reception of another piece of information in a separate broadcast from the same sender. In contrast, the performance of PBA degrades in environments with high message losses. Additionally, the PBA scheme incurs higher communication overhead than PV+TESLA and the standard TESLA approaches due to its larger message size, approximately a 100% increase compared to the PV+TESLA message size. This makes PV+TESLA messages have a higher chance of being transmitted even in high traffic conditions, as they occupy fewer radio resource blocks (i.e., transmission blocks) than PBA messages.

The results demonstrate that the PV+TESLA model, combined with the RMCCS technique and plausibility check, reduces the authentication delay by about 85% compared to standard TESLA, while minimizing the risk of accepting false messages with the support of RMCCS and the plausibility check function.

6.6.2 Evaluation of Attacker Activity in the PV Model

In this section, we assess the significance of the risk associated with PV's vulnerability by modelling and simulating the presence of an attacker in the PV+TESLA approach. We consider a scenario in which an attacker attempts to exploit the PV's vulnerability as described in 5.2.3. We assume the SM case, wherein the attacker uses a vehicle to impersonate another vehicle in the simulation area and uses its disclosed keys to broadcast a sequence of fake messages describing a false trajectory.

During each experiment, an attacker vehicle Va is selected, and then the vehicle to impersonate Vg and the receiver Vr (victim) are randomly chosen from those in appropriate positions relative to Va, as described in 5.2.3.1. Va maintains a fixed distance between itself and the selected impersonated vehicle Vg. The experiment is repeated with the fixed distance

between Va and Vg increasing in steps of 10m from 10m to 200m. The favourable configuration is maintained until the end of each experiment. The attacker vehicle sends its fake messages every 100ms, as would a genuine vehicle. The fake messages are tagged at the sender's side (Va) in order to identify them at the receiver's end. We assume that the attacker is reporting a false trajectory (position, speed) that is different from the impersonated vehicle's real trajectory in order to mislead the receiver and other vehicles within its vicinity. The attacker is limited to reusing the same impersonated vehicle's key in every run. We consider a dual-carriageway road network with three lanes, and traffic is moving in both directions. Vehicles are introduced into the simulation area following the approach described in Chapter 3. The simulations were repeated 20 times using different mobility traces obtained from SUMO, each having an average of 200 vehicles in the simulation area.

In each experiment, we measure and compute the mean of the following:

- FM_T: the total number of fake messages sent by the attacker,
- FM_R: the number of fake messages detected by the RMCCS technique and plausibility check function,
- FM_K: the number of fake messages detected due to the reception of two messages from the same sender signed with the same key, and
- GM: the number of genuine messages received from Vg.

Then, we calculate the following:

- TP (true positive) denotes the total number of fake messages that are correctly detected as fake ones.
- FN (false negative) denotes the number of fake messages that are verified and forwarded to the safety applications, i.e. fake messages accepted as genuine ones.
- TN (true negative) denotes the number of genuine messages (i.e., received from Vg) accepted as genuine ones.
- FP (false positive) denotes the number of genuine messages that are incorrectly classified as fake ones.
- True Positive Rate (TPR): represents the ratio of correctly classified fake messages to the total number of fake messages sent by the attacker, which is expressed as;

$$TPR = \frac{TP}{TP + FN}$$

- Precision: the proportion of the messages classified as fake that are actually fake expressed as:

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

Table 0-4: PV+TESLA Attacker Scenario Results

Va-Vg Distance (m)	FM_T	FM_R	FM_K	TP	FN	FP
10	5309	1703	3288	4991	280	35
20	6178	2023	3538	5561	304	64
30	7857	2508	4677	7185	470	80
40	8282	2638	4921	7559	502	81
50	10365	2325	7026	9351	657	117
60	11065	2911	7155	10066	748	176
70	16445	5033	9502	14535	1247	204
80	11910	3730	7001	10731	879	213
90	15054	5071	8318	13389	1125	247
100	16946	7448	7659	15107	1353	299
110	19921	10465	7257	17722	1687	319
120	21700	11088	7849	19037	1663	345
130	20397	13686	4275	17961	1736	381
140	18737	7820	7750	15570	1525	425
150	18693	8767	6696	15463	1649	468
160	22710	14703	4503	19206	1976	492
170	23585	12683	5165	17848	1982	626
180	64702	31490	19605	51095	6231	876
190	65534	30001	15955	45956	6325	1032
200	76261	28822	25066	52010	7102	1094

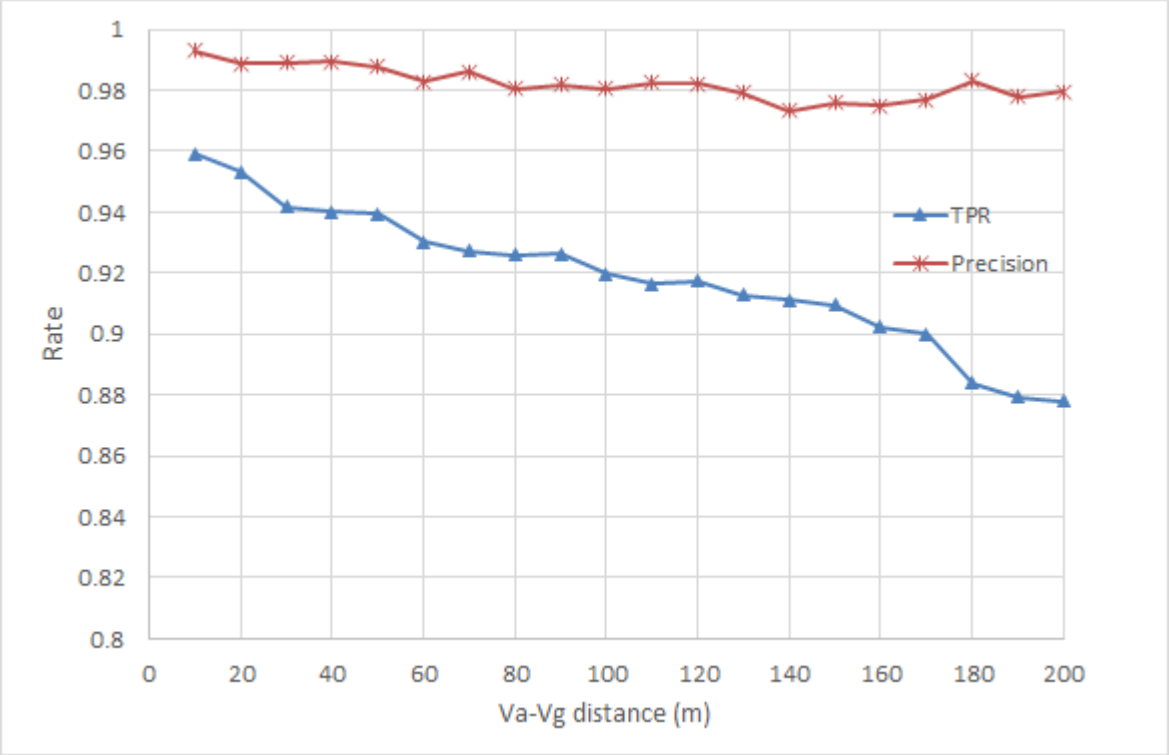


Figure 0.11:Attacker Scenario Results

Fig 6.11 shows how the mean TPR and Precision vary with the distance between the impersonated vehicle and attacker vehicle. It can be seen that the TPR curves shows a slight decrease with increasing Va-Vg separation distance. This is because the closer Va is to Vg, the higher the chances of Vr receiving Vg's genuine messages as well as the fake messages. For distances less than 100m, Vr is able to receive Vg's genuine messages and hence, the fake messages signed with the same key are discarded. This leads to FM_K contributing an average of 65% to the total TP, while FM_R contributes 35%. However, beyond 100m separation distance between Va and Vg, the arrival rate of genuine messages at Vr reduces due to propagation losses. Consequently, FM_K decreases to an average of 38%, and FM_R increases to an average of 62%. Overall, an average of 92% TPR was achieved in the experiments.

The precision curve indicates that more than 97% of the detected fake message are indeed fake in all the experiments. This demonstrates the effectiveness of the RMCCS, the plausibility check function, and detection due to message clash in mitigating the risk associated with the PV's vulnerability. In all the experiments, no more than 9% of the fake messages were

accepted, and fewer than 3% of the genuine messages arriving at the receiver are misclassified as fake ones, possibly due to error in the RMCCS and plausibility check function.

The results highlight the limited opportunity attackers have in exploiting PV's vulnerability in practice. The receiver is able to receive some of the genuine messages from the impersonated vehicle due to the unpredictability of broadcast communication, wherein the transmission range is not well-defined. An attacker has no control over it and cannot prevent the reception of genuine messages in this context. There is no guarantee that vehicles located at a far distance would not receive a given message, or that those at a near distance are guaranteed reception. Thus, finding an optimal configuration to fully exploit PV's vulnerability could be challenging.

