Privacy Leakage Quantification in Vehicular Ad Hoc Networks

A thesis

submitted in partial fulfilment

of the requirements for the Degree

of

Doctor of Philosophy in Computer Science

at

The University of Waikato

by

Atthapan Daramas

THE UNIVERSITY OF
WAIKATO
Te Whare Wānanga o Waikato

2026

Abstract

Vehicular Ad hoc Networks (VANETs) offer the benefits of enhancing road safety and efficiency through the frequent information exchange among vehicles through continuous broadcasting of their locations and kinematics. While VANETs can be a potential solution to reduce road accidents, frequently announcing locations raises concerns about user privacy, as vehicles may become targets of trajectory tracking, resulting in the exposure of sensitive locations visited by vehicle users. Several works in the literature have tackled the problem of privacy leakage and privacy preservation through mechanisms that use periodically changing pseudo-identities—known as pseudonyms in VANETs. However, most of this work was done before standards bodies, such as ETSI and SAE, developed VANET communication standards that described how messages should be formed and broadcast, and when pseudonyms should be changed. The use of VANET communication standards makes attempts by the research community to preserve users' privacy outdated and leaves the quantification of privacy leakage in VANETs unaddressed.

By keeping communication standards as the context, this thesis aims to fill the gap in understanding potential privacy threats in VANETs and to quantify privacy leakage when these standards are enforced. We analyse potential VANET privacy threats using the LINDDUN framework—a privacy threat elicitation framework and uncover 13 privacy threats in VANET communication, ranging from unawareness of vehicle communication to privacy leakage through user reidentification when the full trajectory is reconstructed. The ability to reidentify a user's true identity from a reconstructed trajectory highlights two sides of the same coin in VANET privacy issues: privacy leakage through communication and the adversary's ability to reconstruct vehicle trajectories.

Given the scarcity of real-world datasets, research on VANET privacy quantification can be hindered from being conducted in accordance with communication standardisations. We analyse two real-world VANET communication datasets and identify three limitations in these datasets that render them inadequate for VANET privacy quantification research. Therefore, we develop two VANET communication datasets based on the VANET communication stan-

dards from two major standards bodies, ETSI and SAE, to serve as substitutes for inadequate real-world datasets.

Based on the findings from our privacy threat elicitation, three vehicle trajectory reconstruction models are developed to simulate privacy attacks against the standardised VANET communication. The three proposed trajectory reconstruction models include a path history-based approach, a Bayesian approach, and a Kalman filter approach. Our trajectory reconstruction models show that 65 to 98% of the trajectory can be fully reconstructed, where the amount of fully reconstructed trajectory depends on the tracking models and the communication standards. Our findings indicate that the VANET communication standards may not sufficiently protect users from privacy threats, as most of the trajectory can be reconstructed by the adversary, and highlight the need for enhanced privacy-preserving mechanisms that can be deployed alongside existing standards.

Acknowledgements

First of all, I would like to thank my supervisors, Dr Vimal Kumar and Dr Marinho Barcellos, for their guidance throughout my studies. Their continuing support and guidance on both academics and life are invaluable lessons more than I could have imagined I would receive from pursuing a PhD.

My years of study would not be memorable if I had not met these people in New Zealand. Thank you to the CROW members: Harpreet, Thye Way, Daniel, and Chris, for their encouragement to push through my study. Thanks to my best friend in New Zealand, Hayden L. Borrie, for the friendship we share, the weekday runs, and the weekend cafe-hopping. Thanks to my lovely landlords, Greg and Baiyohk Thompson, for their lovely house and edible garden, and for the countless interesting topics we talked about. Thanks to my energetic teammates and opponents at the Badminton club, especially Medihah, for constantly nudging me to join their games and be healthy.

My warmest gratitude goes to my supportive family, Pairoj, Supravee, and Yingpan Daramas, for their constant encouragement, emotional support, and understanding to let me pursue my dream. I would also like to extend my gratitude to my memorable host mother in Denmark, Agnes Klit Angemann, who encouraged me to try new things and be myself.

Lastly, I would like to thank the Faculty of Information and Communication Technology, Mahidol University, Thailand, for giving me the opportunity to take on this journey.

Contents

List of Figures

List of Tables

Acronyms

ARC-IT Architecture Reference for Cooperative and Intelligent Transportation.

BSM Basic Safety Message.

C2C-CC Car-to-Car Communication Consortium.

CAM Cooperative Awareness Message.

