Universidade do Minho
Escola de Engenharia
Departamento de Informática

Mariana Pereira Fernandes

Traceability and Safety Tradeoffs in Modern Vehicles

March 2022

Universidade do Minho
Escola de Engenharia
Departamento de Informática

Mariana Pereira Fernandes

Traceability and Safety Tradeoffs in Modern Vehicles

Master dissertation
Integrated Master's in Informatics Engineering

Dissertation supervised by
José Carlos Bacelar Ferreira Junqueira Almeida
Ali Shoker

March 2022

ACKNOWLEDGEMENTS

I would like to thank my supervisors, José Bacelar and Ali Shoker, for their help and support throughout the making of this dissertation. Without their knowledge and guidance it would not have been possible to develop work we can all be proud of.

To my teachers, with a special mention to everyone in the Informatics Department of the University of Minho, for passing their wisdom onto me, allowing me to become the skilled and resourceful student I am today.

To Professor António Sérgio Oliveira, for challenging me, for defending his students when they need it, and for calling them out when they deserve it. Thank you for showing me the light I did not know I had.

To my family, for supporting me throughout the years, in spite of all the trouble I may have caused.

To my sister, who has always been my shining sun. Thank you for your love and patience. If it were not for the way you care for me, I would not be the individual I am today. Your quenching of my thirst for knowledge nurtured my intellect, allowing me to absorb and understand the world that surrounds us, in a way that lead me to keep wanting to know more about it, is why I owe every single one of my accomplishments to you.

To my beloved boyfriend, for making my world a little brighter everyday.

To my housemates, who heard me out every time I hit a road block or a dead end, and helped me overcome these difficulties.

To my dear friends at Literatuna - Tuna de Letras da Universidade do Minho for their support and words of encouragement in troubled times, and for the shots of celebration at the end of this climb.

To my colleagues, for the camaraderie throughout our engineering degree.

STATEMENT OF INTEGRITY

I hereby declare having conducted this academic work with integrity.

I confirm that I have not used plagiarism or any form of undue use of information or falsification of results along the process leading to its elaboration.

I further declare that I have fully acknowledged the Code of Ethical Conduct of the University of Minho.

ABSTRACT

In this dissertation, the efficiency of privacy protecting mechanisms in short-range vehicular communications, namely Pseudonym Change Strategies, is investigated. To evaluate these strategies, a set of simulation tools is used, that allow for the assessment of several metrics, such as the privacy level obtained and the real pseudonym consumption, resulting from the use of a representative set of pseudonym change strategies. Most importantly, hybrid strategies were considered, which combine schemes that were previously analysed separately. The results show that combining mix-zones with another scheme provides better privacy in most cases. Lastly, we showcase and analyse the problems found in the process of trying to make the simulated scenarios more realistic, which easily comes into conflict with tool limitations and/or subtle and hard to anticipate interactions between different components.

KEYWORDS V2X, privacy, simulation, vehicles.

RESUMO

Nesta dissertação investiga-se a eficácia de mecanismos de protecção da privacidade em comunicações veiculares de curto alcance, nomeadamente recorrendo a Estratégias de Alteração de Pseudónimos. Para a avaliação dessas estratégias, recorre-se a um conjunto de ferramentas de simulação que permitem aferir diferentes métricas, como o nível de privacidade obtido e o consumo efectivo de pseudónimos, decorrentes da utilização de um conjunto representativo de estratégias de alteração de pseudónimos. Mais importante ainda, foram consideradas estratégias híbridas, que combinam esquemas antes analisados separadamente. Os resultados mostram que combinar zonas mistas com outro esquema proporciona melhor privacidade na maioria dos casos. Por último, apresentam-se e analisam-se problemas encontrados no processo de procurar tornar mais realistas os cenários das simulações realizadas, e que facilmente esbarra com limitações das ferramentas e/ou interações subtis e dificilmente antecipáveis de diferentes componentes.

PALAVRAS-CHAVE V2X, privacidade, simulação, veículos

CONTENTS

LIST OF FIGURES

ACRONYMS

AA Authorization Authority.

BSM Basic Safety Message.

CA Certificate Authority.
CAM Cooperative Awareness Message.
CAMP Crash Avoidance Metrics Partnership.
CAPS Context-Aware Privacy Scheme.
CAS Collision Avoidance System.
CRL Certificate Revocation List.

DENM Decentralized Environmental Notification Message.

ECDSA Elliptic Curve Digital Signature Algorithm.
ETSI European Telecommunications Standards Institute.

ITS Intelligent Transport Systems.
IVC Inter-Vehicular Communication.

OBU On-Board Unit.

PCS Pseudonym Change Strategy.
PKI Public Key Infrastructure.
PP Pseudonym Provider.

RA (Pseudonym) Resolution Authority.
RSU Roadside Unit.

SCMS Security Credential Management System.
SUMO Simulation of Urban MObility.

TAPCS Traffic-Aware Pseudonym Change Strategy.

V2D Vehicle-to-Device.
V2I Vehicle-to-Infrastructure.
V2N Vehicle-to-Network.
V2P Vehicle-to-Pedestrian.
V2V Vehicle-to-Vehicle.

V2X Vehicle-to-Everything.
VANET Vehicular Ad-hoc Network.
VID Vehicle ID.
VIN Vehicle Identification Number.

1

INTRODUCTION

Today's life is heavily impacted by the evolution of day-to-day appliances, and with that come not only many advantages, but also an array of disadvantages. Together with the World Wide Web, came the possibility of being connected to a network that allows for easy access and sharing of information. Although this exchange sped up the world's pacing, it sometimes lets bad actors have an easier time creating successful attacks. In some situations, these attacks generate only mild inconveniences, however, the impact they have in critical systems can be catastrophic. Some systems cannot tolerate undergoing downtime, such as financial transaction management systems, while others deal with extremely sensitive information that cannot be disclosed to unauthorized parties, like hospital software.

As a result of advancements in the vehicular communications field, it has become possible to propose new safety features and improve existing ones. However, misuse of the features and/or the systems that support them, can have devastating effects on the safety and privacy of the drivers. Working closely with the automotive industry, Vortex CoLAB has a keen interest in making sure security and privacy demands for smart vehicles are met. As such, they have proposed to research whether or not privacy assuring schemes work as intended.

1.1 MOTIVATION

Motivated by the need to rapidly exchange location information, vehicles were equipped with the functionality of broadcasting beacon messages. While this has obvious advantages, in terms of providing the vehicle with more information about its surrounding environment, it adds well-known security issues, such as tampering or replaying of beacons. In order to mitigate these drawbacks, digital signatures, supported by a vehicular PKI, were proposed. Despite this measure, it is still possible for a bad actor to eavesdrop the beacons and use the information therein to track vehicles.

In favor of privacy, it has been proposed that messages be signed with a special type of certificate that does not contain identifying information about the driver or vehicle. These special certificates, named pseudonyms, need to be rotated periodically, so as to preserve privacy effectively. There are several approaches for pseudonym rotation, that aim to be effective, practical and scalable. Each Pseudonym Change Strategy has different characteristics, which makes them more or less suitable in different circumstances. It is not yet clear which PCS should be employed, nor what amount of pseudonyms are required in order to obtain an adequate level of privacy. In the next section, we outline how these unanswered questions were translated into practical research.

1.2 OBJECTIVES

Our primary objective was to analyze different PCSs, not only to confirm or deny the affirmations found in the literature, but also to add new, in-depth knowledge about the problem at hand.

Throughout the making of this dissertation, we defined smaller, more attainable objectives that would ultimately aid in the pursuit of our main goal. Firstly, it was imperative to become comfortable with the research topic, as it was something never explored in the curriculum. With this in mind, we did extensive literature research for new PCS proposals and comparisons. Secondly, we discovered a need for a method of testing and comparing PCSs, which resulted in designing and executing V2X simulations. After having a method for evaluating PCSs, we needed to get acquainted with the simulation tools chosen. Finally, we had to analyze the results and draw conclusions on the matter.

1.3 DISSERTATION OUTLINE

In addition to this first introductory chapter, the thesis is composed of five more chapters.

In Chapter 2, *State of the Art*, we present an overview of the state of the art by introducing vehicular communications and V2X applications, exploring the subject of privacy in vehicular communications, thoroughly explaining the pseudonym lifecycle and pseudonym change strategies. We also mention why other cryptographic schemes are not, currently, a viable solution.

In Chapter 3, *Evaluating Privacy in Pseudonym Changing Schemes*, we feature the tools used to simulate V2X scenarios and explain the existing metrics for privacy evaluation. Afterwards, we realized the simulated scenarios and display the results of said simulations.

In Chapter 4, *Hybrid Schemes*, we hypothesize that combining PCSs could provide better privacy than using a single PCS. In the interest of replication, we describe the steps to set up these novel scenarios, and present the results following that.

In Chapter 5, *Acquired Knowledge*, we intend to provide more realistic simulations in order to further test the PCSs. We explain the reasoning behind each scenario tested and the necessary changes to the configurations. Subsequently, we reveal challenges with the setup, as well as the significant problems encountered while running the realistic scenarios. In the end, we revisit the results collected, comparing them to previously obtained ones, taking into careful consideration how the aforementioned problems affected the simulations and therefore, the results.

In Chapter 6, *Conclusions and Future Work*, we summarize and reflect on the research done and review the proposed objectives. Finally, we make recommendations for future work on PCSs.

2

STATE OF THE ART

2.1 SMART VEHICLES AND VEHICULAR COMMUNICATIONS

Technology advancements have influenced many sectors, such as the automotive industry, which is improving existing technologies and introducing new concepts, such as artificial intelligence and the connected car (Morley). Throughout the years, we've seen improvements on fuel efficiency, addition of built-in GPS systems and rear-view cameras, along with many features that have improved the driving experience. Now, further advances, such as self-driving cars, are becoming reliant on the existence of vehicle communications, both with other vehicles and infrastructure.

The sharing of information, such as road conditions and current trajectories, allows a vehicle to perceive its environment beyond the field of view of its on-board sensors, leading to a better situation awareness(Lefèvre et al., 2013). This is achieved with communication between vehicles (V2V), between vehicles and infrastructure (V2I), or Vehicle-to-Everything (V2X), which includes both aforementioned nomenclatures, as well as more types of communications, such as Vehicle-to-Pedestrian (V2P) and Vehicle-to-Device (V2D), among others. V2V is based on vehicles' broadcasting of beacon messages that include the vehicle's position, velocity, acceleration, current time, and some additional information (Paulraj et al.), through Vehicular Ad-hoc Networks (VANETs). Beacon messages are referred as Basic Safety Message (BSM) in the US, and Cooperative Awareness Message (CAM) or Decentralized Environmental Notification Message (DENM) in Europe. This message exchange can greatly improve the safety of vehicles, preventing damage and saving lives by warning the driver to take action, or ultimately, for the car to take action.

In light of the impact vehicle communications have, authentication mechanisms for the messages sent are mandatory, so that receivers can verify that the sender is an authorized vehicle. Without this, an attacker could fabricate or replay messages, which could, in turn, cause dangerous situations where the car behaves in an unwanted way.

In order to identify vehicles and perform message authentication, standardization efforts are following approaches based on asymmetric cryptography, leading to the use of digital certificates using Elliptic Curve Digital Signature Algorithm (ECDSA) to sign messages. Other schemes were proposed, but weren't followed through, either because they had more drawbacks than asymmetrical schemes, performed worse, or were lacking sufficient research to prove their practicality and scalability. An overview of these schemes can be found in section 2.6.

2.2 V2X APPLICATIONS

2.2.1 *Vehicle-to-Vehicle (V2V)*

Vehicle-to-Vehicle (V2V) communications are based on CAM messages (European standard) to share position, speed, velocity or make simple cooperative decisions. V2V allows for cooperative adaptive cruise control, cooperative perception, intersection collision warning, emergency vehicle approaching, cooperative forward collision warning, slow vehicle indication, and overtaking vehicle warning. The vehicle's On-Board Units communicate through Dedicated Short Range Communication (DSRC) wireless(radio) technology. DSRC technology delivers certain features, such as rapid network connectivity, low communication latency, and very secure as well as fast communication for various safety related applications, despite having a coverage limit of about 300 meters "Ghosal and Conti (2019). OBUs are the wireless communication units equipped on vehicles, whereas RSUs are wireless access units located [...] on the road" Union (2020).

2.2.2 *Vehicle-to-Infrastructure (V2I)*

In Vehicle-to-Infrastructure (V2I), vehicles' On-Board Units are directly connected to Roadside Units through DSRC. In addition to the aforementioned applications, V2I includes the broadcast of environmental or infrastructure information such as vulnerable road user warnings, accident zone warnings and weather condition emergency.

2.2.3 *Vehicle-to-Network (V2N)*

Vehicle-to-Network (V2N) was the first type of vehicular communications made available and today's modern connected vehicles are supported with internet connectivity, such as LTE and 5G. This enables the use of applications similar to what can be found in mobile and IoT devices, including Over The Air (OTA) software and firmware updates, platooning, diagnostic, assistance, infotainment, telematics, and access to new or updated high-definition maps.

2.2.4 *Sticky Vehicle-to-Vehicle (Sticky V2V)*

Sticky V2V refers to a particular type of V2V communications between vehicles, mainly trucks, where communication and cooperation is continuous and long lasting, like a moving system. Its main purpose is to facilitate platooning, which in turn has advantages such as improving fuel efficiency, reducing traffic jams and avoiding accidents.

2.2.5 *Vehicle-to-Pedestrian (V2P)*

Vehicle-to-Pedestrian (V2P) communication is done predominantly via mobile phone through VPU (Vulnerable Public User) special messages or CAM. This happens occasionally, for instance, in pedestrian crossings. The main applications are VPU based, such as active roadwork, VRU crossing road, emergency electronic break light, signaling VRU hidden by obstacle, scooter/bicyclist safety with turning vehicle, detection of an animal or pedestrian on a highway, and intelligent traffic lights for all (P2I2V - Pedestrian to Infrastructure to Vehicle).

2.3 PRIVACY IN VEHICULAR COMMUNICATIONS

Privacy is defined in Petit et al. (2015) as *an individual's right to control, edit, manage, and delete information about them[selves] and decide when, how, and to what extent information is communicated to others.* Traditionally, breaching privacy required visualization of the license plate, but V2X could allow for easier remote surveillance if efforts to assure privacy are faulty or non-existent. Securing V2X communications does not guarantee one's privacy, since it would be trivial for an attacker to eavesdrop on the communication channel and perform a traceability attack on the vehicle (Lefèvre et al., 2013), even though the certificates and messages do not include any identifying information (Paulraj et al.). While there is no personal information on the certificates or beacon messages, there is a strong correlation between starting and ending points of a car's journey with the owners home and work locations (Petit et al., 2015). The ability to track a vehicle in such a way is a serious breach of privacy and must be considered when developing vehicle communication systems.
"The ability of a user to use a resource or service without disclosing the user's identity" is the definition of anonymity used by the European Telecommunications Standards Institute (ETSI) (ETSI, 2018b). In the same standard, the definition of pseudonymity used is similar to that of the anonymity, coupled with the ability to prove the user is accountable for the actions done under the use of the pseudonym. In practical terms, a pseudonym is a public-key certificate used to sign V2X messages, which does not include any identity information of the vehicle or user (ETSI, 2018c). Since accountability is a requirement for vehicular communications, pseudonym revocation and identity mapping are available, but only to allowed authorities. Some approaches define that multiple authorities are needed for identity resolution so as to not have the full knowledge of pseudonyms and identities in the power of a single authority. Later, we will discuss how these pseudonyms are managed in a way that allows for privacy protection, and how those pseudonym schemes impact vehicle safety.