Additionally, during the experiments, it was observed that not all of the impersonated vehicle's messages were successfully received by the attacker due to the unreliability of the transmission channel. This was evident from the sequence of the disclosed PV keys reused by the attacker in its sequence of fake messages.

6.7 Formal Analysis of PV+TESLA

The security of PV+TESLA is formally analysed using the well-known Burrows-Abadi-Needham (BAN) logic, which is a tool for analysing authentication schemes in order to prove their robustness or hidden weakness.

6.7.1 BAN Logic Notations:

- P |≡ X : P believes X.
- P ◁ X : P sees X; P has received X.
- P | ~ X : P once said X or P has sent message X.
- P ⇒ X : P controls X or P has jurisdiction over X.
- # (X) : X is fresh.
- P $\overset{k}{\leftrightarrow}$ Q : P and Q can use the shared key K to communicate.
- $\overset{k}{\rightarrow} P$: P has K as a public key.
- $\overset{-k}{\rightarrow} P$: P has K as a private key.
- P $\overset{k}{\Leftrightarrow}$ Q : X is a secret known only to P and Q.

• $\{X\}_K$: X is encrypted with the key K.

• $\{X\}_y$: X combined with Y.

• $(X)_K$: X is hashed with the key K.

• (X , Y) : X or Y is the part of the message (X, Y).

• SK : The session key shared between two entities

6.7.2 Assumptions:

In this BAN logic analysis, we will use the following notations: Vs as the message sender, Vr as the vehicle receiver, X as message sent, K^{PV} is an element that belongs to the PV keychain in used by Vs and it is used for signing and verification of a given message X, H(X) hashed of message X computed with K^{PV}, K_0 as Vs's current commitment and TS as the message timestamp. We have the following assumptions regarding PV approach:

Vs |≡ (#TS, ID): Vs believes its identity and the freshness of its message timestamp.

Vr |≡ Vs ⇒ (#K_0) : Vr believes that Vs has jurisdiction over K_0 and its freshness

Vr |≡ (Vr $\overset{K_0}{\leftrightarrow}$ Vs): Vr believes the commitment key of Vs received from VAS (in case of VAS-centric approach) or Vs itself (in case of vehicle-centric approach).

$(X)_{K^{PV}}$: X is hashed and signed with the key K^{PV}.

Vr |≡ Vs| ~ X: Vr believes that Vs once said X

#(X) >> in the PV model, Vr cannot guarantee the freshness of message X, due PV's vulnerability as described above. Furthermore, although Vr believes that Vs once said X, but it cannot believe that Vs has jurisdiction or control over X, as a malicious user can construct a fake X using K^{PV}, which will appear as true X.

Next, we provide the analysis of the idealized form of PV+TESLA based on the logical postulates of BAN logic and some further assumptions with regards to the combined approach. K^T represents the TESLA keychain element used to sign a given message X.

Vs ⇒ $(X)_{K^T}$: Vs controls X or has jurisdiction over X as it will disclose K^T later to the receiving vehicles to verify its authenticity. So, X cannot be forged.

Vr ◁ $(X)_{KT}$: Vr sees the message signed with TESLA key

Vr $\overset{H(x)_{KT}}{\longleftrightarrow}$ Vs : Vr believes that message is hashed or signed with TESLA key.

6.7.3 Goals

The following are the goals that need to be achieved:

G1 - $\mathrm{Vr\ believes\ Vs} \mid\sim \mathrm{X}$. receiver vehicle to believe that the legitimate sender Vs said message.

G2: Vr believes freshness of message X.

G3 - $Vr\ beleives\ Vs\ beleives\ X$. the receiver vehicle believes that the legitimate sender vehicle believes its true message X

G4 - $Vr\ believes\ X$. the receiver vehicle believes the truthfulness of message X.

G5 - $Vr\ believes\ \mathrm{Vs} \overset{K^T}{\leftrightarrow} \mathrm{Vr}$. The receiver vehicle believes the disclosed TESLA shared key

G1– The message meaning rule:

That is, we have Vr believes that the key K^T is sent by Vs and sees X signed with K^T as genuine, since only Vs can disclose the secret key K^T. Then, Vr believes that Vs once said X. According to the BAN logic postulates, the message-meaning rule can be written as:

$$\frac{Vr\ believes\ \mathrm{Vr} \overset{K^T}{\leftrightarrow} \mathrm{Vs}, \mathrm{Vr} \lhd (\mathrm{X})K^T}{\mathrm{Vr\ believes\ Vs} \mid\sim \mathrm{X}}$$

G2 –The freshness-conjuncatenation rule expresses the check that a message is recent and, hence, that the sender still believes in it. However in PV+TESLA there is no fresh nonces used, but based on K_0 is changed each time Vs send so it guarantees freshness of each session .

So

$$\frac{Vr \mid\equiv \#\,\mathrm{k0}}{Vr \mid\equiv \#(\mathrm{k0}, \mathrm{X})}$$

G3: That is, in our model we have Vr believes that X could have been uttered only recently (in the present) and that Vs once said X (either in the past or in the present), then Vr believes that Vs believes X. Thus, our protocol guarantees freshness rule, and this is second goal we achieve

$$\frac{Vr\ believes\ \#\ (X), \text{Vr believes Vs} \mid\sim \text{X}}{Vr\ believes\ Vs\ believes\ X}$$

G4 - The jurisdiction rule states that if Vr believes that Vs has jurisdiction over X then Vr trusts Vs on the truth of X and so as from above we achieve third goal which is that Vr believes X Message, So our proposed protocol PV+TESLA could achieve the above three goals of BAN logic as mentioned above.

$$\frac{Vr\ believes\ \text{Vs} \Rightarrow \text{X}, Vr\ believes\ Vs\ believes\ X}{Vr\ believes\ X}$$

G5: as Vr receive the hashed (X) from Vs and then later receives key K^T to apply chain of signing messages to check validity of signing the series of messages based on time intervals so

$$\frac{Vr\ believes\ \text{Vr} \overset{H(x)_{K^T}}{\longleftrightarrow} \text{Vs}, Vr\ believes\ Vs\ believes\ X}{\text{Vr believes Vs} \overset{K^T}{\leftrightarrow} \text{Vr}}$$

6.8 Summary

The RMCCS method described in this chapter uses the random noise component and path loss exponent of the LDPLM model of radio signal propagation to provide a distance estimate that can be used to detect vehicles lying about the position information reported in a CAM message. Unlike other RSSI-based techniques, our method works without any underlying assumption about the sender-receiver link characteristics and the composition of objects in the vicinity. Also, it is an independent scheme such that a receiving vehicle acting alone can determine whether another vehicle is lying about its true position. Furthermore, the RMCCS method provides a measure of uncertainty in the estimated distance that is dynamically computed based on the RSSI variations. We showed through simulation and evaluation that RMCCS performs well in terms of distance estimation and the ability to detect false position reports with an accuracy of about 90% for separation distance less than 100m. The accuracy and lightweight characteristics of RMCCS make it a reliable source of evidence to be part of PV+TESLA model to reduce the risk associated with PV's vulnerability.

The RMCCS method, along with a plausibility checking function adopted from previous studies, is integrated into PV+TESLA. Using simulations representing real life scenarios, the performance of this combined approach is evaluated and compared with the PBA scheme, an existing approach found in the literature. Results indicated that PV+TESLA with RMCCS and plausibility check function reduces TESLA's inherent authentication delay by about 85%. It also minimizes the risk of accepting false messages by taking into account the measure of consistency and trajectory change provided by RMCCS and plausibility check function, respectively. This integrated approach outperforms the PBA technique, especially in high-traffic scenario, where PBA's reliance on successful reception of consecutive messages from the same sender to achieve fast verification becomes ineffective.

The chapter also describes a study conducted to assess the risk of PV's vulnerability to impersonation attack. An attacker scenario exploiting this vulnerability and broadcasting fake messages using a legitimate vehicle's key is modelled through simulations. Results indicate that about 97% of fake messages were detected by the combined effort of RMMC, plausibility check function and detection due to reception of two messages apparently from the same sender signed with the same key. This study also highlights the limited opportunities attackers have in exploiting PV's vulnerability in practice, owing to the difficulty in finding the appropriate configuration to evade reception of genuine messages.

The PV+TESLA model, integrated with RMCCS and plausibility checks, has proven effective in reducing TESLA's authentication delays and minimizing security risks. The study showcases the resilience of the system against potential attackers attempting to exploit PV vulnerabilities. Overall, the findings contribute to the suitability of TESLA-like approaches for securing direct V2V communication systems, emphasizing the robustness and practical viability of the proposed PV+TESLA model.

Chapter 7: Conclusion and Future Research Work

This chapter presents a summary of the contributions and achievements that have materialised from our research. In addition, it proposes some directions for future research building on these results.