CDF Cumulative Density Function.

CSA Cloud Security Alliance.

DFD Data Flow Diagram.

ETSI European Telecommunications Standards Institute.

GMSF Generic Mobility Simulation Framework.

IoI items of interest.

ITS Intelligent Transportation System.

ITS JPO Intelligent Transportation Systems Joint Program Office.

LPPM Location Privacy-Preserving Mechanism.

NH Normalised Entropy.

PII Personally Identifiable Information.

PKI Public Key Infrastructure.

RSU Roadside Unit.

SAE Society of Automotive Engineers.

SCMS Security Credential Management System.

SPaT Signal Phasing and Timing.

SUMO Simulation of Urban Mobility.

TAPASCologne Traffic and Parking Simulation Cologne.

THEA Tampa Hillsborough Expressway Authority.

TIM Traveler Information Message.

USDOT The US Department of Transportation.

VANET Vehicular Ad hoc Network.

Chapter 1

Introduction

The concept of privacy means different things to different people, and the perception of retaining it differs from individual to individual. With different perceptions, multiple challenging privacy-related topics arise, including defining, accounting for, quantifying, and protecting privacy.

From the "right to be let alone" [1] to the "right to be forgotten" [2], the notion of privacy has become a more nuanced concept by adding the context to which privacy is attached to [3, 4]. Considering the right of a person and their personal space, Clarke defined and categorised privacy into four groups, including privacy of the person, personal behaviour, personal communications, and personal data [5]. The seminal work by Nissenbaum emphasised that privacy should be examined from both technological context and social norms surrounding privacy [3]. Nissenbaum described privacy violations as occurring when information is shared outside the context for which it was intended. Later on, Finn et al. [4] recognised the impact of evolving technologies and expanded upon Clarke's original classification of privacy [5], increasing the number of categories from four to seven. These types of privacy include privacy of the person, behaviour and action, communication, data and image, thoughts and feelings, location and space, and privacy of association.

Even with a more concrete definition, managing and maintaining privacy risks are still complicated. Since protecting privacy might require trading

off the system's security properties or utility, one might choose to protect the system's security or uphold the system's utilities, leaving privacy risks exposed. An example of trading off privacy for security is the threat analysis framework STRIDE [6], where 'R' represents *repudiation* threat. To preserve the 'non-repudiation' property, the system must be able to trace the user's actions so that the users cannot deny such actions they committed. In other words, the user's activities in the digital system are being traced to detect and prevent misbehaviour and preserve the system's non-repudiation.

Prior studies have also highlighted that balancing privacy and system utilities makes it difficult to achieve a fully privacy-preserving system [7,8]. Buchholz et al. provided a study on using a deep-learning approach for generating synthetic datasets while trying to achieve both privacy guarantees and the dataset's utility [7]. The work highlighted shortcomings, even with deep-learning techniques, in maintaining the dataset's utility and providing a strong privacy guarantee. In another work [8], a safety application in vehicular ad hoc networks (VANETs) was used as an example of balancing between privacy and application utility. Emara et al. proposed a methodology to evaluate the system's utility and privacy protection provided in VANETs. Their work consisted of a vehicle trajectory reconstruction model as a privacy quantification approach, along with metrics for evaluating the user's privacy and the system's utility. The authors highlighted the tradeoff between the user's privacy and the system's utility and provided a countermeasure to secure the users' privacy. The study from Emara et al. highlighted the conflicts between the user's privacy and the system's utility and inspired us to investigate the system further.

Multiple surveys have been conducted in recent years on VANET systems to understand users' perceptions of privacy in vehicle-related systems [9–11]. A 2021 survey by Cichy et al. [9] highlighted that, due to the perception of data ownership and a lack of incentive, users are not willing to share their data (braking, acceleration, and speed), even if it does not affect their pri-

vacy. Another survey was published by Bella et al. [10] on the readability of car manufacturers' privacy policies and users' perceptions of these policies. Their survey found that the distribution of opinions on the importance of personal data collection for the car's functionality is evenly split between those who agree, disagree, and are undecided. On the other hand, a more recent study [11] showed that around 55 percent of survey participants are willing to trade off their privacy to acquire the benefits of VANETs, which highlights a contradiction to the survey by Cichy et al. [9]. It is important to note that the participants in [10] and [11] believed that the data was processed and transmitted securely, However, only in [11], where an implication of privacy leakage was presented to the participants, signified that privacy concerns might have little effect on the users.