2.3.1 *Potential Attacks and Adversaries*

Different types of adversaries exist, each targeting different areas of concern and performing attacks according to their characteristics. Research on the possible adversaries defines the following categories Boualouache et al. (2017):

- *Range*: An adversary's range represents how far they can reach nodes in VANETs. This is divided in two categories, local, where the attacking capabilities are limited by the nodes surrounding the adversary, and global, where attacks can occur in whichever area is wanted.
- *Interaction*: This characteristic represents how the adversary interacts with the network and its nodes. The adversary is considered to be passive when it only collects information, or active if it replays or modifies messages.
- *Authorization*: An adversary can be an authenticated node of the network, making it an internal adversary, or external, who is seen by network members an unauthorized, and, therefore, has more limited attack vectors.

We can also identify the possible threats to vehicular systems:

- Availability
  - Denial of Service
- Integrity
  - Manipulation
  - Masquerade
  - Replay
  - Insert of information
- Confidentiality
  - Eavesdropping
  - Traffic analysis
  - Location tracking
- Authenticity
  - Manipulation
  - Masquerade

For the purpose of this thesis, we want to focus on the adversaries that can break the user's privacy. Following the previous analysis, we conclude that both a global and a local, internal or external passive attacker could eavesdrop on communications with the goal of linking pseudonyms, therefore breaking the user's privacy, and mapping the pseudonyms used to an identity, breaking the user's anonymity.

2.3.2 *Privacy and Pseudonym Requirements in Vehicular Networks*

Vehicular networks have requirements and constraints that must be met, and can be split into three categories: basic system requirements, security requirements and privacy requirements. Schaub et al. (2009) define the following requirements:
Basic system requirements assure appropriate system functionality. VANETs suffer from real-time constraints as information is extremely time-sensitive. This means that both transmitting and processing messages should be done as quickly as possible. VANETs must be highly-available and fault-tolerant and scalable, as safety systems rely on them to function, in addition to supporting vehicle oriented, beacon based communications.
Security requirements aim to protect a system from attacks, which is particularly important in VANETs taking in consideration the amount of harm that can result from successful attempts to jeopardize the system. As mentioned in 2.1, message authentication is required. This includes both sender authentication and message integrity. In this case, sender authentication means it is possible to verify that the sender is an authenticated member of

the network and sender identity is not required. We've already covered how accountability is a requirement so misbehaving vehicles can be held accountable for their actions. Furthermore, non-repudiation comes as an extension of authentication, meaning no sender can refuse being the author of a message sent by them. The credentials used must be short-lived in order to prevent tracking and misuse, as we will discuss in the next section. As with any authentication system, there should be a revocation mechanism for misbehaving nodes.

Privacy requirements differ from security ones, in a sense that they protect different assets and can sometimes be antagonistic. Minimum disclosure defines that information revealed in communication should be done so in a controlled way that keeps the disclosure of personal information to a minimum. In order to achieve privacy, anonymity is required, meaning an attacker cannot distinguish the author of a message from a pool of possible senders. Nevertheless, only conditional anonymity is possible due to the accountability requirement, although only authorized entities may possess the ability to link identities to pseudonyms and such capacity must be distributed between authorities, so that cooperation between several authorities is mandatory. In turn, no unauthorized entity should be able to link items of interest, such as pseudonyms, messages, and identities. In addition, the perfect forward privacy requirement states that identity resolution of a pseudonym must only enable linking of messages sent under that credential and should not provide any information about other pseudonyms owned by the same user.

In figure 1, we present the relations between the different aforementioned requirements. Red arrows show constraining relations whereas green dotted lines express supporting relations. Examining the figure, we can easily notice how constraining relations exist only between requirements of different categories, and supporting relations only occur between requirements of the same category, as they pertain to the same goals. Schaub et al. (2009) conclude that while it is possible to achieve an adequate level of privacy in V2X, meeting these conflicting requirements at once is a challenging venture.

Having explained the requirements for privacy, we can now define the requirements for pseudonyms to safeguard privacy in V2X. Accordingly, pseudonyms should be time-limited and unique. Vehicles must always have a pseudonym available to change to, whether by requesting more or having them stored. For pseudonyms to be effective, other vehicle identifiers, such as the ones used in other communication layers (i.e MAC address) must be changed together with the pseudonym.

2.4 PSEUDONYM LIFECYCLE

In order to authenticate the messages sent by vehicles, there was a need to define a Public Key Infrastructure (PKI) that would specify the procedures for creating, distributing, and revoking digital certificates. Traditional PKIs have different requirements from that of vehicular PKIs, such as privacy, and this generated a need for differences in implementation (for example, the certificates do not contain any identifying information).

Due to the fast growth of the automotive industry, and how cutting-edge vehicles communications are, there are only proposed vehicular PKIs. In Europe, the ETSI produced a document (ETSI, 2018a) specifying a security architecture for Intelligent Transport Systems (ITS) communications.

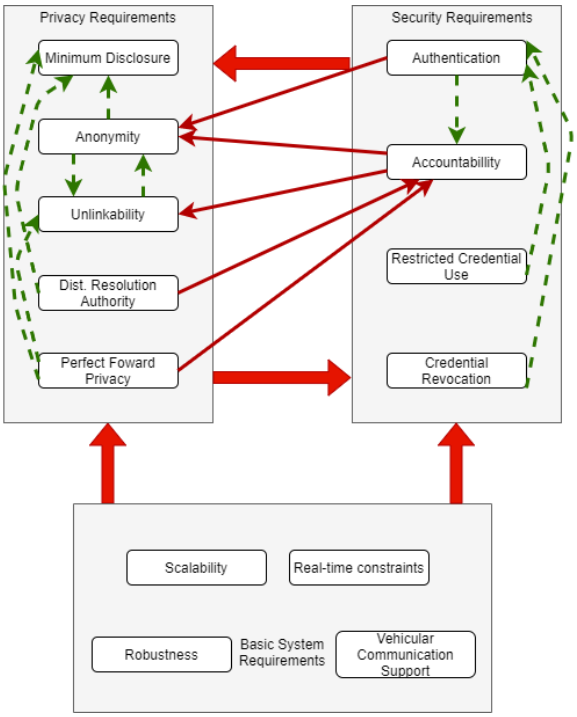

Figure 1: Relations between requirements (Schaub et al., 2009).

Diversely, the Crash Avoidance Metrics Partnership (CAMP) has been working on a Security Credential Management System (SCMS), (Brecht et al., 2018). This system is currently transitioning from research to proof-of-concept. These implementations vary on different levels, but for the purpose of this thesis, we will focus on the certificate provisioning methods and pseudonym change strategies of the european proposal.

Figure 2 shows the pseudonym lifecycle on an asymmetric scheme. From this, we can see the major difference between a traditional PKI and a vehicular PKI lies on the "substitution" of identifying certificates with pseudonym certificates for message signing. Although this may seem a small change, it has a number of consequences that need to be considered when designing how certificates will be issued, managed, verified, and revoked.

Pseudonym Issuance: To begin the process, a vehicle is assigned a Vehicle ID (VID) certificate, by a Certificate Authority (CA) in the moment of its registration. Similar to the Vehicle Identification Number (VIN), which is embossed onto the vehicle chassis by the manufacturer, the VID is a long-term identifier assumed to be pre-installed in a vehicle's On-Board Unit (OBU) (Petit et al., 2015). The vehicle then uses that same VID to request pseudonym certificates to a Pseudonym Provider (PP), or Authorization Authority (AA) , who is then responsible for mapping the vehicle's identity with the pseudonyms used. This raises a question about how many pseudonyms should be issued. While a large number of pseudonyms would be ideal, there are memory constraints that do not allow their storage. This approach would also hamper the revocation process. Too few and the vehicle may run out of pseudonyms before being able to connect with a provider, as it cannot be guaranteed that PPs are always available and in reach.

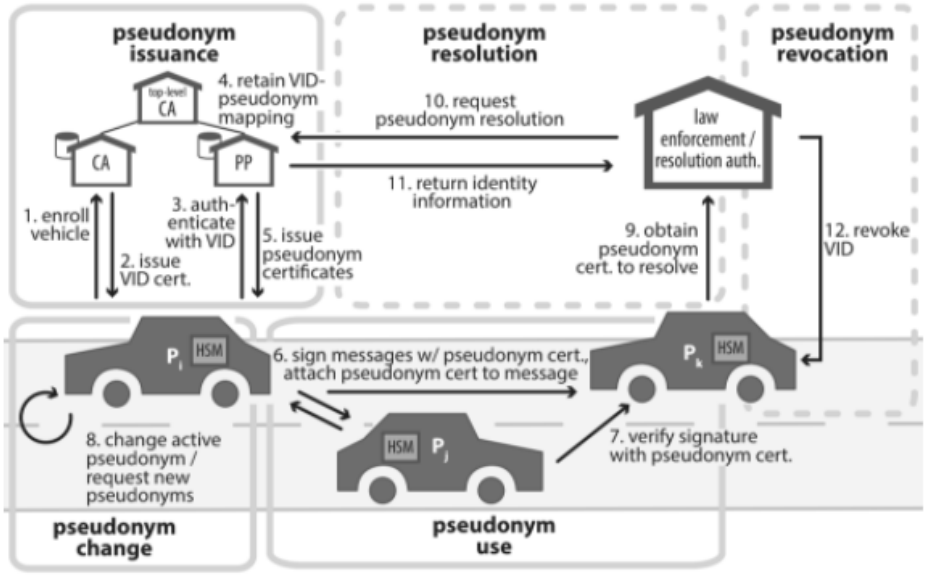

Figure 2: Pseudonym lifecycle for asymmetric cryptography schemes (Petit et al., 2015).

It was also considered that pseudonyms could be self-issued, derived from provided master keys, but these approaches are harder to protect against Sybil attacks. In a Sybil attack, the attacker obtains a large number of pseudonymous identities, and uses them for his/her advantage(Petit et al., 2015; ETSI, 2018c). In VANETs, this could mean simulating a congested path in order to divert vehicles from a target road. To help prevent Sybil attacks, third-party issued certificates are only valid for short periods of time, thus limiting the number of pseudonyms available simultaneously.

Pseudonym Use: The pseudonym certificate is used to sign messages, and is sent together with them. The receiving vehicle then uses the certificate attached to verify the message. Due to network and bandwidth constraints, verification must occur locally. This independence from other services and infrastructure also promotes scalability by keeping communication overhead to a minimum.

Pseudonym Resolution: Pseudonym resolution is necessary in cases where pseudonymity must be undone and as consequence, the identity of a misbehaving node revealed. This procedure must be requested to the (Pseudonym) Resolution Authority (RA), usually by law enforcement. It is proposed that pseudonym resolution capabilities should be restricted, and several authorities would need to cooperate in order to map the pseudonym to the identity. Despite being capable of pseudonym resolution, it is not yet clear if the implemented PKI will be required or forbidden from having this feature (Petit et al., 2015).

Pseudonym Revocation: Traditionally, CAs issue Certificate Revocation Lists (CRLs), with the extensive enumeration of every digital certificate that is still within its time limit, but should no longer be considered valid. Since sending CRLs to every vehicle is unfeasible, pseudonyms are not revoked. Instead, pseudonym providers receive CRLs with the vehicle's VID and will no longer issue more pseudonyms to the vehicle. Vehicles will still be able to communicate pseudonymously, using unexpired certificates in storage, therefore their validity should be set for short periods of time. Consequently, the need to refill pseudonyms will increase.

2.5 PSEUDONYM CHANGE STRATEGIES

Pseudonym change is brought on by the need to cut the association between sent messages, to prevent linking beacon information and, consequently, vehicle tracking. This change must occur on more than one vehicle simultaneously in order to be effective, or it would otherwise be trivial to make the connection between the old and new pseudonym, also known as syntactic linking (Boualouache and Moussaoui, 2017), as shown in Figure 3. Pseudonym change can also be rendered ineffective by semantic linking (Figure 4), which occurs when the attacker is able to link pseudonyms based on the prediction of the position of the vehicle. Such predictions are enable by the frequent broadcast of beacon messages, at a 1-10Hz rate. This rapid rate of beacon broadcasting is required to enable applications such as Forward Collision Warning and Lane Change Warning to function Emara (2017). Therefore, an effective strategy for pseudonym change that does not hamper the safety systems is required to preserve location privacy.

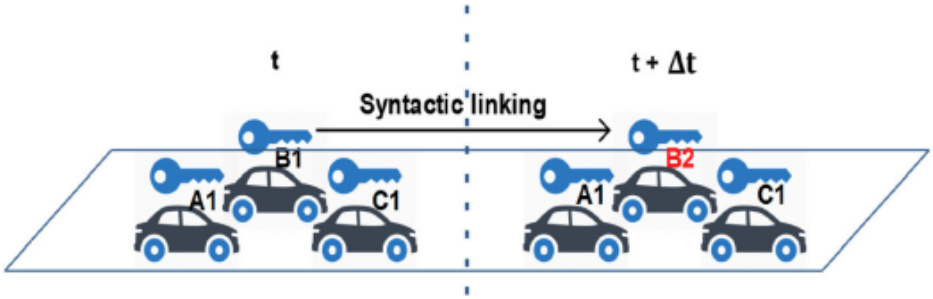

Figure 3: Syntactic Linking (Boualouache and Moussaoui, 2017)

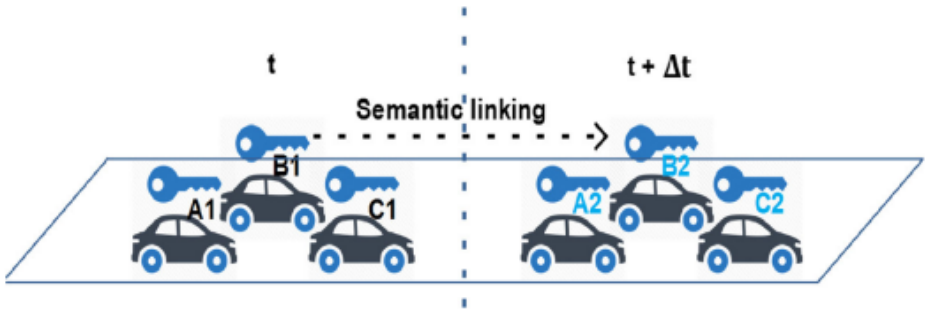

Figure 4: Semantic Linking (Boualouache and Moussaoui, 2017)

- **Fixed Parameter:** A fixed parameter approach is the simplest option, is which a pseudonym is changed based on meters travelled, minutes passed or messages sent. (e.g. change pseudonym every 5 minutes). This strategy is ineffective, as it would be easy for an attacker to assess the parameter value, therefore making tracking trivial (Petit et al., 2015; Lefèvre et al., 2013).