7.1 Achievements of the Research Objectives

This research addresses the security of broadcast V2V message-based communication in an ITS context. Particularly, it focuses on ensuring the authenticity and integrity of the messages that share information needed by applications designed to improve the safety of vehicles on the road, reduce accidents, etc. Challenges in securing such V2V safety messages arise from:

(a) the strict performance requirements of the safety applications, particularly low end-to-end message latency and reliable message delivery,
(b) the dynamic population of neighbours within communication range of a given vehicle,
(c) the high volume of message traffic that receivers are expected to process, and
(d) the safety-criticality of the message content.

Standards organisations and government transportation authorities recommend the use of a public key cryptography-based digital signature scheme requiring a VPKI for issuing and managing public certificates to vehicles. However, several research works have reported performance issues inherent in the VPKI-based scheme due to the computational expense of operations used in PKC that mean that latency requirements of safety applications cannot be met in commonly encountered conditions. Furthermore, they make the applications vulnerable to computation-based DoS attacks. Other studies describe the practical implementation challenges associated with VPKI. For instance, concerns raised in [150] centre around the fundamental challenges associated with supporting vehicles crossing boundaries between regulatory regimes with different root-of-trust authorities and the intricate task of distributing new short-term certificates.

Identity-based cryptography is another option considered in the literature to secure V2V messaging system because it eliminates the need for a widespread infrastructure for certificate distribution and management. This greatly simplifies the complexity of the ITS and

alleviates the communication overhead introduced by including certificates in every broadcast message. However, studies report that identity-based solutions encounter performance issues similar to VPKI-based approaches due to the use of asymmetric primitives (e.g. bilinear pairing) for signature generation and verification. Another option is considered is the TESLA scheme, which is built on symmetric cryptography and hash-chains. This makes it a viable option to support the stringent performance requirements of safety applications. Also, it does not require a large-scale infrastructure for its operations compared to the VPKI-based approach, thus avoiding infrastructure-related problems (e.g., inter-boundary regulations) faced by the VPKI-based approach. However, as TESLA was not specifically designed for a highly dynamic V2V setting, certain modifications are needed to make it suitable for application in V2V.

In addition to these alternatives, various VPKI variants, such as priority-based verification and random-based verification, have been proposed. These schemes aim to reduce the burden of verifying all incoming messages by selectively processing them based on specific criteria. However, they face some security challenges. For example, most priority-based verification schemes rely on sender-receiver distance to prioritise the processing of received messages. A malicious actor can exploit this to broadcast messages with false distance information so that its messages are given priority all the time, thus preventing the verification of messages from legitimate vehicles.

A robust V2V message security mechanism is critical for the practical deployment of V2V safety services and the wide acceptability of ITS in society. Given the limitations of the VPKI-based scheme and its variants, it is therefore imperative to explore alternative security solutions that could be more effective in ensuring the message authenticity and integrity and performance requirement of V2V.

7.1.1 RC 1: Performance Comparison of ECDSA and TESLA

The first research contribution of this thesis is a comparison of the performance of the established VPKI-based digital signature scheme using the ECDSA digital signature algorithm, and TESLA. The two schemes are evaluated by means of analytical modelling using queuing theory, and realistic simulations under different traffic conditions in **Chapter 3**.

The findings reveal that with more than 75 vehicles within range of a receiver, ECDSA incurs message latency above 100ms and message drop rates that negatively affect the performance of safety applications. This is consistent with the conclusions of other research works. The standards do propose a congestion control mechanism that reduces message transmission rates and power when message queues are growing. However, this means that information on fewer vehicles will be available to the safety applications when traffic is densest, and it will be updated less frequently. Another option is to improve the onboard processor performance so that vehicles can process messages at a faster rate, but this means delaying deployment until powerful processors are available in the vehicles' onboard unit.

C-ITS road safety applications have stringent requirements for message latency, communication reliability, and message frequency to ensure vehicles are well-informed about the surrounding traffic conditions and potential hazards. The basic set of C-ITS safety use-cases demands an end-to-end message latency of 100ms and a message frequency of 10Hz, whereas advanced safety use-cases require an end-to-end message latency as low as 3ms and a message frequency as high as 50Hz. The TESLA approach exhibited performance within the requirements of basic safety use-cases throughout the range of traffic conditions studied. It outperforms ECDSA in traffic situation where there are more than 75 vehicles within range of a receiver, but the need for a so-called authentication delay, which results in a fixed latency overhead, makes it less attractive when traffic density is below 75. This indicates that it is worthwhile to further investigate the feasibility/desirability of basing V2V message security on the TESLA approach, which has received little attention to date. We have identified three challenges regarding using the standard TESLA scheme in a V2V context, two of which are addressed by the remaining research contributions.

7.1.2 RC 2: Commitment Distribution Schemes for V2V

The second research contribution focusses on addressing TESLA's need for the distribution of authentication information (known as commitments) by a message sender to all potential message receivers. The commitment must arrive before the receiver needs to validate the current key used to sign messages from that sender. In the context of V2V, distribution of such commitments is challenging mainly due to the mobility of vehicles, which makes it difficult to predict which vehicles need specific commitments. **Chapter 4** introduces and evaluates two distinct solutions to this commitment distribution problem: the VAS-centric

approach, involving selective unicasting by a central server (VAS), and the vehicle-centric approach, a proactive scheme where vehicles themselves periodically broadcast commitments. A comprehensive comparative analysis and performance evaluation, conducted through realistic simulations under varying traffic conditions, considered factors such as timeliness, distribution efficiency, and the impact on safety message delivery. Additionally, an analytical and simulation-based evaluation of a related V2V commitment distribution approach found in [25] was performed and compared with the proposed solutions. The results indicate that the VAS-centric approach outperforms the vehicle-centric solution and the approach in [25]. Desirable properties demonstrated include: delivering most commitments in time, and avoiding distributing commitments not relevant to vehicles, achieving about 98% commitment utilisation rate. Most importantly, the VAS-centric solution does not affect the performance of the safety message latency and delivery, as it uses a different communication channel, which is beneficial to the safety applications that rely on safety messages to function effectively. Security analysis confirms that it is secure under the well-known ROM model and resilient to attacks like commitment forgery, collision, etc. Thus, the VAS-centric approach is potentially suitable for implementation in practice.

7.1.3 RC 3: Analysis of Approaches for reducing Authentication Delay

The third research contribution addresses the challenge of delaying the disclosure of a message verification key until all messages signed with that key have been received. This is inherent to the standard TESLA approach, and it gives rise to a minimum message latency known as the authentication delay. This exceeds the maximum tolerable message latency of around 20ms or less for some advanced safety use-cases (e.g. cooperative sensing, platooning) that have more stringent performance requirements than the basic safety use-cases. **Chapter 5** describes in detail a modified version of TESLA referred to as the prompt verification (PV) model, which entirely removes the authentication delay. In this model, a fresh key is used to sign each outgoing message, and the key is broadcast together with the message. Compared to standard TESLA, the PV model has several advantages, which include fast message verification, no requirement of storage space to temporarily buffer received messages, and elimination of the need for time synchronisation between senders and receivers. However, these benefits come at the cost of a vulnerability that can be exploited by malicious agents. An attacker can extract the key from a legitimate vehicle's message and

use it to construct a false message purporting to come from that vehicle. Vehicles receiving only the false message are unable to distinguish it from a genuine one by cryptographic means. The significance and risk associated with the PV model's vulnerability to impersonation attack have been thoroughly analysed and assessed. Although the window of opportunity for its exploitation is limited, it can endanger human lives or cause damage to vehicles and other road infrastructure. We concluded, therefore, that the additional risk arising from PV's vulnerability to impersonation attack is high. Consequently, different ways to mitigate PV's vulnerability were proposed and their effectiveness in reducing the associated risk were assessed. Our analysis shows that the use of the VAS-centric commitment distribution scheme reduces the vulnerability as it further limits the window of opportunity to the malicious agent quite considerably, making the attack more difficult to perpetrate.

7.1.4 RC 4: PV+TESLA

The fourth research contribution, detailed in **Chapters 5 and 6**, examines a specific way to exploit PV's advantages while mitigating its vulnerability. It introduces, analyses, and evaluates a proposal called prompt verification with TESLA as a backup (PV+TESLA). This protocol, based on TESLA, aims to reduce the effective authentication delay while maintaining an acceptable level of risk. In this model, vehicles primarily verify messages using PV but have the option to employ TESLA for selected messages. The proportion of messages verified using PV directly influences the average authentication delay: a higher fraction results in a smaller average verification delay. However, this choice introduces a trade-off, as a greater reliance on PV increases vulnerability to impersonation attacks.