We could infer that the changes in perception and norms of VANET users regarding data sharing are reflected in the users' perceptions in these surveys [9–11], where the benefit of sharing personal data outweighs the privacy risk. The survey results in [11] also illustrated that a short description of privacy implications could raise awareness of privacy risks among the participants. However, the privacy implications of location sharing could be worse than just information disclosure, as highlighted in the work of [8], which reveals potential privacy leakage through trajectory reconstruction and the identification of users' homes and workplaces. The question of "how to quantify the adversary's capability in the vehicle trajectory reconstruction" sets the main objective of this thesis, which aims to evaluate the potential for privacy leakage in VANET communication.

1.1 Motivation: Privacy in VANETs

While there are pilot projects on VANETs, their full-scale adoption remains limited. Consequently, many privacy risks in VANETs have been primarily studied through simulation and analytical models rather than in large-scale

real-world deployments. Nevertheless, incidents related to car data breaches, which are closely related to VANETs, have been reported in the past few years [12–16].

According to the Cloud Security Alliance's (CSA) report [17], the Toyota data breach [12, 13] exposed vehicle location information for approximately 2.15 million users in Japan and lasted a decade. The cause of the Toyota data breach, however, is a cloud misconfiguration used to collect vehicle locations. Similar breaches have also occurred at other automotive brands, such as Volkswagen [14], where vehicle location was part of the data breach, and Subaru, where a hacker could uncover vehicles' historical locations [15].

Another incident that highlighted a different privacy threat model regarding privacy and institutional access to vehicle data is Tesla handing over Cybertruck vehicle telemetry to law enforcement [16]. While the intention to cooperate with the investigation is genuine, it raises questions about what the authority could do with the data. Connecting this incident to the concept of VANETs, in which vehicle location is collected by the government [18, 19], such architectures may reduce reliance on automakers as intermediaries and could increase the availability of vehicle location information to government agencies, depending on the deployment model and regulatory framework.

Collectively, although these incidents did not arise from VANET communication, they demonstrate that adversaries have demonstrated a clear interest in vehicle location data, and that large-scale collection of mobility information can create significant privacy risks when improperly protected. With VANETs being introduced, such information may become openly accessible through wireless broadcasts exchanged between vehicles and infrastructure, rather than communication between the vehicle and the automaker. Understanding the privacy risks associated with such communication remains an important research topic.

1.1.1 Vehicular Ad hoc Networks

To define and quantify the level of privacy leakage, it is important to thoroughly understand the system's functionality and the regulations that govern it. A Vehicular Ad hoc NETwork (VANET) is a concept of a network in which vehicles can wirelessly communicate with each other in an ad hoc manner. The goal of VANETs is to increase safety and efficiency for road users. The vehicles periodically broadcast their location and kinematics to all entities in their vicinity, announcing their presence in the system. These entities can be other vehicles, roadside units, or road users, such as pedestrians. Multiple types of applications can be implemented on top of the VANETs communication to enhance safety in road usage; for instance, the message receivers can avoid a potential crash without seeing the message sender. Additionally, with a city-wide deployment of roadside units, traffic analysis and road usage planning are feasible through forwarding information to a traffic control centre [20].

Two standard organisations that govern how the vehicles should communicate in VANET are SAE International and the European Telecommunications Standards Institute (ETSI). These organisations standardise VANET communication to ensure harmonised communication between different vehicle manufacturers. The standards from these organisations also serve to ensure trust between entities while preserving users' privacy. Trust between VANET entities can be established through message signing and the use of digital certificates. This certificate is specifically called '*pseudonym*' in the context of VANETs and is used for signing every message that is sent by the entity [21, 22]. Message signing allows entities in VANETs to verify the integrity of the message, while a pseudonym ensures the sender's authenticity.

While trust among the vehicles can be established through message signing, balancing between the system's utilities and users' privacy requires an additional step in the communication to prevent privacy leakage. Since the main objective of vehicle-to-vehicle communication in VANETs is to be used in life safety applications, an instant crash avoidance reaction could be con-

sidered as one of the system requirements. This leads to the design in which messages used for communication in VANETs, which relate to safety-critical applications, must not be encrypted to reduce overhead on both sender and receivers [21, 22]. Using unencrypted messages may lead to the abusive use of the user's identity and location privacy through vehicle tracking. VANETs achieve their first layer of protection to safeguard the user's privacy by using pseudonym certificates instead of the user's real identity. Pseudonyms enable communication in VANETs to be anonymised and prevent adversaries from directly associating information with the user's real identity. Additionally, the pseudonym certificate generation processes are designed to prevent privacy leakage from insider attacks through multi-tier certificate authorities [23, 24]. Lastly, the pseudonym will be periodically changed to prevent the vehicle from being tracked through repetitive use of the certificate [21, 25]. Frequently changed pseudonyms not only protect single-trip tracking but can also prevent association between the trip information and the user's real identity [26].