- **Random:** The pseudonym change could happen randomly, however, there is no guarantee that enough vehicles will change pseudonyms simultaneously for this strategy to be effective, because the less vehicles change pseudonyms at the same time, the easier it is to link the old pseudonym to the new one. The simple fact that the change is random could cause vehicles to change pseudonyms too infrequently, or too frequently. Excessive change will deplete the vehicles' pseudonyms without added privacy, while infrequent change may impair privacy.

- **Silent Period:** This approach uses silent periods after a pseudonym change that, in this case, occurs independently from other vehicles and is triggered by speed or direction changes, making movement prediction harder. The silent period strategy dictates vehicles must not send messages, only process incoming information (ETSI, 2018c) for a period of time after changing pseudonyms. This is specially effective in situations where the trajectory is less predictable, such as intersections and makes semantic linking more difficult. Yet, it can compromise safety features (see 2.5.1).

- **SLOW:** To mitigate the impact of silent periods on safety systems, another scheme was proposed (SLOW) which triggers pseudonym change with radio silence when the vehicle's velocity drops below 30 km/h. While this scheme does a much better job at preserving safety, as vehicles going under 30 km/h are unlikely to have to activate the safety systems, it can be argued how well it can preserve privacy in urban and highway environments, where pseudonym change may become too frequent in a congested, urban scenario, or too infrequent in a highway scenario.

- **Cooperative Pseudonym Change Scheme (CPN) :** A density-based strategy defines a minimum number of surrounding vehicles necessary for pseudonym change, thus avoiding useless changes where pseudonym linking would be trivial. This scheme has been proven to be effective but only for dense zones, and may not perform well in a sparse network.

- **Swap:** The possibility that two vehicles following the same route could swap pseudonyms was considered, but ultimately dismissed because it becomes impossible to satisfy accountability requirements and it does not address compatibility issues, such as pseudonym swap between personal and law enforcement vehicles (ETSI, 2018c).

- **Mix-zone:** Several approaches to mix-zones were proposed to address pseudonym change (Boualouache et al., 2017). These zones, preferably high-density, consist of physical locations where no location information is exchanged. Different strategies for pseudonym change inside the mix-zone can be pondered, the more common one being the exchange of "ready" messages between vehicles, signaling they are ready to change pseudonym, or cryptographic mix-zones, where a nearby Roadside Unit (RSU) issues a symmetric key to participating vehicles. This key is used to encrypt all beacon messages inside the mix-zone. Mix-zones have two main disadvantages, one being the possibility that a vehicle may not go through enough, or any, mix-zones in its journey, and the other being the risk of an internal adversary (inside the mix-zone) aiding the external adversary.

- **Context-Aware Privacy Scheme (CAPS):** CAPS determines whether or not a vehicle should change pseudonym based on its context, with the goal of preventing pseudonym change in situations where it would be trivial to link pseudonyms. It does so by using an internal vehicle tracker that helps determine not only when to change pseudonyms, but also the necessary silent period to induce tracker confusion.
- **Traffic-aware (TAPCS):** Vehicles detect when they are in congested traffic, going lower than a certain speed threshold, for a certain period of time. If so, every vehicle willing to change pseudonym in the congested area will enter a silent period, after which it will change pseudonyms.

In spite of the number of solutions researched, the debate of which one should be enforced is unsettled. The answer also depends on whether or not pseudonyms are obtained using a permanent identity or if they can be exchanged, as swapping greatly affects the revocation process, and if pseudonyms can be reused.

2.5.1 *Impact of Privacy Strategies*

While it may seem that security and privacy would go hand in hand, it is not easy to define pseudonym change strategies that can preserve the cars' ability to communicate with other cars and stay aware of their environment. This is key in order to achieve the desired privacy while maintaining their safety features, such as the Collision Avoidance System (CAS). Lefèvre et al. (2013) show how silent periods could prevent a safety system from intervening due to missing beacon information, and propose a context mechanism to calculate when silent periods should be authorized. ETSI (2018c) list existing approaches in literature for pseudonym change.
Frequent pseudonym change does increase privacy, but causes awareness issues in the neighbouring vehicles if done in critical moments. These issues are known as ghost vehicles (old pseudonyms are still present in neighbouring vehicles' systems for a certain amount of time after a change, and can give the impression of having multiple vehicles on the road) and missing vehicles (if vehicle A, that was out of range of vehicle B, enters a silent period after changing pseudonyms and starts emitting beacon messages when very close to vehicle B).
Each key issued comes with a cost, therefore, it is not wise to issue an unnecessary amount of keys. Vehicles are also not capable of holding a large number of keys, as they have limited storage capabilities. In contrast, too few keys force pseudonym reuse, which takes a big toll on privacy, as it makes tracking easier because the attacker will be able to connect the data sent in beacon messages under the same pseudonym.

2.5.2 *Pseudonym Reuse*

Reutilization of pseudonyms is very useful, as vehicles possess limited storage capabilities, and it is thought they will only be issued new pseudonyms every 3 years (Saini et al., 2019). Pseudonym reuse was considered in order to decrease the number of pseudonyms vehicles need to store, and the frequency of pseudonym refills. This means that a vehicle could switch to a pseudonym it has used before but also makes tracking easier by linking space and time data (ETSI, 2018c). Yet, not considering pseudonym reuse at all may not be feasible due to increased memory, network, and financial costs of consuming more pseudonyms. There is no consensus on

whether or not pseudonym reuse should be implemented, as it brings advantages but forces pseudonym change strategies to be redesigned to consider new privacy issues that arise with the reuse.

2.6 OTHER CRYPTOGRAPHIC SCHEMES

Several cryptographic schemes have been proposed (Petit et al., 2015) as an alternative to the aforementioned vehicular PKI, which try to better satisfy certain requirements and mitigate issues that other schemes have. Some of these schemes are based off of an existing cryptography standard, while others combine different approaches, so as to offer a better tradeoff between several constraints, such as network availability and internal storage capacity. We have focused on asymmetric schemes, since they are the preferred scheme for standardization efforts. However, we would like to mention the existence of other proposed schemes, since they too are a component of the state of the art for the matter of this dissertation.

On one hand, there are group signature schemes, in which the group's participants use a shared public key and an individual private key. While this simplifies pseudonym change, and eliminates the need to contact the CA and PP, it raises new issues, such as the election of a group manager to provide the keys and revoke anonymity. Even with several papers published that attempt to mitigate this scheme's biggest flaws, they have failed to create a holistic, practical solution with tested scalability.

On the other hand, identity-based cryptography schemes could also be a solution as they are very similar to asymmetric cryptography, with differences in key generation. However, the computational overhead and revocation issues make this scheme less viable than traditional asymmetric algorithms.

Finally, symmetric cryptographic schemes are also considered as they are more computationally efficient and require less communication overhead than asymmetric schemes, yet they rely heavily on road-side infrastructure for message verification and lack properties provided by public key schemes, such as non-repudiation, which in turn means there is no sender accountability, that is required for VANETs.

2.7 SUMMARY

In this chapter, we covered the progress of the automotive industry and the safety and security challenges associated with modern vehicles, as well as offered a detailed description on vehicular network requirements. We also explored the current efforts to improve on user's privacy.

In the next chapter we aim to use the knowledge acquired from this research to define a concrete problem to tackle within pseudonym change, find ways to evaluate the level of privacy provided and conclude whether or not said level can be considered sufficient.

3

EVALUATING PRIVACY IN PSEUDONYM CHANGING SCHEMES

3.1 SIMULATION TOOLS

In this section, we will explain the software used to simulate V2X networks, and why these simulations are helpful in understanding and evaluating the privacy of PCSs. Firstly, we introduce the network simulator, OMNET++, which serves as base for the ensuing frameworks, INET, VEINS, and SUMO, required to simulate vehicular networks. In its turn, PREXT builds on the functionalities of the aforementioned programs in order to provide a way to simulate PCSs and analyse their efficiency at protecting privacy.

3.1.1 *OMNET++*

OMNeT++ (https://omnetpp.org/intro/) "is an extensible, modular, component-based C++ simulation library and framework, primarily for building network simulators. OMNeT++ can be used in various domains, such as modeling wired and wireless networks, protocol modeling, modeling of queueing networks, validating of hardware architectures, and evaluating performance aspects of complex software systems. In general, it is capable of modeling and simulating any system where the discrete event approach is suitable, and can be conveniently mapped into entities communicating by exchanging messages." The documentation (Varga) explains that "OMNeT++ itself is not a simulator of anything concrete, but rather provides infrastructure and tools for writing simulations. One of the fundamental ingredients of this infrastructure is a component architecture for simulation models. Well-written modules are truly reusable, and can be combined in various ways like LEGO blocks.". Additionally, "OMNeT++ simulations can be run under various user interfaces." (Varga). Graphical, animated, user interfaces such as the one in figure 6 are "highly useful for demonstration and debugging purposes" (Varga), as it is possible to see the elements of the simulation (e.g.: nodes in bottom right corner), the messages being sent and the event log (bottom). On the other hand, "command-line user interfaces are better for batch execution." (Varga). The OMNET++ built-in editor is shown in figure 5, showing the project explorer on the left, the file being edited in the middle, a toolbar for configurations and simulation running, and a console on the bottom which gives verbose outputs of the compilation of the simulation project. Figure 6 shows the graphical environment, with a toolbar on the top that allows for adjustments to the simulation speed, amongst other settings, a graphical representation of

the RSU and nodes, which are close together due to a traffic accident, forming a traffic jam (this is VEINS sample simulation), and at the bottom is the log of the simulation.

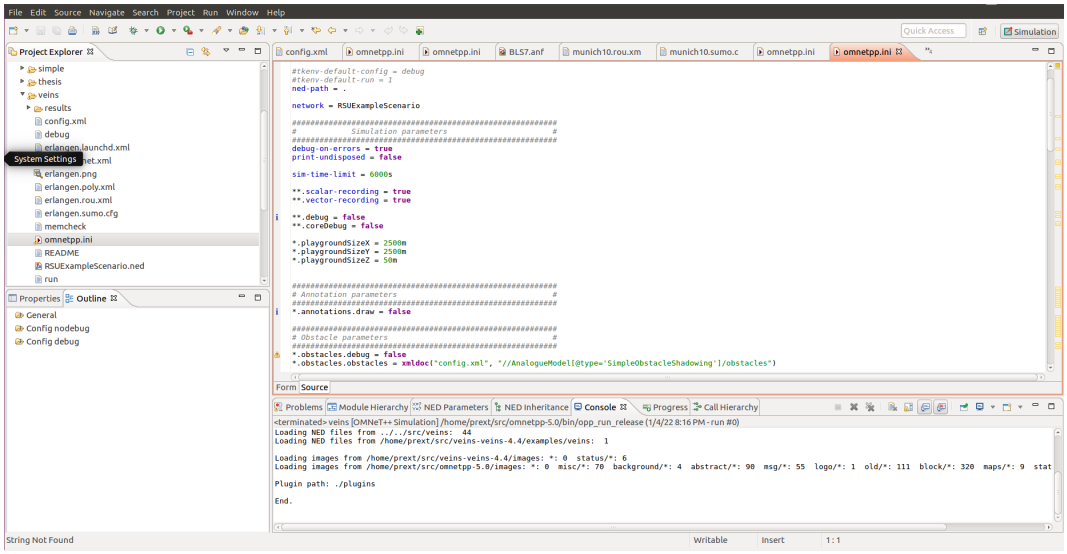

Figure 5: OMNeT++ Editor

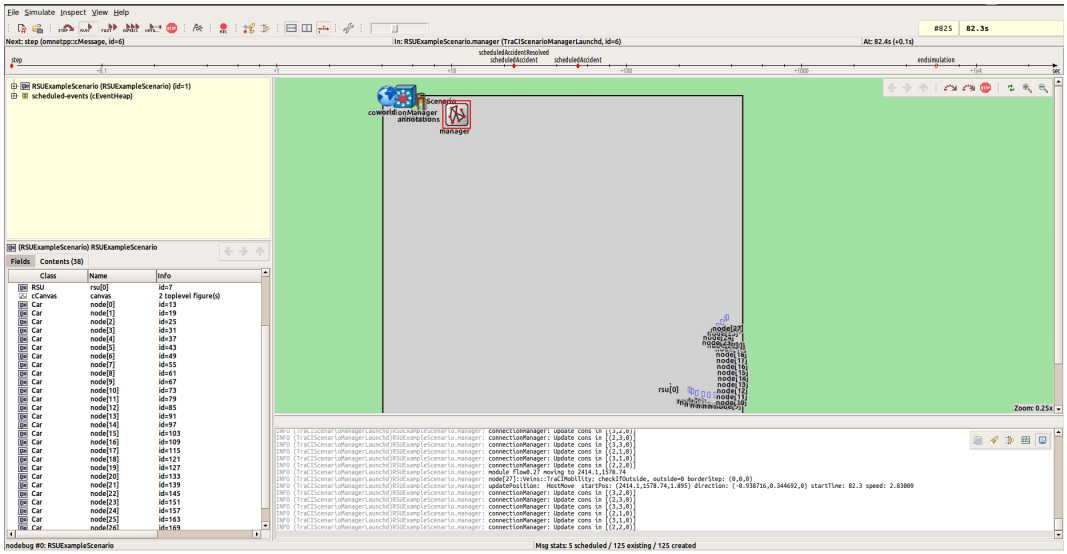

Figure 6: OMNeT++ Graphical Simulation Environment (Tkenv)

The full set of functionalities and instructions for OMNET++ can be found in the manual, in https://doc.omnetpp.org/omnetpp/manual.

3.1.2 *INET*

According to the user manual OMNET++, "INET is an open-source framework for OMNET++ [that] provides protocols, agents, and other models for simulation. INET is especially useful when designing and validating new protocols, or exploring new or exotic scenarios." The manual states that "INET supports a wide class of communication networks, including wired, wireless, mobile, ad hoc, and sensor networks. It contains models for the Internet stack (TCP, UDP, IPv4, IPv6, OSPF, BGP, etc.), link layer protocols (Ethernet, PPP, IEEE 802.11, various sensor MAC protocols, etc), refined support for the wireless physical layer, MANET routing protocols, DiffServ, MPLS with LDP and RSVP-TE signalling, several application models, and many other protocols and components. It also provides support for node mobility, advanced visualization, network emulation and more."

The aforementioned manual adds, "several other simulation frameworks take INET as a base and extend it into specific directions, such as vehicular networks, overlay/peer-to-peer networks, or LTE. INET is built around the concept of modules that communicate by message passing. Agents and network protocols are represented by components, which can be freely combined to form hosts, routers, switches, and other networking devices. New components can be programmed by the user, and existing components have been written so that they are easy to understand and modify."

Additionally, it is noted that "INET benefits from the infrastructure provided by OMNeT++. Beyond making use of the services provided by the OMNeT++ simulation kernel and library (component model, parameterization, result recording, etc.), this also means that models may be developed, assembled, parameterized, run, and their results evaluated from the comfort of the OMNeT++ Simulation IDE, or from the command line." (OMNET++).