For the PV+TESLA approach to be effective, a mechanism is required to decide for a given received message, whether to rely on PV or to employ TESLA, and so incur the authentication delay. Criteria, which may be used in combination include: random selection, and use of TESLA at the start of /after a break in the sequence of messages from a given sender, or if there is reason to be suspicious of a message. In the case of a sequence of messages describing a consistent trajectory, successful verification of one or more messages using TESLA provides evidence that the whole trajectory is trustworthy. Any evidence indicating that an attacker is trying to exploit the PV vulnerability is an important input to this decision. A signature of such exploitation is that the true position of the message sender is different from the position

reported in the message. One way to gather evidence of this is to check whether the claimed position is consistent with other sources of position information including physical features of the transmission signal.

7.1.5 RC 5: RMCCS

The fourth research contribution, detailed in **Chapter 6**, comprises of two key aspects. The first introduces a novel RSSI-based distance estimation mechanism called RMCCS. The main advantage over other approaches to distance estimation based on the LDPLM model of radio signal propagation is that it does not require information about environmental conditions (i.e., built-up areas, traffic, etc.) from external sources in order to determine the path loss exponent. Instead, it extracts this parameter value from the random noise component of the radio signal. The method also generates a measure of uncertainty on the distance estimate. The distance estimate can be compared with the true distance computed from the receiver's known position and the claimed sender position reported in the message. This provides means to judge whether a message sender is reporting a false position in its message. The performance of RMCCS has undergone a rigorous evaluation using a GEMV^2 simulation framework that incorporates realistic signal propagation characteristics, such as the presence of buildings, foliage, and other vehicles. Results demonstrate that RMCCS achieves an accuracy level of approximately 90% for separation distances less than 100m, positioning it as a promising source of evidence for the PV+TESLA approach. The application of RMCCS is not limited to this context. It can be used generally as a data-centric based misbehaviour detection mechanism to detect false position reported by a node or to estimate distance between sender and receiver.

The second contribution involves integrating and evaluating the performance of RMCCS and a plausibility check function within the PV+TESLA model to provide signs of potential exploitation of PV's vulnerability to impersonation attack. The distance comparison facilitated by RMCCS allows for confirmation of correctness regarding the claimed sender's position reported in the message. The plausibility check function is introduced to provide additional evidence by comparing the sender's trajectory reported in the current message with record of its last trajectory available at the receiver. Other source of evidence considered include reception of two messages from apparently the same sender ID signed with the same key. A thoughtfully designed decision-making algorithm incorporates these sources of evidence to

determine whether to accept an incoming message based on PV verification or to invoke TESLA verification when suspicious activity is evident.

The performance of this integrated model is extensively evaluated through simulations. Comparative analyses are conducted, including the assessment of a similar TESLA-like technique found in the literature called PBA. Results reveal that PV+TESLA reduces authentication delay by approximately 85% compared to standard TESLA, while minimizing the risk of accepting false messages. This integrated model outperforms the PBA technique, especially in high-traffic scenarios where message losses are inevitable. PV+TESLA is not sensitive to message losses, unlike PBA in which the verification delay degrades with message losses as it requires the successful reception of consecutive messages from the same sender to achieve its immediate verification property.

Moreover, a comprehensive risk assessment of PV's vulnerability to impersonation in this integrated model is conducted through modelling and simulation of attacker activity. The combined effort of RMCCS, the plausibility check, and message clash-based detection achieve a true positive rate of up to 97%, demonstrating the effectiveness of the integrated approach in mitigating security risks.

The relationships between the research questions, research contributions, and thesis chapters are shown in Table 7.1.

Table 7-1: Mapping of Research Questions, Research Contributions and Thesis Chapters

Research Question	Research Contribution (RC)	Chapters
-	Performance comparison of ECDSA and TESLA	3
1	Distribution of Commitments in V2V	4
2	Analysis of Approaches for reducing authentication delay	5
2	Prompt Verification with TESLA as back up (PV+TESLA)	5 and 6
3	RSSI-based Message Consistency Checking Scheme (RMCCS), and integration of RMCCS and plausibility check function with PV+TESLA model	6

The proposed security approach for ensuring authenticity and integrity of V2V messages using a combination of PV+TESLA to balance a reduction in effective authentication delay against vulnerability to impersonation attack, with RMCCS and plausibility check function to help confirm correctness of information reports in the messages, and the VAS-centric to distribute

commitments to vehicles, is highly promising. However, there are some open issues that need to be address in order to make it more solid and robust. These are discussed below together with ideas for research to address them.

7.2 Future work and Research Directions

In this section, we describe future research directions for building a viable security mechanism based on TESLA for V2V messaging. In particular, we point out some areas that can be explored to improve the techniques proposed in this research work and indicate practical implementation considerations of TESLA-like schemes in this context.

7.2.1 Improving the relevance function used by the VAS.

In the VAS-centric solution described in Chapter 4, the VAS executes a relevance prediction function to determine vehicles that would influence each other's behaviour within a given time interval. The current version of the function is based on the straight-line distance between two given vehicles and the average vehicle speed in the locality. This can be improved by taking into account other factors such as the direction of vehicles' movement and the topology of the road system the vehicles are moving on. This will reduce the reception of commitments that are not associated with relevant safety messages and also the number of relevant messages that cannot be used due to late or non-delivery of a commitment, which in turn improves the accuracy, timeliness and distribution efficiency of the scheme.

7.2.2 Further simulation and performance evaluation of attacker activity in the PV model

In section 6.6.2, we conducted a simulation with a single attacker scenario to assess the significance of PV's vulnerability. However, further investigation is needed, involving the implementation of a broad set of attack scenarios exploiting PV's vulnerability. For example, the MM and MS attack cases described above. This will require the generation of different false messages intended to influence the behaviour of various target vehicles. A subset of vehicles should be classified as attackers, each running the specified attacking algorithm. The simulation should cover a diverse range of traffic scenarios and attacker densities. Conducting this could facilitate the creation of a comprehensive dataset featuring diverse attack scenarios. This dataset can be made available to the research community to develop, test,

and compare misbehaviour detection mechanisms, similar to the datasets produced in [151] [152]. The authors generated, through simulation, a comprehensive dataset called VeReMi that encompasses a wide range of traffic behaviours and various implementation of position falsification attacks.

7.2.3 Application of Machine Learning Algorithms in PV model

The adoption of artificial intelligence (AI) and machine learning (ML) techniques within the ITS is growing rapidly. They have been widely applied in various areas such as autonomous driving, intelligent traffic control, designing highly accurate traffic flow prediction algorithms, network traffic control, etc. With regards to security in V2V, both the classification ability of supervised ML and the clustering ability of unsupervised ML makes them suitable for detecting malicious or anomalous behaviour. These types of ML algorithms have the ability to learn complex patterns and behaviour, making them capable to distinguish nodes that deviate from normal behaviour. For example, the work in [153] employs K-Nearest Neighbour (K-NN) and Random Forest ML algorithms to detect false position information reported in received messages. We propose to investigate the feasibility of applying machine learning algorithms to detect attacks exploiting PV's vulnerability described in Chapter 5. A dataset consisting of different attack types can be used to train the ML algorithms to classify messages as legitimate or false.

7.2.4 Investigating non-repudiation mechanism for TESLA-like schemes

As described in Chapter 2, one of the challenges of the standard TESLA is that it lacks non-repudiation. The problem is that verification of the authenticity of a message depends on knowing that the message was sent before the key used to sign it was disclosed. A genuine message sent at the time claimed, before key disclosure, cannot be distinguished from a fabricated message constructed with the same key at a later time after the key has been disclosed. This means, for example, that Vehicle A cannot prove to a third party that Vehicle B sent it a particular message. Thus, there is a need to investigate and analyse different techniques that can be combined with TESLA to efficiently achieve a non-repudiation capability. Providing this feature would also be effective in dealing with PV's vulnerability to impersonation attack.

A few research works have proposed solutions to achieving non-repudiation in TESLA-like schemes. For example, the authors of [154] introduce an authentication token that is linked with the verification key for each vehicle. Also, the study in [155] employs a trusted central node that communicates securely with the vehicles. Each vehicle sends its disclosed key to the central node, which then distributes it to the receivers. Other techniques such as time-bound symmetric keys, trusted time stamp or logging mechanisms should also be explored. s part of such research, it is important to evaluate the trade-off between the degree of non-repudiation and the overhead due to the additional information necessary for the non-repudiation property.

7.2.5 Standardisation of TESLA-like scheme

This thesis is concluded by strongly suggesting that a TESLA-like security solution should be adopted and used in practice for V2V messaging. Thus, we advocate that Working Group 5 Security of ETSI-ITS, the National Highway Traffic Safety Administration (NHTSA), which is actively working on V2V in the US, automotive industry projects on V2V security e.g. Car2Car Communication Consortium (C2C CC) Security WG, work towards its standardisation as a viable alternative to the current VPKI approach. We will seek opportunities to present the analysis and findings conducted in this research to committees of these bodies in order to persuade them to start the standardisation process.

References

[1] ITS, "Intelligent Transport Systems (ITS); Communications Architecture, ETSI EN Standard 302 665 V1.1.1,," Sept. 2010.