Despite attempts to safeguard user privacy, multiple studies have shown that privacy threats still exist in VANETs [27, 28]. Dok et al. identified two subjects where a privacy-related issue may occur [27]. These include vehicle privacy, which concerns information about the vehicle, and personal privacy, which concerns information about the driver. While Dok et al. classified privacy threats into two types, Hanh et al. focused only on the person's privacy [28]. They classified the privacy threats to a person into three groups: identity, location, and behaviour. The research has established concerns about privacy issues in VANETs, especially the threat to users' privacy when they can be tracked, profiled, and impersonated.

To address the tracking and profiling issues, various techniques were proposed from the perspective of VANET and location-based services in the last decade [29–35]. Because these techniques attempt to perturb the information, they usually do not align with VANET's communication standards and may also decrease the application's utility. Multiple works have also at-

tempted to propose pseudonym-changing mechanisms in the past decade [36–46]. Wahid et al. [46] categorised these pseudonym-changing mechanisms into four approaches, including group formation [36, 38], fixed-place [43], vehicle-centric [37, 40], and cooperative schemes [37, 45]. However, most of these techniques did not account for the existence of VANET standards; either the work was published before the standards were established, or the objective of the work was to propose an alternative approach to the standards themselves.

Most research on privacy-preserving communication in vehicular ad hoc networks (VANETs) has identified a connection between threats to location and identity privacy [27, 28]. Vehicle tracking can expose sensitive locations, such as a user's home, workplace, frequently visited shopping areas, and other places where the user wishes to remain anonymous. The disclosure of home and office locations can lead to deanonymisation of an individual [26], which may result in long-term tracking and disclosure of other locations visited by the individual. As the purpose of VANETs is to improve road users' safety and efficiency, the capability to track, profile, and deanonymise individuals creates a tension between the system's utility and the users' privacy. Various privacy-preserving techniques proposed so far were either prior to the standardisation of the VANETs or diverged from their standards [36,37,40]. To our knowledge, few prior studies attempted to align their privacy-preserving mechanisms and privacy quantification with the VANET's standards [8]. Although it is clear that VANETs experience privacy leakage issues, it is essential to systematically identify and quantify the privacy leakage through the lens of VANET standards, and address the question: *"To what extent can the seriousness of privacy leakage in VANETs be?"*.

1.1.2 Privacy Quantification

To quantify the severity of privacy leakage or compare results from different adversary models, a privacy metric can be used as a tool for quantitative analysis. Privacy metrics are tools for measuring the level of privacy leakage or the

effectiveness of a privacy-preserving technique [8,47,48]. With a large number of metrics available for use in privacy quantification, choosing a suitable privacy metric can be overwhelming. Wagner evaluated the strength of privacy metrics for VANET's privacy quantification in [49]. The strength of a privacy metric is measured by its monotonicity, which indicates whether the privacy metric's outcome changes in the expected direction as the adversary's strength increases or decreases. Wagner emphasised the importance of utilising a combination of privacy metrics to evaluate privacy leakage issues from multiple perspectives. Later in 2019, a survey by Wagner and Eckhoff [48] classified privacy metrics based on the output into eight groups, including uncertainty, information gain or loss, data similarity, indistinguishability, adversary's success probability, error, time, and accuracy or precision, where privacy metrics in data similarity groups and accuracy or precision are widely used in VANET and location-based research [35, 50–54].

A privacy metric can also expand upon the classification by Wagner and Eckhoff by incorporating context-specific information [8, 50, 55]. To evaluate the effectiveness of their vehicle tracking model against the pseudonym changing mechanism, Emara et al. [8] used *"tracking percentage"* and *"average number of confusions"* as their privacy metrics. Without considering a vehicle tracking model, Corser et al.'s model evaluated continuous anonymity when vehicles communicate as a group [50]. In their work, three aspects of privacy were evaluated: the anonymity set size, the distance between vehicles within the same anonymity set size, and the duration during which a vehicle remains anonymised. Another notable work accounted for the relationship between the user's identity and the visited locations [55]. Unlike the work in [8], Ma et al. assumed that the probability between the user's identity and the locations associated with the identity already existed [55]. These existing works highlighted that the VANET's context can shape the definition of privacy and the method of evaluating privacy loss. However, these works also contain diversions from the actual VANET standards, with Emara et al. [8] being the

closest work to the VANET standards.