3.1.3 *SUMO*

According to its authors, "Simulation of Urban MObility (SUMO) is an open source traffic simulation package designed to handle large simulation, as well as intermodal scenarios. It comes with a set of tools for scenario creation that enable traffic generation and map editing. It allows to simulate how a given traffic demand, which consists of single vehicles, moves through a given road network." (Lopez et al., 2018). Additionally, "The simulation allows to address a large set of traffic management topics. It is purely microscopic: each vehicle is modelled explicitly, has its own route, and moves individually through the network. Simulations are deterministic by default but there are various options for introducing randomness. Included with SUMO is a wealth of supporting tools which automate core tasks for the creation, execution, and evaluation of traffic simulations, such as network import, route calculations, visualization, and emission calculation. SUMO can be enhanced with custom models and provides various APIs to remotely control the simulation." (Lopez et al., 2018). The map of the city of Munich is represented in figure 7, and in figure 8 we can see the vehicles (yellow triangles) stopped at a red traffic light.

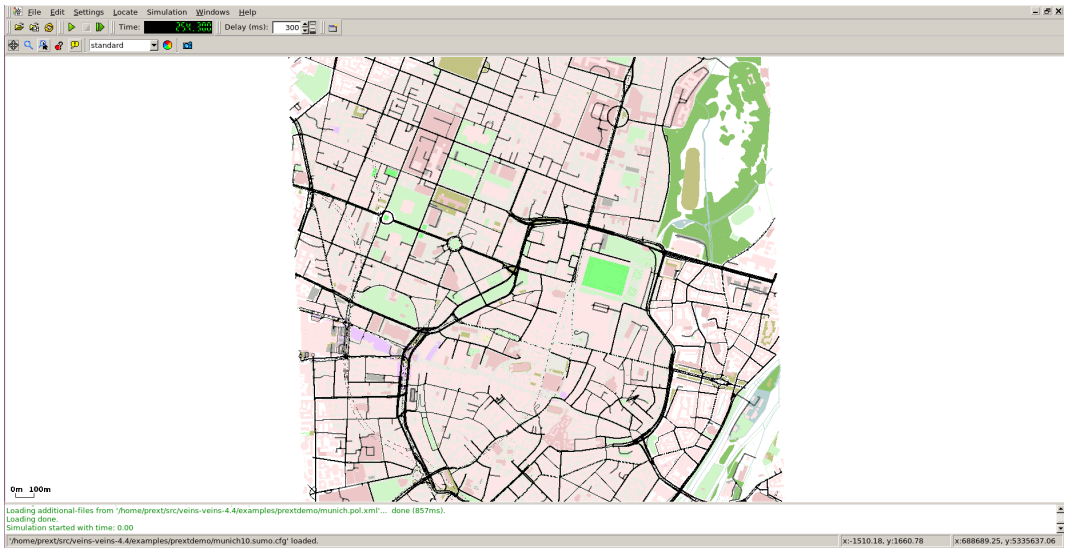

Figure 7: SUMO Graphical Interface - Map of Munich

Figure 8: SUMO Graphical Interface - Map of Munich Zoomed In

3.1.4 *VEINS*

VEINS stands for Vehicles in Network Simulation. It is an open-source framework based on OMNET++ and SUMO, extending them "to offer models for Inter-Vehicular Communication (IVC)." (Sommer et al.). According to the website `veins.car2x.org`, "the Veins framework includes a comprehensive suite of models to make vehicular network simulations as realistic as possible without sacrificing speed. The GUI and IDE of OMNeT++ and SUMO can be used for quickly setting up and interactively running simulations.". The documentation extends this by explaining that "This constitutes a simulation framework. What this means is that Veins is meant to serve

as the basis for writing application-specific simulation code. While it can be used unmodified, with only a few parameters tweaked for a specific use case, it is designed to serve as an execution environment for user written code. Typically, this user written code will be an application to be evaluated by means of a simulation. The framework takes care of the rest: modeling lower protocol layers and node mobility, taking care of setting up the simulation, ensuring its proper execution, and collecting results during and after the simulation." (Sommer et al.).

Furthermore, Sommer et al. states that "Veins contains a large number of simulation models that are applicable to vehicular network simulation in general. Not all of them are needed for every simulation – and, in fact, for some of them it only makes sense to instantiate at most one in any given simulation. The Veins simulation models serve as a toolbox: much of what is needed to build a comprehensive, highly detailed simulation of a vehicular network is already there.". Additionally, "As discussed before, with Veins each simulation is performed by executing two simulators in parallel: OMNeT++ (for network simulation) and SUMO (for road traffic simulation). Both simulators are connected via a TCP socket. The protocol for this communication has been standardized as the Traffic Control Interface (TraCI). This allows for bidirectionally-coupled simulation of road traffic and network traffic. Movement of vehicles in the road traffic simulator SUMO is reflected as movement of nodes in an OMNeT++ simulation. Nodes can then interact with the running road traffic simulation, e.g., to simulate the influence of IVC on road traffic." (Sommer et al.).

The Veins simulation models constitute the current state of the art in vehicular network simulation research (Sommer et al., 2011; Sommer et al.) .

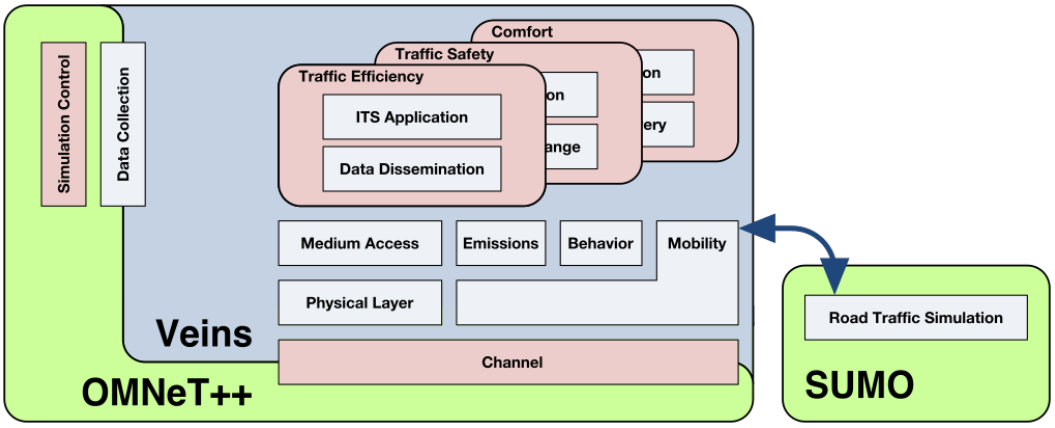

Figure 9: VEINS Architecture - from Sommer et al. (2011)

3.1.5 *PREXT*

PREXT is a framework that provides the means to simulate pseudonym change strategies in VANETS and assess their performance. It has been developed as an extension for VEINS by implementing a privacy layer with different pseudonym changing schemes. It includes a global passive adversary that eavesdrops beacon messages and attempts to reconstruct the journey of the vehicles, together with a vehicle tracker based on the nearest neighbor

probabilistic data association algorithm. PREXT also displays privacy metrics such as traceability, entropy, and anonymity set size, as well as pseudonym statistics, as seen in figure 11. In Emara (2016) words, PREXT consists of the following three main components, as shown in figure 10: "1) A privacy layer inside cars that manages pseudonyms and silence periods and also communicates with mix-zone controllers. 2) A road-side unit controlling a mix-zone by advertising its location and effective range. 3) A global adversary (eavesdropper) who installs several receivers over the road network which in turn report eavesdropped messages to a central vehicle tracker."

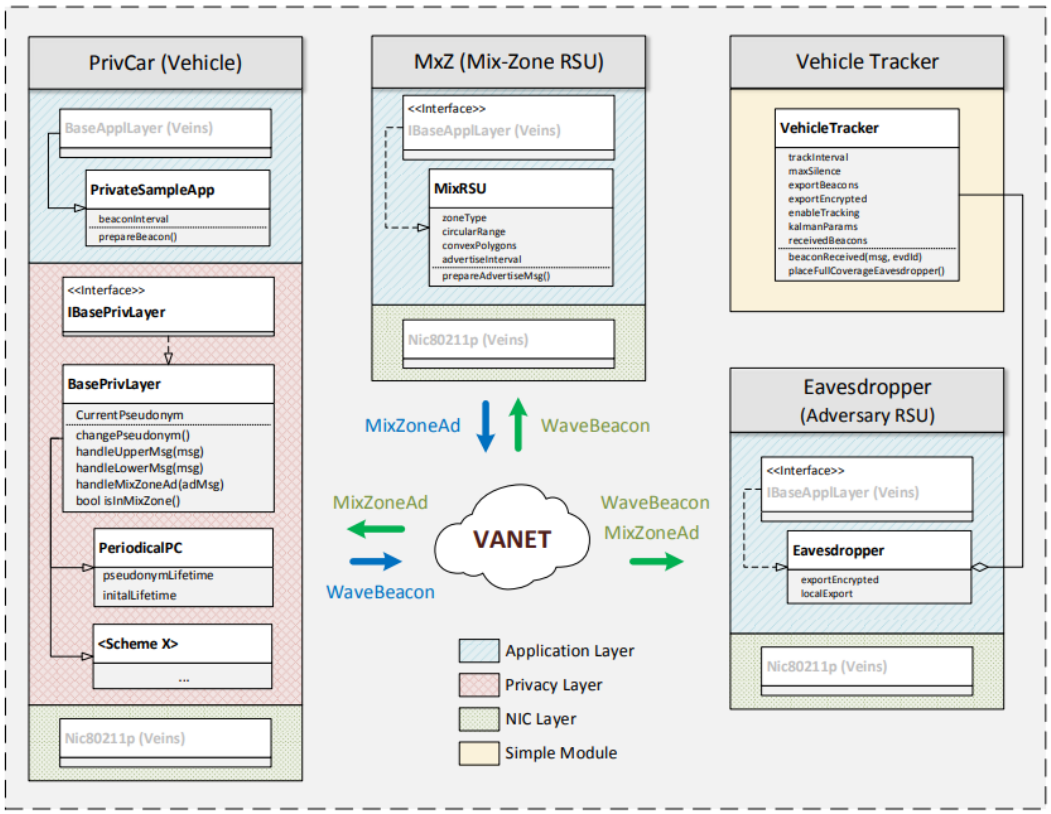

Figure 10: PREXT Architecture - from Emara (2016)

While there are many proposed strategies for pseudonym change, not all of them have been studied enough in regards to practicality, scalability, and most of all, privacy. Some of the strategies present in the literature are different variations in implementation of the same strategy (e.g Mix-zones). PREXT provides 7 PCSs, as stated in Emara (2016), that capture the best available strategies at the time of writing:

- **Periodical Pseudonym Change (PPC)** - Each vehicle changes its pseudonym periodically at fixed or random times, based on the pseudonym lifetime.
- **Random Silent Period (RSP)** - "The random silent period (RSP) scheme allows a vehicle to change its pseudonym after a fixed pseudonym time and keep silent for a uniformly random period within a preset

```
[INFO]  Elapsed Time: 391
[INFO]  Evaluation Time: 0.001584
[INFO]  Number of traces: 169
[INFO]  Number of traces that changed pseudonyms: 143
[INFO]  Traceability %90: 100
[INFO]  Normalized traceability %90: 100
[INFO]  Average max AS size per trace: 0.846154
[INFO]  Average max entropy per trace: 1.16627e-05
[INFO]  Average sum entropy per trace: 1.16627e-05
[INFO]  Average Pseudonym change per trace: 2
[INFO]  Average Confusions per pseudonym change: 0
[INFO]  Average Confusions per trace: 0

End.
```

Figure 11: PREXT Simulation Results

range (e.g., from 3 to 13 s). It can be considered as a special type of mix zones where it is not necessary to place the zone in fixed locations." (Emara, 2016).

- **Coordinated Silent Period (CSP)** - "coordinates all vehicles in the network to remain silent and change pseudonyms synchronously". Since this approach, while appearing effective in theory, cannot be executed in real life, it won't be examined further nor contemplated in our simulations.

- **SLOW** - "In SLOW, a vehicle continuously checks its current speed and broadcasts beacons only when its speed is higher than a preset threshold. If a vehicle does not send beacons for a preset amount of seconds, it changes its pseudonym." (Emara, 2016).

- **Cooperative Pseudonym change scheme (CPN)** - "In CPN, vehicles monitor their neighbors within radius `R` and wait until they reach a threshold `K`. When this trigger occurs, the vehicle sets an internal flag `readyFlag`, broadcasts this flag within the beacon, and changes the pseudonym with the next beacon. When a vehicle receives a beacon with a set `readyFlag` or its internal flag is set already, it changes pseudonym immediately. The CPN class counts neighbors, through received beacons, and checks if the positions are within the configured radius. It also records if any beacon received whose `readyFlag` is set. When a beacon is received from the application layer, CPN changes the pseudonym if the internal `readyFlag` is set or a beacon is received whose `readyFlag` is set since the last sent beacon. It also sets the internal `readyFlag` if the number of counted neighbors exceeds the configured threshold `K`." (Emara, 2016).

- **Context-Aware Privacy Scheme (CAPS)** - The concept "is to determine the appropriate context in which a vehicle should change its pseudonym. This approach aims at increasing the effectiveness of such changes against tracking and avoid wasting pseudonyms in easily traceable situations. Also, it determines the

sufficient silence period that leads to a probable tracker confusion. It employs an internal local vehicle tracker using beacons received by its on-board communication unit." (Emara, 2016).

- **Mix-zones** - "A mix-zone is an unobserved area where an adversary cannot observe messages broadcast by vehicles. It is typically placed at road intersections to make it difficult to predict the vehicle movements. If vehicles would change their pseudonyms within a mix zone, the adversary cannot correlate leaving vehicles with those entering the zone earlier since their messages are hidden and vehicle movements are unpredictable. While in mix-zones, messages would typically be encrypted, in PREXT they are not actually encrypted but the `IsEncrypted` field is set to true, and the messages are excluded from the vehicle tracking process." (Emara, 2016).