[2] USDOT, "Connected Vehicles," [Online]. Available: https://its.dot.gov/research_archives/connected_vehicle/connected_vehicle.htm. [Accessed 15 March 2021].

[3] A. Festag, "Cooperative intelligent transport systems standards in Europe," *IEEE communications magazine,* vol. 52, no. 12, pp. 166-172, 2014.

[4] H. Makano, "C-ITS and Connected Automated Driving in Japan," 2017.

[5] ETSI, "Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 2: Specification of Cooperative; Awareness Basic Service. ETSI EN 302 637-2 V1.3.1," 2014.

[6] SAE, "SAE J2735 - Dedicated Short Range Communications (DSRC) Message Set Dictionary," *SAE Int. DSRC Committee,* 2009.

[7] ETSI, "Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 3: Specifications of Decentralized Environmental Notification Basic Service. ETSI EN 302 637-3 V1.2.1," 2014.

[8] ETSI, "Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Definitions. ETSI TR 102 638 V1.1.1," 2009.

[9] 3GPP, "3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Study on LTE support for Vehicle to Everything (V2X) Services (Release 14) 3GPP TR 22.885 V14.0.0," 2015.

[10] 3GPP, "3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Study on enhancement of 3GPP Support for 5G V2X services (Release 16). 3GPP TR 22.886 V16.1.1," 2018.

[11] J. B. Kenney, "Dedicated short-range communications (DSRC) standards in the United States," *Proceeding of IEEE,* vol. 99, no. 7, pp. 1162-1182, 2011.

[12] A. Festag, "Cooperative intelligent transport systems standards in Europe.," *IEEE communications magazine,* vol. 52, no. 12, pp. 166-172, 2014.

[13] S. Chen, J. Hu, Y. Shi, Y. Peng, J. Fang, R. Zhao and L. Zhao, "Vehicle-to-everything (V2X) services supported by LTE-based systems and 5G," *IEEE Communications Standards Magazine,* vol. 1, no. 2, pp. 70-76, 2017.

[14] IEEE, "1609.2-2016 - IEEE Standard for Wireless Access in Vehicular Environments Security Services for Applications and Management Messages. IEEE P1609.2/D9.0," 2016.

[15] E. T. 097, "Intelligent Transport Systems (ITS); Security; Security header and certificate formats; Release 2," ETSI, 2021.

[16] M. B. Brahim, E. B. Hamida, F. Filali and N. Hamdi, "Performance impact of security on cooperative awareness in dense urban vehicular networks," in *IEEE 11th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, 2015.

[17] M. A. Javed and E. B. Hamida., "On the interrelation of security, QoS, and safety in cooperative ITS.," *IEEE Transactions on Intelligent Transportation Systems ,* vol. 18, no. 7, pp. 1943-1957, 2016.

[18] S. Tzeng, S. Horng, T. Li, X. Wang, P. Huang and M. K. Khan., "Enhancing Security and Privacy for ID-based batch verification schemes in VANET," *IEEE Transactions on Vehicular Technology,* vol. 66, no. 4, pp. 3235-3248, 2017.

[19] S. Tangade, S. S. Manvi and P. Lorenz, "Decentralized and scalable privacy preserving authentication scheme in vanets," *IEEE Transactions on Vehicular Technology,* vol. 67, no. 9, pp. 8647-8655, 2018.

[20] A. Perrig, R. Canetti, J. D. Tygar and D. Song, "The TESLA broadcast authentication protocol," *Rsa Cryptobytes,* vol. V, no. 2, pp. 2-13, 2002.

[21] USDOT, "Vehicle-to-Vehicle Communications: Readiness of V2V Technology for Application. DOT HS 812 014," U.S. Department of Transportation. National Highway Traffic Safety Administration (NHTSA) , 2014.

[22] C. Lyu, D. Gu, Y. Zeng and P. Mohapatra., "PBA: Prediction-based authentication for vehicle-to-vehicle communications.," *IEEE transactions on dependable and secure computing,* vol. 13, no. 1, pp. 71-83, 2015.

[23] B. Ying and A. Nayak., "Anonymous and lightweight authentication for secure vehicular networks," *IEEE Transactions on Vehicular Technology,* vol. 66, no. 12, pp. 10626-10636, 2017.

[24] B. Ying, D. Makrakis and H. T. Mouftah., "Privacy preserving broadcast message authentication protocol for VANETs.," *Journal of Network and Computer Applications ,* vol. 36, no. 5, pp. 1352-1364, 2013.

[25] S. Bao, W. Hathal, H. Cruickshank, Z. Sun, P. Asuquo and A. Lei, "A lightweight authentication and privacy-preserving scheme for VANETs using TESLA and bloom filters," *Elsevier ICT Express,* 2017.

[26] ETSI, "Intelligent Transport Systems (ITS); Communications Architecture. ETSI EN 302 665 V1.1.1," 2010.

[27] J. B. Kenney, "Dedicated short-range communications (DSRC) standards in the United States," *Proceedings of the IEEE ,* vol. 99, no. 7, pp. 1162-1182, 2011.

[28] n. T. S. (ITS), "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 EN 302 636-4-1, 2017.

[29] 302-636-5-1, "Intelligent Transport Systems (ITS); Vehicular Communications; GeoNetworking;Part 5: Transport Protocols; Sub-part 1: Basic Transport Protocol," ETSI EN 302 636-5-1, 2017.

[30] ETSI-ES, "Intelligent Transport Systems (ITS); European profile standard for the physical and medium access control layer of Intelligent Transport Systems operating in the 5 GHz frequency band," ETSI ES 202 663, 2009.

[31] Z. Mir and F. Filali, " LTE and IEEE 802.11 p for vehicular networking: a performance evaluation.," *EURASIP Journal on Wireless Communications and Networking,* vol. 1, pp. 1-15, 2014.

[32] X. Wu, S. Subramanian, R. Guha, R. White, J. Li, K. Lu, A. Bucceri and T. Zhang, "Vehicular communications using DSRC: challenges, enhancements, and evolution.," *IEEE Journal on Selected Areas in Communications,* vol. 31, no. 9, pp. 399-408, 2013.

[33] M. Boban, A. Kousaridas, K. Manolakis, J. Eichinger and W. Xu, "Connected roads of the future: Use cases, requirements, and design considerations for vehicle-to-everything communications.," *IEEE vehicular technology magazine, ,* vol. 13, no. 3, pp. 110-123, 2018.

[34] 3.-T. 22.885, "3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Study on LTE support for Vehicle to Everything (V2X) services TR 22.885 V14.0.0 (Release 14)," 2015.

[35] 3. T. 23.285, "Universal Mobile Telecommunications System (UMTS); LTE; Architecture Enhancements for V2X services TS 23.285, v14.2.0," 2017.

[36] 3GPP, "3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Multimedia Broadcast/Multicast Service (MBMS); Architecture and Functional Description TS 23.246 V16.1.0 (Release 16)," 2019.

[37] 3GPP, "3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Proximity-based services (ProSe); TS 23.303 Stage 2 (Release 17) V17.0.0," 2021.

[38] S. Chen, J. Hu, Y. Shi, Y. Peng, J. Fang, R. Zhao and L. Zhao, "Vehicle-to-everything (V2X) services supported by LTE-based systems and 5G.," *IEEE Communications Standards Magazine ,* vol. 1, no. 2, pp. 70-76, 2017.

[39] 36.101, "Evolved Universal Terrestrial Radio Access (E-UTRA);User Equipment (UE) radio transmission and reception (3GPP TS 36.101 version 14.3.0 Release 14)," 2017.

[40] M. Wang, M. Winbjork, Z. Zhang, R. Blasco, H. Do, S. Sorrentino, M. Belleschi and Y. Zang, "Comparison of LTE and DSRC-based connectivity for intelligent transportation systems.," *In IEEE 85th vehicular technology conference (VTC Spring),* 2017.

[41] T. Bey and G. Tewolde, "Evaluation of DSRC and LTE for V2X.," *In IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC),* pp. 1032-1035, 2019.

[42] A. Bazzi, B. Masini, A. Zanella and I. Thibault, "On the performance of IEEE 802.11 p and LTE-V2V for the cooperative awareness of connected vehicles.," *IEEE Transactions on Vehicular Technology,* vol. 66, no. 11, pp. 10419-10432..

[43] SAE, "Dedicated Short Range Communications (DSRC) Message Set Dictionary, SAE Standard J2735_201601,," 2016.

[44] ETSI, "Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Definitions ETSI TR 102 638. V.1.1.1," *ETSI,* 2009.

[45] 3GPP, ""“Service requirements for enhanced V2X scenarios (release 15),” 3rd Generation Partnership Project, Technical Report. 3GPP TR 22.186 V15.0.0," 2017.

[46] D. Dolev and Y. Andrew, "On the security of public key protocols," *IEEE Transactions on information theory 29, no. 2,* pp. 198-208, 1983.