Pseudonym Change Strategys are configured in the `omnetpp.ini` file. Here, the several parameters for each PCS are set. In listing 3.1, we see the configuration for the periodical pseudonym change scheme. The vehicles (nodes) are given a PCS type, that regulates the node's behavior. Afterwards, the parameters are set, taking into account that different PCSs have different parameters that need to be set.

```
[Config PeriodicalPsynmChange]
#Periodical pseudonym change every 1 min
*.node[*].privType = "org.car2x.veins.modules.Prext.schemes.PeriodicalPC"
*.node[*].priv.initPsynmLifetime = uniform(0s, 30s)
*.node[*].priv.minPsynmLifetime = 60s
*.node[*].priv.maxPsynmLifetime = 60s
```

Listing 3.1: Periodical Pseudonym Change Configuration

Similarly, mix-zones are configured in the `omnetpp.ini` file. The nodes are set to have a privacy layer, responsible for handling beacons, but no PCS. The mix-zones are then set with the `MixRSU` type, that implements the mix-zone logic. In listing 3.2, mix-zones are set to be circular, with a circular range of 150 meters. Below, we see how the 6 zones are placed by setting the `x` and `z` values.

```
[Config Mix-zone]

*.nMXRSU = 6      #number of mix zones
*.node[*].privType = "org.car2x.veins.modules.Prext.base.BasePrivLayer"
*.MxZ[*].applType = "org.car2x.veins.modules.Prext.base.MixRSU"
*.MxZ[*].appl.debug = false
*.MxZ[*].mobility.z = 1.895
*.MxZ[*].appl.headerLength = 256 bit
*.MxZ[*].appl.dataLength = 100 byte
*.MxZ[*].appl.advertiseInterval = 3s
*.MxZ[*].appl.zoneType = 1 # 1 circular, 2 convex polygon (mix-zone polygons not
    supported yet)
*.MxZ[*].appl.zoneCircularRange = 150 m
*.MxZ[*].appl.zoneConvexPolygons = ""

*.MxZ[0].mobility.x = 780
```

```
*.MxZ[0].mobility.y = 1700

*.MxZ[1].mobility.x = 800
*.MxZ[1].mobility.y = 2500

*.MxZ[2].mobility.x = 1260
*.MxZ[2].mobility.y = 1370

*.MxZ[3].mobility.x = 2050
*.MxZ[3].mobility.y = 1300

*.MxZ[4].mobility.x = 2100
*.MxZ[4].mobility.y = 1925

*.MxZ[5].mobility.x = 1960
*.MxZ[5].mobility.y = 2300

*.tracker.maxSilence = 10s
```

Listing 3.2: Mix-zone Configuration

3.2 PRIVACY METRICS AND PREXT STATISTICS

In order to evaluate the best way to change pseudonyms, several metrics to evaluate privacy have been proposed (ETSI, 2018c; Emara et al., 2016), such as entropy, anonymity set size, and traceability. However, entropy is not an intuitive metric as it uses a logarithmic scale and is unbounded, therefore drawing conclusions from entropy values is difficult, because it can be difficult to relate them to practical privacy implications, as explained in Petit et al. (2015). The same paper claims anonymity set size is a more intuitive metric as it represents the number of vehicles that are indistinguishable. On the other hand, Emara (2017) states that the disadvantage of anonymity set size is that it cannot deal with nonuniform probability distributions of the anonymity set and, therefore, neither entropy nor anonymity set size are suitable metrics to measure privacy.

The traceability metric is thoroughly explained in Emara et al. (2016). In short, it is the ability of the adversary to reconstruct the tracks of the vehicles. According to its definition, the privacy of the driver is considered breached if the adversary is able to track 90% of the route of the vehicle. In PREXT, tracks are assigned to the vehicle traces globally after the tracking process is complete because this approach yields better results than assigning tracks to vehicle traces during the tracking process Emara et al. (2016). The normalized traceability metric is calculated in the same manner as the traceability, but without considering the vehicles that never changed pseudonyms, which are considered trivial to track. Examples of this scenario are cars that entered later in the simulation and didn't change pseudonyms by the time the simulation ended, or cars that never met the requisite for pseudonym change, whether that be speed (in SLOW), number of surrounding vehicles (in CPN), or location (in mix-zones).

As noted before, PREXT (Emara, 2016) includes an adversary module based on the nearest neighbor probabilistic data association (NNPDA) algorithm which is able to track vehicles effectively and efficiently. This adversary is controlled by configuring a type of coverage (full, or otherwise) and object shadowing (how much objects, such as buildings, block communications). The adversary gathers knowledge about the vehicles, which is translated into several privacy metrics such as traceability, entropy, anonymity set size and pseudonyms statistics. PREXT provides privacy metrics and other statistics relevant for simulation evaluation and comparison via console output at the end of the simulations, and more extensively in the results files. These values are documented on Emara (2016) and table 1.

Statistic	Definition
Elapsed Time	Amount of seconds it took to run the simulation.
nPseudonyms	Total number of distinct pseudonyms encountered by the eavesdropper.
nTraces	Total number of vehicles encountered by the tracker.
maxEntropy	An array of the max entropy encountered by each vehicle during the whole simulation.
avgMaxEntropy	The calculated average of the maxEntropy array values
avgMaxEntropy per trace	The calculated average of the maxEntropy array values, divided by nTraces
maxAnonymitySetSize	An array of the max anonymity set size encountered by each vehicle during the whole simulation.
avgMaxAnonymitySetSize	The calculated average of the maxAnonymitySetSize array values.
avgMaxAnonymitySetSize per trace	The calculated average of the maxAnonymitySetSize array values, divided by nTraces.
nTracesChngPsynms	Total number of vehicles encountered by the tracker and changed their pseudonyms at least once.
Traceability90	Traceability metric defined in Emara et al. (2016)
N_Traceability90	Normalized traceability metric
avgSumEntropy per trace	The calculated average of the sumEntropy array values, divided by nTraces
Average Pseudonym Change per trace	Number of detected pseudonym changes divided by nTraces
nTrackerConfusion	A histogram of the number of tracker confusions per vehicle.

Table 1: Statistics and Privacy Metrics

3.3 SIMULATION ENVIRONMENTS

In order to proceed in the simulations, it is required to install the aforementioned applications. PREXT requires OMNET++ 5.0, SUMO 0.25.0 and VEINS 4.4 in order to work. An installation guide can be found in the PREXT

repository here https://github.com/karim-emara/PREXT. In this repository, a virtual machine can be found that already includes all the necessary software in the correct versions, as well as a running demonstration of PREXT, using the city of Munich Germany as an example.

To begin our experimentations, we ran PREXT sample/example simulations, without changing any parameters or configurations. The default parameters are as shown in table 2. For PCSs with pseudonym lifetimes(*), the first pseudonym is set a lifetime between 0 and 30 seconds. A map of the city of Munich is used, in conjunction with a set of 169 vehicles with random routes around the city streets.

Module	Parameter	Value
Veins	Transmission power	20 mW
	Bit rate	18 Mbps
	Thermal noise	-110 dBm
	Packet header length	256 bit
	Beacon payload length	100 byte
	Beacon rate	1 Hz
	Obstacle Shadowing per cut	0.1 dB
	Obstacle Shadowing per meter	0.0001 dB
Tracker	Eavesdropper range	300 m
	Eavesdropper overlap	30 m
	Track interval	1 s
	Full Coverage?	True (default)
General	Simulation Time Limit	300 s
	Initial Pseudonym Lifetime*	[0,30] s

Table 2: Default PREXT Parameters

PCS	Parameter	Value
PPC	Pseudonym Lifetime	60 s
RSP	Pseudonym Lifetime	60 s
	Silent Period	[3,9] s
SLOW	Speed Threshold	8 m/s
	Silent Threshold	5 s
CPN	Radius	100 m
	Neighbor Threshold	2
CAPS	Pseudonym Range	[60,180] s
	Silence Range	[3,13] s
	Missed beacons silence threshold	2 beacons
	Neighborhood radius	50 m
Mix-zone	Advertisement Interval	3 s
	Zone shape	Circular
	Zone range (radius)	150 m
	Number of zones	6

Table 3: PCS Parameters

To begin running the simulations, one must first run the SUMO launcher in the terminal. In the command line, run:

```
~/src/veins-veins-4.4/sumo-launchd.py -vv -c ~/src/sumo-0.25.0/bin/sumo
#or with sumo-gui to see SUMO's graphical interface
~/src/veins-veins-4.4/sumo-launchd.py -vv -c ~/src/sumo-0.25.0/bin/sumo-gui
```

OMNET++ allows for simulations to be run in different environments. For debugging and demonstration purposes, one can use the default graphical environment by pressing "Run Menu > Run" or the green button in the toolbar. In the window that pops up, the desired privacy scheme can be selected. Then one can choose the speed at which to run the simulation by clicking "Run", "Fast" or "Express". The different speeds update the graphical interface less, or not at all, in order to speed the performance. Alternatively, for batch execution, we can use the following command to run the simulations in a command line interface (CLI), after navigating to the folder that contains the `omnetpp.ini` file, in this case, `~/src/veins-veins-4.4/examples/prextdemo`

```
opp_run_release -r 0 -n ../veins/:../../src/veins --image-path=../images -l
    ../../src/veins --debug-on-errors=false -u Cmdenv -c name_of_PCS omnetpp.
    ini
```

3.4 SIMULATION/PREXT RESULTS

In this section, we will present the results of simulating the several PCSs that are provided by PREXT. When the simulation terminates successfully, PREXT prints the results in the environment the simulation was ran on. In figure 11, we can see the results of a simulation that was run in the command line environment. The full results are available under the *results* folder, in the `simulationName.sca` file and the `simulationName.vec` that contain the scalar and vector result values, respectively.

To process the results files, the OMNET++ tutorial for Python's data processing library `pandas` is recommended (`https://docs.omnetpp.org/tutorials/pandas/`). To convert the aforementioned `.sca` files into CSV, we used the OMNET++ `scavetool` command, as shown in listing 3.3. Here, the `x` flag is the option to export the `*.sca` files of the next parameter, into the output file `results.csv`.

```
scavetool x *.sca -o results.csv
```

Listing 3.3: Use of `scavetool`

At this point, the results are in a widely accepted and easy to analyse format. Here, we opted to use a spreadsheet software to facilitate the plotting of the graphs. The traceability and normalized traceability values are represented in Figure 12. In comparison with the results in Emara (2016), we obtained very similar values, with the exceptions of SLOW and CAPS where we have higher traceability and normalized traceability, and mix-zones, where our traceability results were higher, but normalized traceability was around the same. Upon studying these results and reanalyzing the PREXT paper, we realized that the sample simulations we used that come with PREXT are not the exact same as the ones used in the paper. The difference is the simulations we used have a limit of 300 seconds, during which a total of 169 vehicles enter the simulation. PREXT simulations last 10 minutes, where in the first 5 minutes, the same 169 vehicles (supposing they use the traffic file they provide) enter the simulation, and in the last 5 minutes the vehicles simply exit the simulation when their route, established in the traffic file, finishes.

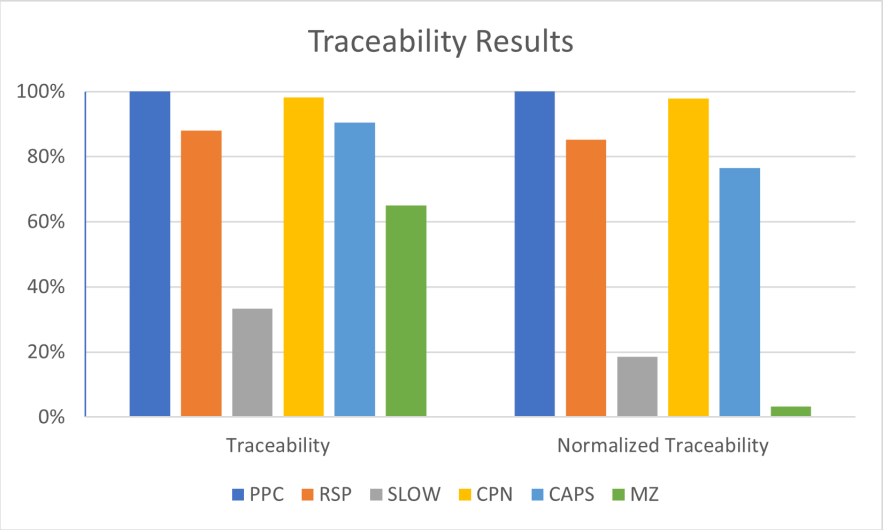

Figure 12: Traceability and Normalized Traceability Graph

In Figure 13, we can see the time it took to run each simulation, in seconds. Most simulations took around 6 minutes to complete, with SLOW being the fastest, needing only 3 minutes to complete. This happens due to SLOW's silent period rule, which in turn causes less beacons to be transmitted, reducing the computational overhead of the scheme, and mix-zones being the slowest, as a consequence of the additional advertisement messages and the vehicles checking their existence within the mix-zone (Emara, 2016).

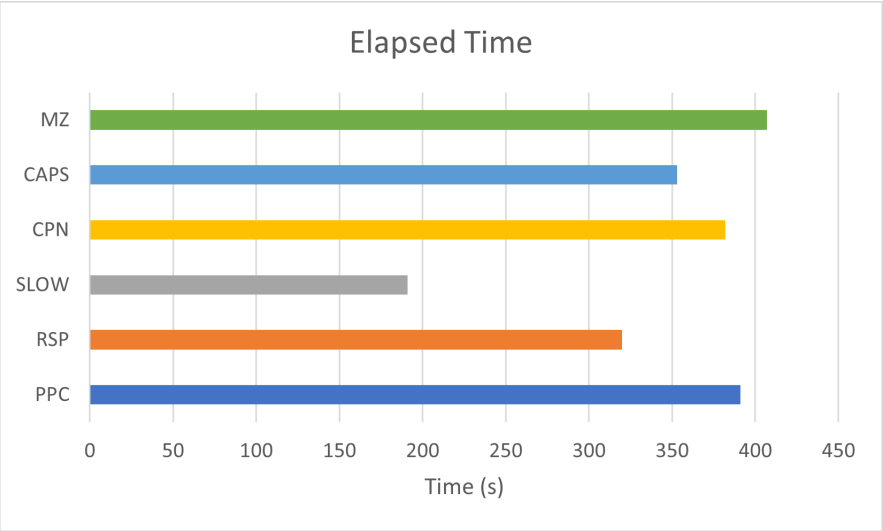

Figure 13: Elapsed Time Graph

In order to get the true pseudonym consumption values, we had to retrieve each vehicles' `psynmTimes::count`, which holds the number of pseudonyms used by the vehicle. The `average pseudonym change per trace` metric only takes into account the vehicles and pseudonyms that the adversary has knowledge of,

not of all the vehicles. We calculated the average and maximum values using the `describe()` function of `pandas`, that calculates these statistics automatically for each PCS. These values are represented in Figure 14, where the columns are the average (left y-axis) and the maximum is represented bt the line (right y-axis). The CPN strategy consumes a significant amount of pseudonyms, with an average consumption per vehicle of almost 36 pseudonyms, and a maximum of 179, as shown in Figure 4. Consequently, CPN use does not seem feasible in real life, unless a way to decrease this consumption is found. Previously, it was revealed that SLOW had the lowest traceability, in contrast, it has the highest pseudonym consumption, following CPN.

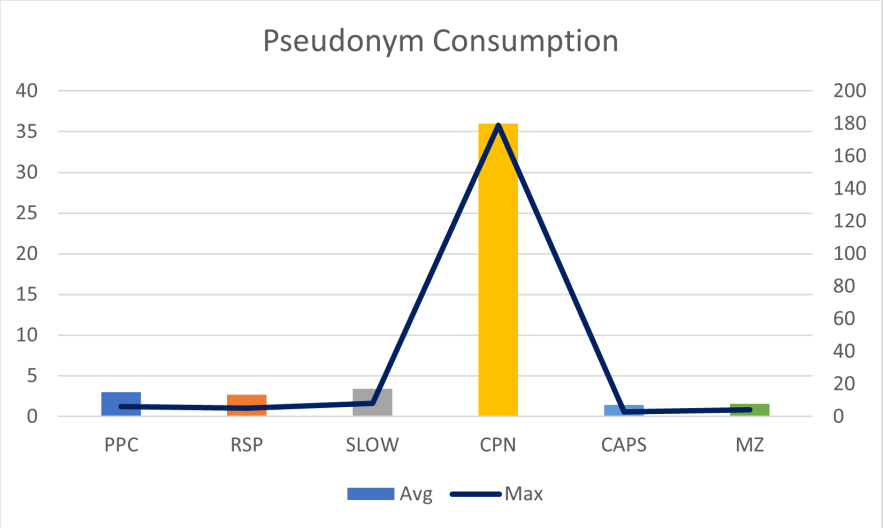

Figure 14: Average and Max Pseudonym Consumption Graph

PCS	Average	Maximum
PPC	3.01	6
RSP	2.67	5
SLOW	3.41	8
CPN	35.98	179
CAPS	1.41	3
MZ	1.57	4

Table 4: Pseudonym Consumption Results

4

HYBRID SCHEMES

4.1 MOTIVATION

From the research done, we conclude that the best pseudonym change strategy and the number of pseudonyms issued and refilled still remain as open questions. Paulraj et al. state the importance of understanding how many certificates a device should hold for a sufficient level of privacy and refer that they "feel" a number around 50 certificates per week will provide a proper level of privacy. Brecht et al. (2018) define a SCMS for V2X that supports a range of pseudonym change strategies in which the number of certificates issued is controlled by the SCMS manager, instead of being defined.

Moving forward, we want to research whether a combined strategy could help provide more privacy in a real world scenario, where the pseudonym refill capacity and frequency are limited. Moreover, we've shown the advantages and disadvantages of the different schemes, having also explored which scenarios are covered by each PCS. For example, a mix-zone works in specific physical locations, such as intersections and SLOW is more suited for urban, and not highway, scenarios. We want to test if having a PCS, as well as mix zones, will increase privacy. The idea behind this is that mix zones have advantages that others don't, but other PCS's cover more scenarios than that of mix-zones. We hypothesized that a hybrid scheme, by combining mix-zones, which have low pseudonym change and low normalized traceability (meaning they provide a lot of privacy for vehicles that enter these zones), with the other strategies, with the objective of improving the privacy outside of the range of mix-zones, could yield better privacy results than the use of a single PCS, without causing a surge in pseudonym consumption.

4.2 SIMULATION SETUP

In this section, we will explain the changes done to the simulations in Chapter 3 in order to obtain a hybrid scheme, composed of one of the PCS provided by PREXT together with mix-zones.

After carefully analyzing PREXT, we can see that the implementation of the PCSs, with the exception of mix-zones, is done through the `PrivCar` module, as shown in figure 10. On the other hand, mix-zones are implemented as a type of RSU. This architecture makes it so it is easy to combine mix-zones with one of the other PCSs, as stated in Emara (2016). In other words, most of the PCSs in PREXT, with the exception of mix-zones,

have their behavior implemented as a module for the vehicle, whilst the mix-zone has its logic come from an RSU module, meaning they work independently and can be easily configured to work simultaneously.

In practice, we only need to combine the vehicle's PCS configuration with the mix-zone configuration, in the `omnetpp.ini` file, with the exception of `*.node[*].privType`, which is defined by the strategy being paired with the mix-zones. The resulting hybrid strategies are as follows:

- **MPPC** - Mix-zones and Periodical Pseudonym Change
- **MRSP** - Mix-zones and Random Silent Period
- **MSLOW** - Mix-zones and SLOW
- **MCPN** - Mix-zones and Cooperative Pseudonym change scheme
- **MCAPS** - Mix-zones and Context-Aware Privacy Scheme

```
[Config MzPPC]
#Periodical Pseudonym Change configuration, as shown before
*.node[*].privType = "org.car2x.veins.modules.Prext.schemes.PeriodicalPC"
*.node[*].priv.initPsynmLifetime = uniform(0s, 30s)
# ...
#Mix-zone configuration, as show before
*.nMXRSU = 6
#No *.node[*].privType
*.MxZ[*].applType = "org.car2x.veins.modules.Prext.base.MixRSU"
*.MxZ[*].appl.debug = false
#...
*.tracker.maxSilence = 10s
```

Listing 4.1: Configuration of MPPC

After configuring the `omnetpp.ini` file with the intended simulations, they were run using the command line environment, as explained in the previous chapter. The parameters of the simulations remained the same as the "simple" simulations (tables 2 and 3).

4.3 RESULTS AND DISCUSSION

In line with what was done in Section 3.4, we took the results of the simulations and processed them, producing the figures seen below.

In Figure 15, we plotted the traceability and normalized traceability results, comparing the simple scenarios of the previous chapter, in blue, with the results obtained with the hybrid scheme, in orange. We also present the mix-zone simple scenario, for comparison. In general, both traceability metrics improve, in comparison with using just PPC, RSP, SLOW, CPN, or CAPS. However, MRSP, MCPN and MCAPS did not perform better than using mix-zones alone. In some of these cases, although traceability decreased slightly, the normalized traceability

metric increased significantly, in comparison with the simple mix-zone scenario. With MPPC and MSLOW, we see an improvement in traceability in contrast with their simple counterparts. Despite the fact that the mix-zone scenario has less normalized traceability than the former scenarios, the traceability results are better in the latter. This signifies that combining mix-zones with PPC or SLOW leads to better privacy protection, because the pseudonym change is more effective.

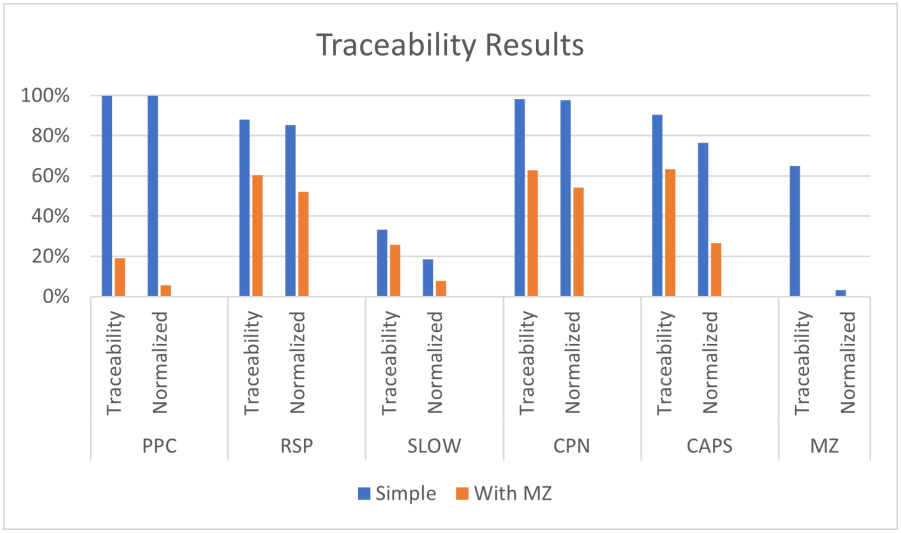

Figure 15: Traceability and Normalized Traceability Graph

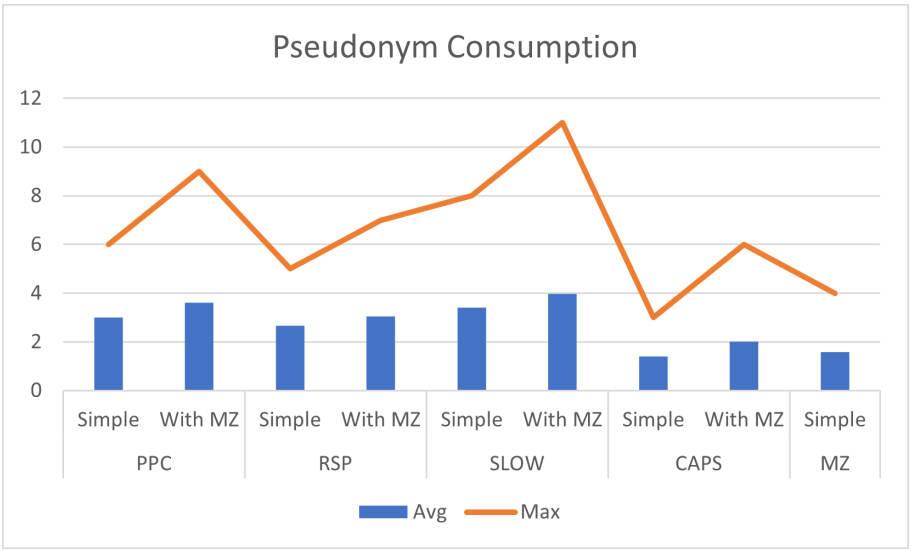

Figure 16: Pseudonym Consumption Graph

It is also possible to conclude that more pseudonyms are consumed by the hybrid schemes, in relation to the simple ones. Both the average and maximum values have risen in these simulations, as shown in Figure 16. The

CPN scenarios were excluded from this graph, in order to prevent the graph from deforming due to an outlier value. The full results for average and maximum pseudonym consumption can be found in Table 5.

PCS	Simple		Hybrid	
	Average	Maximum	Average	Maximum
PPC	3.01	6	3.62	9
RSP	2.67	5	3.04	7
SLOW	3.41	8	3.96	11
CPN	35.98	179	36.56	175
CAPS	1.41	3	2.01	6
MZ	1.57	4	—	—

Table 5: Pseudonym Consumption Results

In terms of the time needed to run each scenario, we see in Figure 17 that there is, in fact, a slight increase in hybrid scenarios, in contrast with the simple ones. This is expected, considering mix-zones was the worst performing scheme. SLOW had the best performance, in line with what was found in Section 3.4.

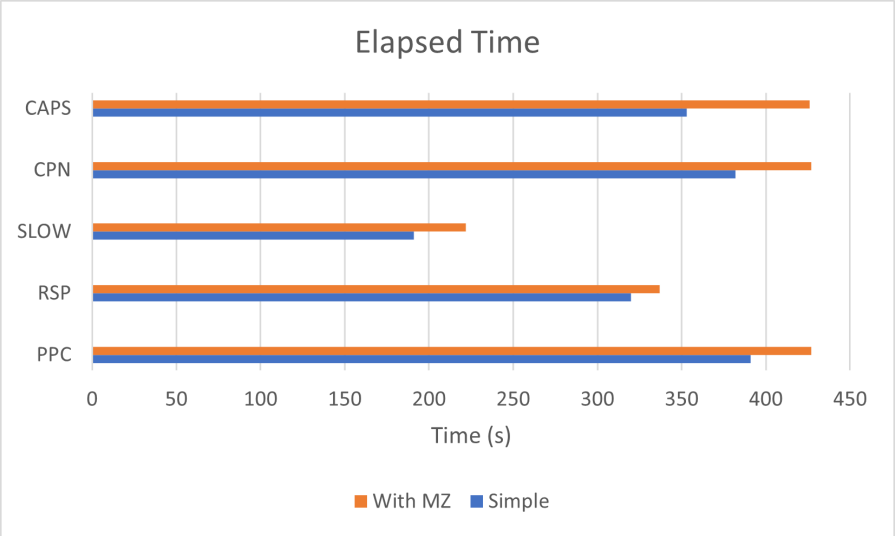

Figure 17: Elapsed Time Graph

5

EXPERIMENTAL CHALLENGES AND LESSONS LEARNT

5.1 TOWARDS A COMPREHENSIVE SET OF EXPERIMENTS

Having completed the first round of experiments, we felt these were somewhat lacking, in terms of their extent and realism. Moving forward with our research, we wanted to understand the impact of changing some parameters, such as pseudonym lifetime, on the privacy levels of the schemes tested previously. Furthermore, the initial simulations had a short length and only one traffic density scenario (one vehicle entering the simulation per second), so it was important to explore what would happen if these simulations were scaled up.

5.1.1 *Simulated Scenarios*

The simulated scenarios are based off the example scenario included in PREXT. Our first goal was to evaluate privacy in a more realistic scenario, so the following changes were made to the configuration of the simulations:

- Simulation time was extended from five minutes to one hour - vehicle traffic was generated to reflect this change.
- Beacon rate was increased from 1Hz to 10Hz
- Object shadowing was rolled back to the default values - as the PREXT paper author suggests.

The duration of the simulation has a direct impact, in the sense that it allows for more vehicles to enter said simulation, improving the chances for different events to happen in traffic, which will, hopefully, give us a better understanding of how well a PCS will behave in the wild. It will also aid in studying the consumption of pseudonyms throughout time, which, in turn, will be key in knowing which strategies are feasible to implement in the real world. The increase of beacon rate is justified by the standards that specify a rate of 1-10 Hz for beacon messages (Petit et al., 2015) and because rapid rate of beacon broadcasting is required to enable applications such as Forward Collision Warning and Lane Change Warning to function (Emara, 2017). Finally, the object shadowing alteration is due to the fact that, in the extended PREXT paper that can be found in the GitHub repository mentioned in Section 3.3, the author suggests reconfiguring the values to realistic ones, albeit there is no mention as to what those values might be. Despite the fact that the paper mentions their obstacle shadowing values are set in negligible values so the adversary can eavesdrop communications without problems, our hypothesis is that if we set normal

obstacle shadowing values, the simulation will be more realistic, as in a real world scenario the attacker would also be affected by this. The reasoning behind these changes is simple, yet the effect caused is unknown at this point.

Afterwards, we analyzed which parameters we would want to change and compare the respective results. Additionally, we wanted to investigate the impact of these changes, both in the simple and the mix-zone scenarios. Furthermore, we wanted to understand the behavior of the PCS in sparse, as well as densely populated scenarios, so a sparse and dense version of each scenario was created. The process of creating these SUMO scenarios is detailed in Section 5.2

We wanted to study the impact of pseudonym lifetime and, at the same time, figure out an optimal configuration where the pseudonym consumption is within realistic values without compromising privacy, so for Periodical Pseudonym Change, we envisioned the following scenarios:

Parameter	Default Scenario	Scenario 1	Scenario 2
`initPsynmLifetime`	(0s,30s)	(30s,60s)	(30s,60s)
`minPsynmLifetime`	60	300s	600s
`maxPsynmLifetime`	60	300s	600s

Table 6: Realistic PPC Parameters

The value specified for the initial pseudonym lifetime parameter is the interval given to the `fuction uniform(lowerbound,upperbound)`, which provides a value from a uniform distribution between the lower and the upper bounds. The initial lifetime values were changed purposefully to randomize the initial pseudonym state, as intended with the default values, while reflecting the longer pseudonym lifetime attributed.

For RSP, we wanted to test not only the impact of the pseudonym lifetime, but also the effect that different silent periods would have on privacy. For this, we fixated the pseudonym lifetime with the same values as PPC, and for each one, we tried two sets of silent periods, taking in consideration the literature, that mentions silent periods must remain as short as possible, otherwise they might impact the safety features powered by V2X communications.