[47] M. Raya and J.-P. Hubaux., "The security of vehicular ad hoc networks.," *In Proceedings of the 3rd ACM workshop on Security of ad hoc and sensor networks,* pp. 11-21, 2005.

[48] PRESERVE, "PREparing SEcuRe VEhicle-to-X Commmunications System. Security Requirements of Vehicle Security Architecture," 2011.

[49] E.-1. 893, "Intelligent Transport Systems (ITS);Security; Threat, Vulnerability and Risk Analysis (TVRA)," 2017.

[50] 3GPP-TR33.885, "3rd generation partnership project; Technical Specification Group Services and System Aspects; Study on Security Aspects for LTE Support of Vehicle-to-Everything (V2X) Services (release 14)," *3GPP,* 2017.

[51] IEEE, "IEEE Standard for Wireless Access in Vehicular Environments Security Services for Applications and Management Messages, Revision of IEEE Std 1609.2-2013," IEEE, 2016.

[52] ETSI, "TS 102 940 Intelligent Transport Systems (ITS); Security; ITS Communications Security Architecture and Security Management," ETSI, 2018.

[53] S. Ibrahim and M. Hamdy, "A comparison on VANET authentication schemes: Public Key vs. Symmetric Key.," *In Tenth International Conference on Computer Engineering & Systems (ICCES),* pp. 341-345, 2015.

[54] F. Haidar, A. Kaiser and B. Lonc, "On the performance evaluation of vehicular PKI protocol for V2X communications security.," *In IEEE 86th vehicular technology conference (VTC-Fall),* pp. 1-5, 2017.

[55] Q. Li, "A V2V Identity Authentication and Key Agreement Scheme Based on Identity-Based Cryptograph.," *Future Internet,* vol. 15, no. 1, 2023.

[56] I. Ali, L. Tandoh and F. Li., "An efficient identity-based signature scheme without bilinear pairing for vehicle-to-vehicle communication in VANETs," *Journal of Systems Architecture ,* vol. 103, 2020.

[57] C. Zhang, P.-H. Ho and T. Janos, "On batch verification with group testing for vehicular communications," *Wireless Networks ,* vol. 17, pp. 1851-1865, 2011.

[58] T. W. Chim, S.-M. Yiu, H. Lucas C and V. O. Li., "SPECS: Secure and privacy enhancing communications schemes for VANETs," *Ad Hoc Networks,* vol. 9, no. 2, pp. 189-203, 2011.

[59] S.-J. Horng, S.-F. Tzeng, Y. Pan, F. Pingzhi, W. Xian, L. Tianrui and K. K. Muhammad, "b-SPECS+: Batch verification for secure pseudonymous authentication in VANET.," *IEEE transactions on information forensics and security,* vol. 8, no. 11, pp. 1860-1875, 2013.

[60] M. A. Javed and E. B. Hamida., "Adaptive security mechanisms for safety applications in internet of vehicles," *In IEEE 12th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob),* pp. 1-6, 2016.

[61] A. Perrig, R. Canetti, J. D. Tygar and D. Song, "The TESLA broadcast authentication protocol," *Rsa Cryptobytes,* vol. 5, 2005.

[62] C. Lyu, D. Gu, Y. Zeng and P. Mohapatra, "PBA: Prediction-based authentication for vehicle-to-vehicle communications," *IEEE Transactions on Dependable and Secure Computing,* vol. 13, no. 1, pp. 71-83, 2016.

[63] C. Lyu, D. Gu, X. Zhang, S. Sun and Y. Tang, "Efficient, fast and scalable authentication for vanets," in *IEEE Wireless Communications and Networking Conference (WCNC)*, 2013.

[64] M. Lalli and G. S. Graphy, "Prediction based dual authentication model for VANET," in *International Conference on Computing Methodologies and Communication (ICCMC)*, 2017.

[65] H.-C. Hsiao, A. Studer, C. Chen, A. Perrig, F. Bai, B. Bellur and A. Iyer, "Flooding-resilient broadcast authentication for VANETS," in *Proceedings of the 17th annual international conference on Mobile computing and networking*, 2011.

[66] B. Ying, D. Makrakis and H. T. Mouftah, "Privacy preserving broadcast message authentication protocol for VANETs," *Journal of Network and Computer Applications,* vol. 36, pp. 1352-1364, 2013.

[67] A. Studer, F. Bai, B. Bellur and A. Perrig, "Flexible, extensible, and efficient VANET authentication," *Journal of Communications and Networks,* vol. 11, no. 6, pp. 574-588, 2009.

[68] Y.-C. Hu and K. P. Laberteaux, "Strong VANET security on a budget," *Proceedings of Workshop on Embedded Security in Cars (ESCAR),* vol. 6, pp. 1-9, 2006.

[69] H. Jin and P. Panos, "DoS-resilient cooperative beacon verification for vehicular communication systems," *Ad Hoc Networks 90,* p. 101775, 2019.

[70] J. Xu, W. Liangliang, W. Mi, L. Yu and C. Kefei, "DPB-MA: Low-Latency Message Authentication Scheme Based on Distributed Verification and Priority in Vehicular Ad Hoc Network," *IEEE Transactions on Vehicular Technology ,* vol. 72, no. 4, pp. 5152-5166, 2022.

[71] S. Biswas and J. Mišić, "Relevance-based verification of VANET safety messages," *In 2012 IEEE International Conference on Communications (ICC),* pp. 5124-5128, 2012.

[72] S. Biswas and J. Mišić., "A cross-layer approach to privacy-preserving authentication in WAVE-enabled VANETs.," *IEEE Transactions on Vehicular Technology ,* vol. 62, no. 5, pp. 2182-2192, 2013.

[73] M. A. Javed, B. H. Elyes, A.-F. Ala and B. Bhargava., "Adaptive security for intelligent transport system applications.," *IEEE Intelligent Transportation Systems Magazine,* vol. 10, no. 2, pp. 110-120, 2018.

[74] M. Raya, P. Panos and J.-P. Hubaux., "Securing vehicular communications," *IEEE wireless communications,* vol. 13, no. 5, pp. 8-15, 2006.

[75] S. Biswas and J. Mišić, "Location-based anonymous authentication for vehicular communications.," *IEEE 22nd International Symposium on Personal, Indoor and Mobile Radio Communications,* pp. 1213-1217, 2011.

[76] C. Zhang, X. Lin, R. Lu, P.-H. Ho and X. Shen., "An efficient message authentication scheme for vehicular communications," *IEEE transactions on vehicular technology,* vol. 57, no. 6, pp. 2257-3368, 2008.

[77] M. Patra, T. Rahul and R. M. C. Siva, "Improving delay and energy efficiency of vehicular networks using mobile femto access points," *IEEE Transactions on vehicular Technology,* vol. 66, no. 2, pp. 1496-1505, 2016.

[78] S. Chen, J. Hu, Y. Shi, L. Zhao and W. Li., "A vision of C-V2X: Technologies, field testing, and challenges with Chinese development.," *IEEE Internet of Things Journal,* vol. 7, no. 5, pp. 3872-3881, 2020.

[79] 3GPP_TR36.885, "3rd Generation Partnership Project: Technical Specification Group Radio Access Network: Study on LTE-based V2X Services; (Release 14) V14.0.0," 3GPP, 2016.

[80] A. Bazzi, B. M. Masini and A. Zanella., "How many vehicles in the LTE-V2V awareness range with half or full duplex radios?," *2017 15th International Conference on ITS Telecommunications (ITST),,* pp. 1-6, 2017.

[81] T. 22.186, "“Enhancement of 3GPP support for V2X scenarios; Stage 1”, Release 16, V16.1.0. December, 2018," *3GPP,* December 2018.

[82] X. Lin, X. Sun, X. Wang, C. Zhang, P. Ho and X. Shen, "TSVC: Timed efficient and secure vehicular communications with privacy preserving.," *IEEE transactions on wireless communications,* vol. 7, no. 12, pp. 4987-4998, 2008.

[83] NXP, "RoadLINK® SAF5400 Single Chip Modem for V2X," [Online]. Available: https://www.nxp.com/products/wireless/dsrc-safety-modem/roadlink-saf5400-single-chip-modem-for-v2x:SAF5400. [Accessed 05 07 2021].

[84] Auto-talks, "CRATON2," [Online]. Available: https://www.auto-talks.com/product/craton2/. [Accessed 06 07 2021].

[85] M. A. R. Baee, E. F. Leonie Simpson and J. Pieprzyk., "Broadcast authentication in latency-critical applications: On the efficiency of IEEE 1609.2," *IEEE Transactions on Vehicular Technology,* vol. 68, no. 12, pp. 11577-11587, 2019.

[86] J. C. S. Arenas, T. Dudda and L. Falconetti., " Ultra-low latency in next generation LTE radio access.," *In11th International ITG Conference on Systems, Communications and Coding,* pp. 1-6, 2017.