Parameter	Default Scenario	Scenario 1	Scenario 2	Scenario 3	Scenario 4
`initPsynmLifetime`	(1s,30s)	(30s,60s)	(30s,60s)	(30s,60s)	(30s,60s)
`psynmLifetime`	60s	300s	600s	300s	600s
`maxSilentTime`	11s	1s	1s	3s	3s
`minSilentTime`	3s	0.5s	0.5s	2s	2s
`tracker.maxSilence`	11s	1s	1s	3s	3s

Table 7: Realistic RSP Parameters

For SLOW, we decided to maintain the speed threshold as 8 meters per second (30 kilometers per hour), because that's the definition of SLOW, and adjust the silent time threshold, which is how long the car must be silent before it will change pseudonyms. The `tracker.maxSilence` parameter was set to double the silent threshold, as the default simulation was set like this, but there is nothing documenting why this is necessary.

Parameter	Default Scenario	Scenario 1	Scenario 2
silentTimeThreshold	5s	0.5s	1s
speedThreshold	8mps	8mps	8mps
tracker.maxSilence	10s	1s	2s

Table 8: Realistic SLOW Parameters

For CPN, we decided to let the neighbourRadius remain the same, while experimenting with changing the amount of neighbouring vehicles necessary for a pseudonym change to trigger. Our reasoning was that the map used was the same, albeit the rate at which vehicles enter the simulation is also the same, the simulation time is longer, meaning that throughout the simulation, the number of vehicles will increase, so it made sense to require more neighbours, in order to lower pseudonym consumption while maintaining privacy levels. The idea behind it is that more vehicles will increase adversary confusion, as there are more possibilities to which pseudonym each vehicle changed to.

Parameter	Default Scenario	Scenario 1	Scenario 2	Scenario 3
neighbourRadius	100m	100m	100m	100m
kNeighbours	2	2	5	10

Table 9: Realistic CPN Parameters

For CAPS, we wanted to do the same as for PCS and RSP. We were also interested in tweaking other parameters and exploring the results but such experimentations were cut off from the final list of simulations considering the amount of scenarios already planned and the time available.

Parameter	Default Scenario	Scenario 1	Scenario 2	Scenario 3	Scenario 4
initPsynmLifetime	(1s,30s)	(30s,60s)	(30s,60s)	(30s,60s)	(30s,60s)
minPsynmLifetime	60	300s	600s	300s	600s
maxPsynmLifetime	60	600s	900s	600s	900s
maxSilentTime	13s	1s	1s	3s	3s
minSilentTime	3s	0.5s	0.5s	2s	2s
missedBeaconthreshold	2	2	2	2	2
nSilentNeighbours	1	1	1	1	1
neighbourThreshold	50m	50m	50m	50m	50m
tracker.maxSilence	13s	1s	1s	3s	3s

Table 10: Realistic CAPS Parameters

For mix-zones, the set of parameters used was the default one, as we did not see an advantage in changing its parameters, and only having one mix-zone scenario not only made it easier to compare the results, but also significantly reduced the total number of simulations needed. Otherwise, it would have been required to run every scenario multiple times, for the different mix-zones scenarios, much like we must repeat every PCS scenario using the sparse and the dense SUMO scenarios.

In the end, there were a total of 62 scenarios, because every privacy scenario in 11 was run once using the a sparse traffic setup, and again using a dense traffic setup. In this table, each scenario name is the respective PCS name, followed by the scenario number(S1, S2...) and if it also employs mix-zones, it is preceded by the letter "M". The process of generating both traffic scenarios is detailed in Section 5.2.2.

Category	Simple Scenarios	Compound Scenarios
PPC	PPCS1 PPCS2	MPPC1 MPPCS2
RSP	RSPS1 RSPS2 RSPS3 RSPS4	MRSPS1 MRSPS2 MRSPS3 MRSPS4
SLOW	SLOWS1 SLOWS2	MSLOW1 MSLOWS2
CPN	CPNS1 CPNS2 CPNS3	MCPNS1 MCPNS2 MCPNS3
CAPS	CAPSS1 CAPSS2 CAPSS3 RCAPS4	MCAPSS1 MCAPSS2 MCAPSS3 MRCAPS4
MZ	MZS1	—

Table 11: List of Simulated Scenarios

5.2 CHALLENGES IN THE SETUP

In this section, we will detail the aforementioned changes made to the example scenario, in a replicable way. We will begin by detailing the changes done to the configuration files in order to set the desired parameters. Afterwards, we will explain how we generated the sparse and dense traffic scenarios using scripts provided by SUMO.

5.2.1 *Changes in Configurations*

Object shadowing is handled by VEINS, which includes a simple obstacle shadowing model that has been calibrated and validated against real world measurements, as explained in the VEINS documentation: `https://veins.car2x.org/documentation/modules/`. To enable this module, the `ObstacleControl` module must be present and include an appropriate `SimpleObstacleShadowing`. Apart from the `ObstacleControl` module offering the possibility to manually add obstacles to the simulation, the `TraCIScenarioManagerLaunchd` will automatically detect the presence of an `ObstacleControl` module and automatically instantiate one obstacle per polygon of type `building` in SUMO's polygon file, in this case, the `munich.pol.xml` file. The object shadowing property can be found in the `config.xml` file and it was changed to the base values of VEINS sample simulation, mentioned in the documentation. The final result should be the following:

```
<AnalogueModel type="SimpleObstacleShadowing">
  <parameter name="carrierFrequency" type="double" value="5.890e+9"/>
```

```
      <obstacles>
        <type id="building" db-per-cut="9" db-per-meter="0.4" />
      </obstacles>
    </AnalogueModel>
```

Aside from the traffic generation process, which will be detailed in the next subsection, and object shadowing, all of the changes in the configurations are done via the `omnetpp.ini` file. The parameter names and values are detailed in section 5.1.1. The simulation duration was adjusted through setting the `sim-time-limit` to 3600 seconds. In order to change the beacon rate to 10Hz, the `beaconInterval` parameter must be changed to 0.1 seconds, from 1 second.

5.2.2 *Traffic Generation*

Due to the changes in the simulation time, we had to generate the traffic using the scripts SUMO provides for that effect. Firstly, we used the `randomTrips.py` that generates trips for a given network file (map). We specify the duration of the simulation in seconds using the `-e` option, and the output file using the `-o` option. Here, we used the Munich city map that comes with PREXT, and created two trips files, one with one car entering the simulation per second, and the other with two cars entering the simulation per second.

```
    // one car entering the simulation per second
    sumo-0.25.0/tools/randomTrips.py -n munich.net.xml -e 3600 -o munich.trips.
        xml

    // two cars entering the simulation per second
    sumo-0.25.0/tools/randomTrips.py -n munich.net.xml -e 3600 --period 0.5 -o
        munich.trips.xml
```

Listing 5.1: Command to Generate the Trips

```
<?xml version="1.0"?>
<!-- generated on 2021-05-17 12:34:04.567762 by $Id: randomTrips.py 19495
    2015-12-02 12:36:28Z behrisch $  options: -n munich.net.xml -e 3600 -o munich.
    trips.xml-->
<trips>
    <trip id="0" depart="0.00" from="242358060#0" to="156648669#3" />
    <trip id="1" depart="1.00" from="-307308724" to="19524046#58" />
    <trip id="2" depart="2.00" from="310707184#2" to="-151304010#1" />

    ...

    <trip id="3598" depart="3598.00" from="-27145266" to="271888999#0" />
    <trip id="3599" depart="3599.00" from="290407136#39" to="16786171#0" />
```

```
</trips>
```

Listing 5.2: Trips File - munich.trips.xml

The generated file has the start and end of each trip, so we use the `duarouter` script to generate the path between the start and end edges provided by the trips file. Not every trip generated by `randomTrips` will be valid, as there might not exist a valid path from the start edge to the end edge, so the `--ignore-errors` flag was added. The remaining flags were added to prevent cluttering the output of the command, and these flags were the same as in the traffic generation for the PREXT default example.

```
    // Generate the path for each trip in the trips file
    sumo-0.25.0/bin/duarouter -n munich.net.xml --route-files munich.trips.xml -o
         munich2.rou.xml --ignore-errors --no-warnings --no-step-log
```

Listing 5.3: Command to Generate the Routes

```
<?xml version="1.0" encoding="UTF-8"?>
[...]
<routes xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:
    noNamespaceSchemaLocation="http://sumo.dlr.de/xsd/routes_file.xsd">
    <vehicle id="0" depart="0.00">
        <routeDistribution last="0">
            <route cost="178.22" probability="1.00000000" edges="242358060#0
                -242358060#2 123499640 40317919#0 50636516#0 118282031#1
                257527112#0 28338977#0 167443079#0 167443078 167443080#0
                31232317#0 30965375#0 144278190#0 144278190#2 145571054#0
                145571054#1 31159271#2 31159271#3 315220271#0 143927076 72546416#0
                 72546416#3 145628205#0 145628205#1 145628205#2 145628205#3
                145628205#6 316433714#0 145628202#0 27202778#1 146504814 146504815
                 37897187 243452654#0 223891292#0 4673733#0 4673733#1 4673733#2
                4008984 389867038 156648669#0 156648669#2 156648669#3"/>
        </routeDistribution>
    </vehicle>
    <vehicle id="2" depart="2.00">
        <routeDistribution last="0">
            <route cost="113.33" probability="1.00000000" edges="..."/>
        </routeDistribution>
    </vehicle>

    [...]

    <vehicle id="3597" depart="3597.00">
        <routeDistribution last="0">
            <route cost="165.98" probability="1.00000000" edges="-168045901#2
                146808037#4 37342343#3 134941607#0 134941607#3 134941607#6
                134941607#8 134941607#9 40317927 362368395#0 362368395#2
                362368395#3 51476080#0 51476080#2 51476080#5 51508352#0 51508352#2
```

```
                -392088975#1 -392088974#0 -392069508#0 -89840558#4 -89840558#3
               -392088973 -58807376#5 -58807376#2 39321854#0 -47224976"/>
        </routeDistribution>
    </vehicle>
</routes>
```

Listing 5.4: Routes File - munich.rou2.xml

Next, we ran the traffic simulation in SUMO, and realized that sometimes vehicles will have been attributed a route that, while connected in the network file, is not a valid route. Cars that get stuck for 300 seconds are teleported to the next edge of their route. This has an obvious impact on the ability of the adversary to track a vehicle that teleports, so we corrected the routes or deleted the vehicle from the `munich.rou.xml` file that had incorrectly formed routes. Regardless, there were still vehicles being teleported out of the fact that they were stuck in traffic jams for more than 300 seconds, so, after careful analysis, we removed traffic lights `tls3` and `tls8`. This reduced teleportation to around 1% of the total vehicles present in the simulation.

5.3 PROBLEMS

In this section, we will present and explain the problems encountered when running the aforementioned simulations. Owing to these issues, it was impossible to run a part of the set of simulations, and the results of the remaining were unusable/of no value.

5.3.1 *Efficiency and memory limitations*

As mentioned previously, in section 3.1, PREXT offers different environments from which to run the simulations. As shown, all the simulations were run through the command-line interface, because they are run more efficiently, as there is no need to update a graphical interface. Even so, the time needed to complete each simulation was much longer than anticipated, varying between schemes, where SLOW simulations were the fastest, lasting approximately 8 hours, the rest of the schemes taking around 12 hours, and CAPS needing upwards of 17 hours to finish. This made it very difficult to run multiple scenarios with different parameters for the same scheme. There were also attempts to run high density scenarios, with two vehicles entering the simulation each second, but they either crashed from memory shortage or hadn't progressed much in over 24 hours.

We also encountered a significant memory problem while running CAPS simulations. When attempting to run CAPS scenarios, the simulations terminated with a memory error, right before printing the simulation results. This is the moment where the metrics are calculated, as explained in section 3.2. We tried to increase the RAM available for the virtual machine, which worked for the simple CAPS simulations, but not for MCAPS, because it was either given too little RAM to run the simulation, or so much that the host machine wasn't left with enough memory to run the virtual machine. This lead us to believe that there might be a memory leaking problem in PREXT.

These unpredictable, unfortunate and time consuming problems made it impossible for us to produce a set of more realistic simulations, as we intended.

5.3.2 *The Impact of Shadowing*

When analyzing the results, we noticed very low levels of traceability, which seemed promising at first. However, the levels of normalized traceability were null, or close. This raised suspicion, so we decided to create a baseline scenario, with no pseudonym change, which also had questionable traceability and normalized traceability values, as shown in Figure 19. This forced us to go back to the first set of experiments and try to figure which change caused the unusual results. Again, we ran a baseline scenario, and came to the conclusion that the object shadowing was impacting the simulation more than anticipated. It is noted in the PREXT paper that realistic shadowing values should be avoided when using full attacker coverage, but this reasoning is not explained nor explored further. In reality, the eavesdropping stations can end up inside a building when using the adversary in full coverage mode, impacting immensely the ability to receive beacons. In Figure 18 it is represented the amount of vehicles that were detected by the adversary, on average, for each category. The total number of vehicles that were present on the simulation was 2627, but no adversary came close to that value. This can be partly explained by the fact that some cars could have such short journeys that they go undetected. Even so, it is clear that the adversary was unable to detect a large portion of the vehicles in the simulation, even in the baseline scenario. A series of short, baseline simulations were ran with different shadowing values, however, it was not possible to draw any conclusions from these experiments, seeing that there was no clear correlation between the object shadowing values and the results.

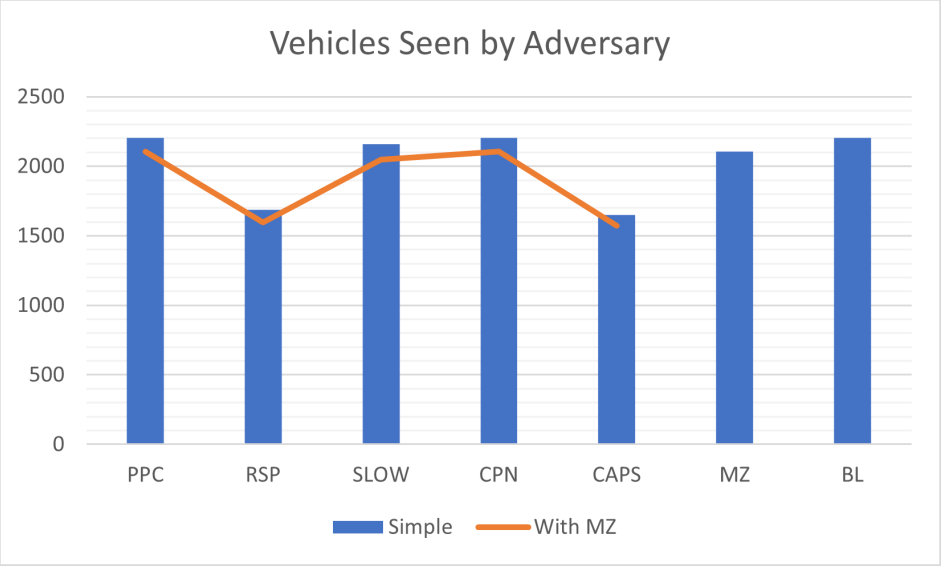

Figure 18: Average Number of Vehicles Seen by the Adversary Graph

5.4 REVISITING RESULTS

As mentioned previously, shadowing affected the simulations more than had been anticipated, and as such, the traceability and normalized traceability results cannot be considered statistically significant. This is further justified by looking at Figure 19, in which the baseline scenario, with no pseudonym change, has lower traceability than most of other scenarios, as a result of the adversary not have been able to receive the beacon messages.