[87] Y. Bidi, M. Dimitrios and Hussein T. Mouftah, "Privacy preserving broadcast message authentication protocol for VANETs," *Journal of Network and Computer Applications,* vol. 36, pp. 1352-1364 , 2013.

[88] M. A. Javed and E. B. Hamida., "On the interrelation of security, QoS, and safety in cooperative ITS," *IEEE Transactions on Intelligent Transportation Systems,* vol. 18, no. 7, pp. 1943-1957, 2016.

[89] C. Lyu, D. Gu, Y. Zeng and P. Mohapatra, "PBA: Prediction-based authentication for vehicle-to-vehicle communications," *IEEE Transactions on Dependable and Secure Computing,* vol. 13, no. 1, pp. 71-83, 2016.

[90] E. B. Hamida and M. A. Javed., "Channel-aware ECDSA signature verification of basic safety messages with k-means clustering in VANETs.," *In IEEE 30th International Conference on Advanced Information Networking and Applications (AINA),* pp. 603-610, 2016.

[91] S. Nacy, T. Oh and J. Leone, "Implementation of SHA-1 and ECDSA for vehicular ad-hoc network using NS-3," *In Proceedings of the 2nd annual conference on Research in information technology,* pp. 83-88, 2013.

[92] J. H and P. P., "DoS-resilient cooperative beacon verification for vehicular communication systems.," *Ad Hoc Networks,* p. 101775, 2019.

[93] J. J. Haas, Y.-C. Hu and K. P. Laberteaux., "Real-world VANET security protocol performance," *In GLOBECOM IEEE Global Telecommunications Conference,* pp. 1-7, 2009.

[94] NS3, "Network Simulator 3," [Online]. Available: https://www.nsnam.org/. [Accessed 12 07 2021].

[95] D. Krajzewicz, J. Erdmann, M. Behrisch and L. Bieker, "Recent development and applications of SUMO-Simulation of Urban MObility.," *International journal on advances in systems and measurements ,* vol. 5, no. No. 3&4 , 2012.

[96] SUMO. [Online]. Available: https://www.eclipse.org/sumo/. [Accessed 07 07 2021].

[97] G. Calandriello, P. Papadimitratos, J.-P. Hubaux and A. Lioy., "On the performance of secure vehicular communication systems.," *IEEE transactions on dependable and secure computing ,* vol. 8, no. 6, pp. 898-912, 2010.

[98] D. Duarte, L. Silva, B. Fernandes, M. Alam and J. Ferreira., ""Implementation of Security Services for Vehicular Communications.," *In International Conference on Future Intelligent Vehicular Technologies, Springer, Cham,* pp. 79-90, 2016.

[99] J. Petit and Z. Mammeri, "Authentication and consensus overhead in vehicular ad hoc networks.," *Telecommunication systems,* vol. 52, no. 4, pp. 2699-2712, 2013.

[100] ETSI, "ETSI TS 102 894-1 Intelligent Transport Systems (ITS); Users and applications requirements; Part 1: Facility layer structure, functional requirements and specifications," *ETSI,* vol. V1.1.1, 2013.

[101] 3GPP, "3rd Generation Partnership Project; Technical Specification Group Services and Systems Aspects; Architecture Enhancements for V2X services (Release 16) 3GPP TS 23.285 V16.2.0," 2019.

[102] M. A. Javed and B. H. Elyes, "On the interrelation of security, QoS, and safety in cooperative ITS.," *IEEE Transactions on Intelligent Transportation Systems,* vol. 18, no. 7, pp. 1943-1957, 2016.

[103] A. Vinel, L. Nikita and I. Pavel, "Modeling of V2V communications for C-ITS safety applications: A CPS perspective.," *IEEE Communications Letters,* vol. 8, pp. 1600-1603, 2018.

[104] F. Abrate, V. Andrea and S. Riccardo, "An analytical packet error rate model for WAVE receivers," *In IEEE Vehicular Technology Conference (VTC Fall),* pp. 1-5, 2011.

[105] P.-C. Lin and R.-H. Hwang., "Enhancing misbehavior detection in 5G Vehicle-to-Vehicle communications.," *IEEE Transactions on Vehicular Technology ,* 2020.

[106] ETSI, "ETSI TS 101 539-1 Intelligent Transport Systems (ITS); V2X Applications; Part 1: Road Hazard Signalling (RHS) application requirements specification, V1.1.1," ESTI TS, 2013.

[107] M. Bellare and R. Phillip, "Random oracles are practical: A paradigm for designing efficient protocols.," *In Proceedings of the 1st ACM Conference on Computer and Communications Security,* pp. 62-73, 1993.

[108] B. Palaniswamy, C. Seyit, E. Foo, S. Leonie, A. R. B. Mir and P. Josef, "Continuous authentication for VANET," *Vehicular Communications ,* no. 25, p. 100255, 2020.

[109] I. Ali, Y. Chen, U. Niamat, K. Rajesh and H. Wen, "An efficient and provably secure ECC-based conditional privacy-preserving authentication for vehicle-to-vehicle communication in VANETs.," *IEEE Transactions on Vehicular Technology ,* vol. 70, no. 2, pp. 1278-1291, 2021.

[110] J. Zhang, Z. Hong, C. Jie, T. Miaomiao, X. Yan and L. Lu, "Edge computing-based privacy-preserving authentication framework and protocol for 5G-enabled vehicular networks.," *IEEE Transactions on Vehicular Technology ,* vol. 69, no. 7, pp. 7940-7954, 2020.

[111] D. Pointcheval and S. Jacques, "Security arguments for digital signatures and blind signatures," *Journal of cryptology,* vol. 13, pp. 361-396, 2000.

[112] ETSI, "Intelligent Transport Systems (ITS);V2X Applications;Part 3: Longitudinal Collision Risk Warning (LCRW) ETSI TS 101 539-3," *ETSI Technical Specification,* vol. 1.1.1, 2013.

[113] A. Talebpour, H. S. Mahmassani and F. E. Bustamante., "Modeling driver behavior in a connected environment: Integrated microscopic simulation of traffic and mobile wireless telecommunication systems.," *Journal of the Transportation Research Record ,* vol. 1, pp. 75-86, 2016.

[114] M. Sun, M. Li and R. Gerdes, "A data trust framework for VANETs enabling false data detection and secure vehicle tracking," *In IEEE Conference on Communications and Network Security (CNS),* pp. 1-9, 2017.

[115] R. Hussain, J. Lee and S. Zeadally., "Trust in VANET: A survey of current solutions and future research opportunities," *IEEE transactions on intelligent transportation systems,* vol. 22, no. 5, pp. 2553-2571, 2020.

[116] E.-T. 102-731, "Intelligent Transport Systems (ITS); Security; Security Services and Architecture," ETSI, 2010.

[117] ETSI, "Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Local Dynamic Map (LDM); Rationale for and guidance on standardisation. ETSI TR 102 863," vol. V1.1.1, 2011.

[118] A. Sarker and H. Shen., "A data-driven misbehavior detection system for connected autonomous vehicles," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies ,* vol. 4, no. 2, pp. 1-21, 2018.

[119] S. Ruj, C. Marcos A., H. Zhen, N. Amiya and S. Ivan, "On data-centric misbehavior detection in VANETs," *In IEEE Vehicular Technology Conference (VTC Fall),* pp. 1-5, 2011.

[120] ETSI, "Intelligent Transport Systems (ITS); V2X Applications; Part 2: Intersection Collision Risk Warning (ICRW) application requirements specification ETSI TS 101 539-2 V1.1.1," *ETSI ,* 2018.

[121] F. Ahmad and A. Asma, "A novel context-based risk assessment approach in vehicular networks," *In 30th International Conference on Advanced Information Networking and Applications Workshops (WAINA),* pp. 466-474, 2016.

[122] S. P. Kadhirvelan and S.-R. Andrew, "Threat modelling and risk assessment within vehicular systems.," Master's Thesis, Chalmers University of Technology, 2014.

[123] M. Lalli and G. S. Graphy, "Prediction based dual authentication model for VANET," *In IEEE International Conference on Computing Methodologies and Communication (ICCMC),* pp. 693-699, 2017.

[124] G. A. Issac and A. J. Mary, "Validation Scheme for VANET."," *In IEEE 2nd International Conference on Signal Processing and Communication (ICSPC),* pp. 11-15, 2019.

[125] K. Zaidi, M. Milos B., R. Veselin, N. Arumugam and R. Muttukrishnan, "Host-based intrusion detection for VANETs: A statistical approach to rogue node detection," *IEEE transactions on vehicular technology,* vol. 65, no. 8, 2015.

[126] M. Fogue, M. Francisco J., G. Piedad, F. Marco, C. Carla-Fabiana, C. Claudio, C. Juan-Carlos, C. Carlos T. and M. Pietro, "Securing warning message dissemination in VANETs

using cooperative neighbor position verification," *IEEE Transactions on Vehicular Technology,* vol. 64, no. 6, pp. 2538-2550, 2015.

[127] H. Fernández, L. Rubio, V. Rodrigo-Peñarrocha and J. Reig, "Path loss characterization for vehicular communications at 700 MHz and 5.9 GHz under LOS and NLOS conditions.," *IEEE Antennas and Wireless Propagation Letters,* vol. 13, pp. 931-934, 2014.

[128] M. Giordani, T. Shimizu, A. Zanella, T. Higuchi, O. Altintas and M. Zorzi, "Path loss models for V2V mmWave communication: performance evaluation and open challenges," *IEEE 2nd Connected and Automated Vehicles Symposium (CAVS),* pp. 1-5, 2019.

[129] Y. Yao, B. Xiao, G. Wu, X. Liu, Z. Yu, K. Zhang and X. Zhou, "2017, June. Voiceprint: A novel Sybil attack detection method based on RSSI for VANETs.," *In 47th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN) ,* pp. 591-602, 2017.

[130] R. He, M. Andreas F., T. Fredrik, Z. Zhangdui, A. Bo and Z. Tingting, "Vehicle-to-vehicle propagation models with large vehicle obstructions," *IEEE Transactions on Intelligent Transportation Systems ,* vol. 15, no. 5, pp. 2237-2248, 2014.

[131] T. Abbas, S. Katrin, K. Johan and T. Fredrik, "A measurement based shadow fading model for vehicle-to-vehicle network simulations," *International journal of antennas and propagation ,* 2015.

[132] Y. Yao, B. Xiao, G. Wu, X. Liu, Z. Yu, K. Zhang and X. Zhou, " Multi-channel based Sybil attack detection in vehicular ad hoc networks using RSSI.," *IEEE Transactions on Mobile Computing,* vol. 18, no. 2, pp. 362-375, 2018.

[133] Y. Yao, B. Xiao, G. Yang, Y. Hu, L. Wang and X. Zhou, "Power control identification: A novel sybil attack detection scheme in vanets using rssi.," *IEEE Journal on Selected Areas in Communications,* vol. 37, no. 11, pp. 2588-2602, 2019.

[134] M. T. Garip, P. H. Kim, P. Reiher and M. Gerla., ""INTERLOC: An interference-aware RSSI-based localization and Sybil attack detection mechanism for vehicular ad hoc networks."," *14th IEEE Annual Consumer communications & networking conference (CCNC),* pp. 1-6, 2017.

[135] W. Ahmad, S. Ahmed, N. Sheeraz, A. Khan, A. Ishtiaq and M. Saba, " Localization Error Computation for RSSI Based Positioning System in VANETs.," *In IEEE International Conference on Advances in the Emerging Computing Technologies (AECT) ,* pp. 1-6, 2019.

[136] M. S. Grewal and A. Angus P., Kalman filtering: Theory and Practice with MATLAB., John Wiley & Sons, , 2014.

[137] K. F. Haque, A. Ahmed, Y. Venkata P. and Y. Kumar, "A LoRa Based Reliable and Low Power Vehicle to Everything (V2X) Communication Architecture," *In IEEE International Symposium on Smart Electronic Systems (iSES)(Formely iNiS),* pp. 177-182, 2020.

[138] R. Parker and V. Shahrokh, "Vehicular node localization using received-signal-strength indicator," *IEEE Transactions on Vehicular Technology,* vol. 56, no. 6, pp. 3371-3380, 2007.

[139] N. Alam, T. B. Asghar and D. Andrew G., "Relative positioning enhancement in VANETs: A tight integration approach.," *IEEE Transactions on Intelligent Transportation Systems ,* vol. 14, no. 1, pp. 47-55, 2012.

[140] N. Alam, K. Allison and D. Andrew G., "An INS-aided tight integration approach for relative positioning enhancement in VANETs," *IEEE Transactions on Intelligent Transportation Systems ,* vol. 4, no. 2013, p. 14, 1992-1996.

[141] R. W. Schafer, "What is a Savitzky-Golay filter?[lecture notes]," *IEEE Signal processing magazine,* vol. 28, no. 4, pp. 111-117, 2011.

[142] J. Chen, J. Per, T. Masayuki, Z. Gu, M. Bunkei and E. Lars, "A simple method for reconstructing a high-quality NDVI time-series data set based on the Savitzky–Golay filter.," *Remote sensing of Environment,* vol. 91, no. 3-4, pp. 332-344, 2004.

[143] M. Kamal, S. Gautam and T. Muhammad, "Blockchain-based lightweight and secured v2v communication in the internet of vehicles," *IEEE Transactions on Intelligent Transportation Systems,* vol. 22, no. 7, pp. 3997-4004, 2020.

[144] C. Wang, Z. Likun, G. Liangyi, Z. Zhentang, Y. Lei, L. Zheli and C. Xiaochun, "Accurate sybil attack detection based on fine-grained physical channel information.","* Sensors ,* vol. 3, p. 878, 2018.

[145] M. Boban, J. Barros and O. K. Tonguz, "Geometry-based vehicle-to-vehicle channel modeling for large-scale simulation," *IEEE Transactions on Vehicular Technology,* vol. 63, no. 9, pp. 4146-4164., 2014.

[146] Z. H. Mir, "Assessing the impact of realistic simulation environment on vehicular communications," *In IEEE Fifth HCT Information Technology Trends (ITT),* pp. 312-317, 2018.

[147] Z. H. Mir and F. Fethi, "Simulation and performance evaluation of vehicle-to-vehicle (V2V) propagation model in urban environment.," *In IEEE 7th International Conference on Intelligent Systems, Modelling and Simulation (ISMS),* pp. 394-399, 2016.

[148] H. Van Der, W. Rens, L. Thomas and K. Frank, "Veremi: A dataset for comparable evaluation of misbehavior detection in vanets," *In Security and Privacy in*

Communication Networks: 14th International Conference, SecureComm 2018, Singapore. Springer International Publishing, pp. 318-337, 2018.

[149] X. Liu, Y. Lily, A. Ignacio, S. Kathiravetpillai, M. Arvind, O. Fabian, B. Cornelius, S. Manoj and G. B. Leonardo, "MISO-V: Misbehavior detection for collective perception services in vehicular communications," *In 2021 IEEE Intelligent Vehicles Symposium (IV),* pp. 369-376, 2021.

[150] M. Zhao, W. Jesse and W. Chieh-Chih, "Challenges and opportunities for securing intelligent transportation system.," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems ,* vol. 3, no. 1, pp. 96-105, 2013.

[151] R. W. Heijden, L. Thomas and K. Frank, "Veremi: A dataset for comparable evaluation of misbehavior detection in vanets.," *In International Conference on Security and Privacy in Communication Systems Springer, Cham,* pp. 318-337, 2018.

[152] J. Kamel, R. A. Mohammad, P. A. K. Jonathan, B. J. Ines and P. Urien, "Simulation framework for misbehavior detection in vehicular networks.," *IEEE transactions on vehicular technology,* vol. 69, no. 6, pp. 6631-6643, 2020.

[153] A. Sharma and J. Arunita, "Machine learning approach for detecting location spoofing in VANET," *In IEEE nternational Conference on Computer Communications and Networks (ICCCN),* pp. 1-6, 2021.

[154] W. Hathal, C. Haitham, S. Zhili and C. Maple., "Certificateless and lightweight authentication scheme for vehicular communication networks.," *IEEE Transactions on Vehicular Technology ,* vol. 69, no. 12, pp. 16110-16125, 2020.

[155] S. Abdelnaby, A. E.-G. Mohamed A. and K. HyungWon, "Time Efficient Lightweight Authentication Protocol for V2X Networks."," *한국통신학회 학술대회논문집 (2018): 443-444.,* pp. 443-444, 2018.

[156] Q. Yang, J. Zheng and L. Shen, "Modeling and performance analysis of periodic broadcast in vehicular ad hoc networks.," *In IEEE Global Telecommunications Conference-GLOBECOM.,* pp. 1-5, 2011.

[157] M. J. Khabbaz, W. Fawaz and C. M. Assi, "Modeling and delay analysis of intermittently connected roadside communication networks," *IEEE Transactions on Vehicular Technology ,* vol. 6, no. 61, pp. 2698-2706., 2012.

[158] P. Cincilla, O. Hicham and C. Benoit, "Vehicular PKI scalability-consistency trade-offs in large scale distributed scenarios.," in *IEEE Vehicular Networking Conference (VNC)*, 2016.

[159] X. Lin, X. Sun, W. Xiaoyu, Z. Chenxi, H. Pin-Han and S. Xuemin, "TSVC: Timed efficient and secure vehicular communications with privacy preserving.," *IEEE transactions on wireless communications ,* vol. 7, no. 12, pp. 4987-4998., 2008.