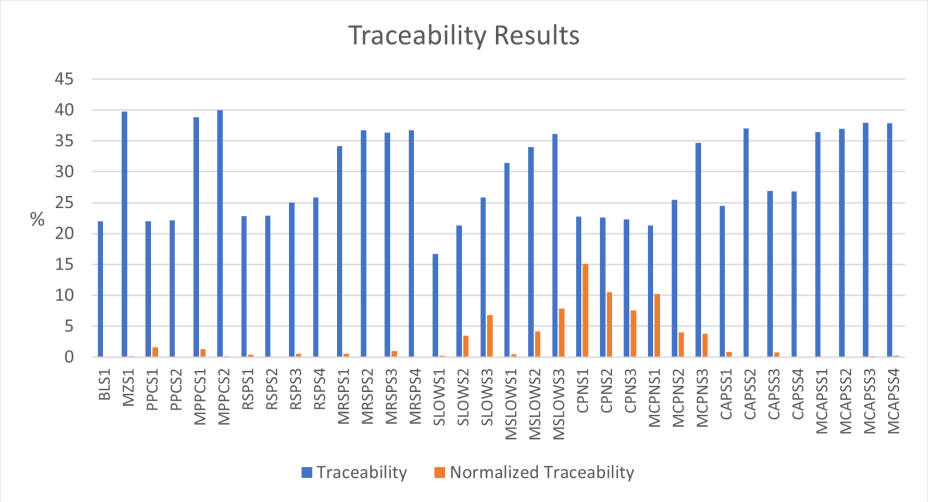

Figure 19: Traceability and Normalized Traceability Graph

The average elapsed time of the different scenarios is depicted in Figure 20. Here, the blue bars represent the average elapsed time of the simple scenarios, for each PCS. The orange line represents the average elapsed time of the compound scenarios, with a PCS strategy and mix-zones. There is only one scenario for `MZ` and one for `BL`, therefore, each of the bars represents the time each simulation took to complete, respectively, instead of representing an average of the category of PCS's, unlike the rest of the results in the graph. The y-axis on the left has a scale in seconds, while the y-axis on the right has a scale in hours, so as to facilitate the comprehension on the graph.

In the image of previous results, `SLOW` was the PCS that took less time to finish, again, due to the lower amount of beacons sent, as a consequence to the silent period below thirty kilometers an hour rule. Due to the issues encountered when running `CAPS`, this category of scenarios had its length shortened to forty five minutes, or two thousand and seven hundred seconds. This gives the impression that these scenarios ran in a reasonable amount of time, when in reality it would take more than seventeen hours for them to reach 99%. Because of the difficulty in running `CAPS` scenarios, and seeing as the simple scenarios were the first to be simulated, there are cases where the compound scenario took less time than its counterpart. We believe that increasing the RAM of the virtual machine to run the `CAPS` scenarios could have an impact in the elapsed time of the simulations that followed. However, seeing as running these simulations is a lengthy process, it was prohibitive to attempt to prove this hypothesis.

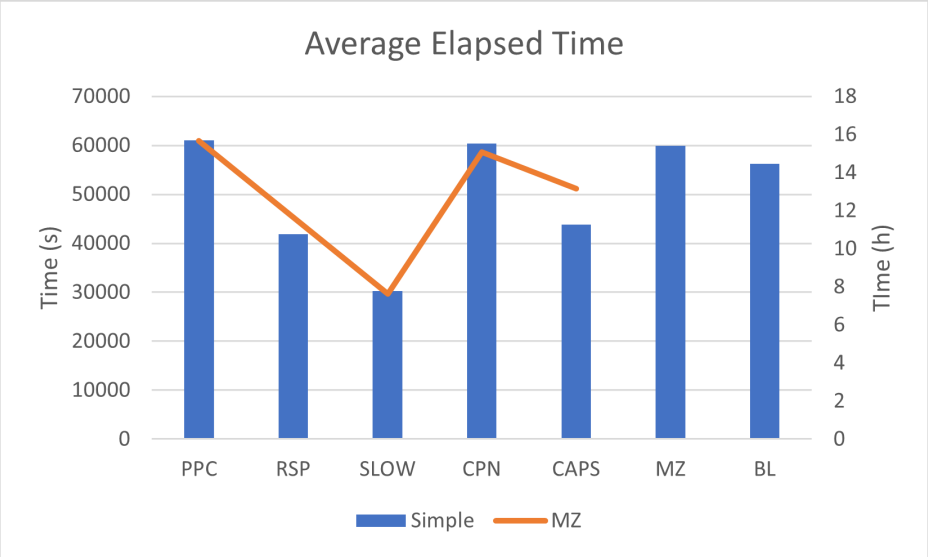

Figure 20: Average Elapsed Time Graph

Considering the aforementioned results and analysis, the pseudonym consumption results must be interpreted with caution. Once again, we left out the results from CPN scenarios, in the interest of preserving the readability of the graph.

Due to the amount of scenarios that were simulated, the average pseudonym consumption for each category of PCS was calculated by averaging said values by category, and is represented in the blue columns. Likewise, the values in orange are obtained by calculating the average of APC grouped by category, for hybrid scenarios. Alike Figure 18, the `MZ` and `BL` scenarios are directly represented in the columns, as there is only one of each.

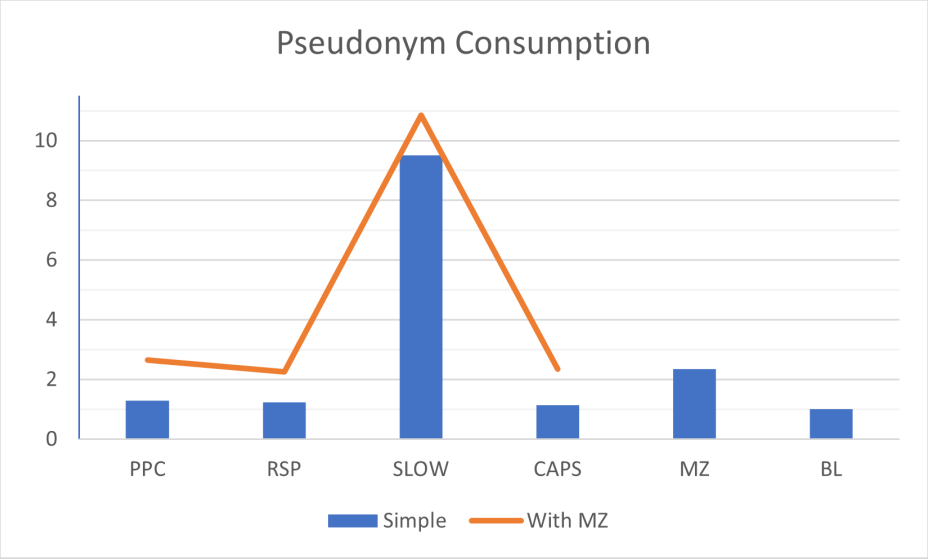

Figure 21: Pseudonym Consumption Graph

6

CONCLUSIONS AND FUTURE WORK

6.1 CONCLUSIONS

First, it must be stated that while the results in Chapter 4 were promising, it would be naïve to say which PCS is decisively the best. While there is some previous research on the topic, extensively presented in Chapter 2, it is often lacking in terms of providing a thorough explanation of the practical steps taken to obtain the results and is often contradictory.

In the scope of this dissertation, we not only described how to run V2X simulations using PREXT in a detailed, tutorial-like manner, but also managed to conduct a novel experimentation by combining mix-zones with other schemes. In addition, we analyzed the PCSs from the standpoints of both privacy and applicability, through the evaluation of pseudonym consumption.

Moreover, we designed a set of realistic scenarios in Chapter 5 with the aim of being able to assert with confidence which PCS is best. In retrospective, this goal was far from attainable, due to the several constraints of vehicular communications, as seen in Chapter 2. Although these experiments were not successful, it was crucial to raise awareness to the lack of larger scale studies and how the adopted tools must be used with caution in those cases. Unfortunately, the time required to run bigger simulations is a heavy constraint, ultimately limiting further research on the matter. In hindsight, it would have been better if we experimented in a more methodical way, using intermediate experiments not only to check if the simulation is working, but also to analyze and confirm the validity of the results.

Finally, we conclude that there is still a large amount of work that should be done before the widespread use of vehicular DSRC in vehicles' communications.

6.2 PROSPECT FOR FUTURE WORK

Having concluded this work, we suggest further investigation in some key topics. We advise that significant efforts in research must be put in place before reaching a decision on which PCSs should be enforced, as current literature often fails to cover different dimensions of the problematic at hand. Additionally, as noted before, there were cases where it was not possible to expand our work due to time constraints.

In the future, a better way to evaluate PCSs, such as a score that takes into account not only several privacy metrics, but also pseudonym consumption and impact on safety features, should be defined. This scoring system would have to be calculated using several preset scenarios that should cover different traffic situations, such as different traffic densities (high vs low) and types (city vs highway). This would provide a fair means of comparison between PCSs, as well as extensively testing schemes in various scenarios.

Likewise, it would be interesting to investigate new PCSs, or new combinations of already existing schemes. Nevertheless, it is imperative to expand on the current research on PCSs in order to confirm their results, and to expose these schemes to more complete, real-life scenarios.

Furthermore, research on PREXT is crucial, both to improve efficiency, which will allow for bigger and more complex simulations to be run, and to increase the amount of PCSs available in the extension. In addition, more knowledge on object shadowing and its impact, on simulations and in real life is also necessary.

Alternatively, more efforts could be made to come up with a different approach to ensure the authenticity and integrity of messages, in a way that does not require pseudonyms at all. Future production of more modern OBUs equipped with faster processing, greater storage, and better network connectivity, as well as infrastructure development, will make new, or currently unfeasible solutions, possible. However, we are aware that this will only be achievable after years of research efforts.

BIBLIOGRAPHY

Abdelwahab Boualouache and Samira Moussaoui. TAPCS: Traffic-aware pseudonym changing strategy for VANETs. *Peer-to-Peer Networking and Applications*, 10(4):1008–1020, 2017. ISSN 19366450. doi: 10.1007/s12083-016-0461-4.

Abdelwahab Boualouache, Sidi-Mohammed Senouci, and Samira Moussaoui. A survey on pseudonym changing strategies for vehicular ad-hoc networks, 2017. URL https://arxiv.org/abs/1704.00679.

Benedikt Brecht, Dean Therriault, Andre Weimerskirch, William Whyte, Virendra Kumar, Thorsten Hehn, and Roy Goudy. A security credential management system for V2X communications. *IEEE Transactions on Intelligent Transportation Systems*, 19(12):3850–3871, 2018. ISSN 15249050. doi: 10.1109/TITS.2018.2797529.

Karim Emara. Poster: Prext: Privacy extension for veins vanet simulator. pages 1–2, 12 2016. doi: 10.1109/VNC.2016.7835979. URL https://github.com/karim-emara/PREXT.

Karim Emara. Safety-aware location privacy in vanet: Evaluation and comparison. *IEEE Transactions on Vehicular Technology*, 66(12):10718–10731, 2017. doi: 10.1109/TVT.2017.2736885.

Karim Emara, Wolfgang Woerndl, and Johann Schlichter. Context-based pseudonym changing scheme for vehicular adhoc networks, 2016.

ETSI. ETSI TS 102 940 V1.3.1 - Security; ITS communications security architecture and security management. Technical report, 2018a.

ETSI. Intelligent Transport Systems (ITS); Security; Trust and Privacy Management. Technical Report 102 941, 2018b. URL http://www.etsi.org/deliver/etsi{_}ts/102900{_}102999/102941/01.01.01{_}60/ts{_}102941v010101p.pdf{%}0Ahttp://files/455/ts{_}102941v010101p.pdf.

ETSI. ETSI TR 103 415 V1.1.1:" Intelligent Transport Systems (ITS); Security; Pre-standardization study on pseudonym change management". Technical report, 2018c.

Amrita Ghosal and Mauro Conti. Security issues and challenges in v2x: A survey. 03 2019.

Stéphanie Lefèvre, Jonathan Petit, Ruzena Bajcsy, Christian Laugier, and Frank Kargl. Impact of V2X privacy strategies on Intersection Collision Avoidance systems. *IEEE Vehicular Networking Conference, VNC*, (May 2014):71–78, 2013. ISSN 21579857. doi: 10.1109/VNC.2013.6737592.

Pablo Alvarez Lopez, Michael Behrisch, Laura Bieker-Walz, Jakob Erdmann, Yun-Pang Flötteröd, Robert Hilbrich, Leonhard Lücken, Johannes Rummel, Peter Wagner, and Evamarie Wießner. Microscopic traffic simulation using sumo. In *The 21st IEEE International Conference on Intelligent Transportation Systems*. IEEE, 2018. URL https://elib.dlr.de/124092/.

Chad Morley. Seven Automotive Connectivity Trends Fueling the Future. https://www.jabil.com/blog/automotive-connectivity-trends-fueling-the-future.html. Accessed: 2020-11-30.

OMNET++. Inet manual. URL https://inet.omnetpp.org/docs/users-guide/ch-introduction.html. Accessed: 2022-02-01.

Dominic Paulraj, Nazeer Shaik, and André Weimerskirch. V2X Communication Security , Cyber Security , and Privacy.

Jonathan Petit, Florian Schaub, Michael Feiri, and Frank Kargl. Pseudonym Schemes in Vehicular Networks: A Survey. *IEEE Communications Surveys and Tutorials*, 17:228–255, 2015. ISSN 1553877X. doi: 10.1109/COMST.2014.2345420.

Ikjot Saini, Sherif Saad, and Jaekel Arunita. Evaluating the effectiveness of pseudonym changing strategies for location privacy in vehicular adhoc network. *Security and Privacy*, page e68, 05 2019. doi: 10.1002/spy2.68.

Florian Schaub, Zhendong Ma, and Frank Kargl. Privacy requirements in vehicular communication systems. *Proceedings - 12th IEEE International Conference on Computational Science and Engineering, CSE 2009*, 3 (March):139–145, 2009. doi: 10.1109/CSE.2009.135.

Christoph Sommer, Reinhard German, and Falko Dressler. URL https://veins.car2x.org/documentation/. Accessed: 2022-02-01.

Christoph Sommer, Reinhard German, and Falko Dressler. Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis. *IEEE Transactions on Mobile Computing (TMC)*, 10(1):3–15, January 2011. doi: 10.1109/TMC.2010.133.

International Telecommunication Union. Security guidelines for vehicle-to-everything (V2X) communication. Technical Report 2, International Telecommunication Union, 2020.

Andras Varga. Omnet++ manual. URL https://doc.omnetpp.org/omnetpp/manual/. Accessed: 2022-02-01.