It is also worth mentioning a seminal framework by Shokri et al. for location privacy quantification [47], although the framework was not specifically designed for VANETs. The authors break down a privacy preservation system into six parts: user, user's actual trace, location privacy-preserving mechanism (LPPM), observable trace, adversary, and privacy metrics. The framework describes a user who publishes an observable trace by applying LPPMs to the actual trace. The adversary's goal is to reconstruct an actual trace from the observable trace, and privacy metrics measure their success rate in achieving this goal. The authors also assumed that the adversary knows the details of location privacy-preserving mechanisms. While the work shares common perspectives with VANETs, e.g., the pseudonym-changing mechanism is known to the adversary, adaptations to the framework are required to fit the VANET concept. For instance, the pseudonym-changing mechanism in VANETs is nearly deterministic, unlike the LPPMs in the framework, which are probabilistic.

It is also important to highlight a shared approach in the literature, where simulated datasets have been used in the evaluation process. While the US Department of Transportation has published real-world VANET datasets, the divergence from the VANET standards in the literature makes the dataset incompatible with those works. Another aspect that renders the datasets unusable is the pre-processing before data publication to preserve users' privacy. While the adversary models in [8, 47] have already performed well on a small-scale dataset, the impact of dataset size on privacy leakage is not yet well understood.

1.2 Contributions and Thesis Structure

In this thesis, we systematically quantify the privacy leakage in VANET communication following the standards published by the SAE and ETSI. While

multiple applications can be used in VANETs, this thesis focuses on VANET's *basic safety message*, a foundation for safety applications. In such a setting, road users participate in VANETs to use the basic application to reduce the potential for accidents and increase road safety. We assume that all users are legitimate and follow the VANET's protocol and communication standards, where messages are signed and exchanged according to the standards. Due to the openness of their communication, users are concerned about whether the privacy-preserving mechanisms, namely the pseudonym-changing mechanism, can sufficiently protect their privacy. The contributions of this thesis are as follows:

- A systematic VANET privacy threats identification based on SAE and ETSI's standards with LINDDUN privacy threats elicitation framework.
- Datasets of VANET communication based on SAE and ETSI's standards, including a dataset of Basic Safety Messages and Cooperative Awareness Messages.
- Three vehicle trajectory reconstruction models based on BSM and CAM's path history information, Bayesian inference of the vehicle's future location and kinematics, and a Kalman filter approach for estimating the vehicle's future location and kinematics.
- Privacy leakage quantitative comparisons between pseudonym changing mechanisms based on SAE and ETSI's standards, and the effectiveness of trajectory reconstruction models.

These contributions are organised and presented in the following chapters of the thesis. The summary of each chapter, its relation to our contributions, and its relation to other chapters are as follows:

Chapter 2 We present the background and related work that connect to privacy quantification in VANETs through a trajectory reconstruction approach. The chapter begins by presenting sufficient information to understand

the underlying concept of VANET communication and its architecture, as well as the privacy-preserving approaches presented in the standards. To help understand the privacy threat elicitation framework, we explain the privacy threats within the LINDDUN framework, the methodology for applying the framework, and the related work that employed LINDDUN in VANETs. We then explore the privacy quantification methodology from a seminal location privacy framework, related work on privacy quantification in VANETs, the privacy metrics used in this work, and the datasets used during their evaluation. The chapter ends with the concept of the Kalman filter, along with an example to help understand its application in trajectory reconstruction.

Chapter 3 We begin our analysis of privacy threats in VANET communication by eliciting privacy threats using the LINDDUN framework. In applying threat elicitation, we analyse and utilise the Architecture Reference for Cooperative and Intelligent Transportation (ARC-IT), an architecture supported by the US Intelligent Transportation Systems Joint Program Office, to build an understanding of VANET communication and identify privacy threats based on real-world communication architecture. With an understanding of VANET communication and the standards that govern it, we formalise the communication process into mathematical formulas, preparing to propose trajectory reconstruction approaches and systematically quantify the privacy loss in VANETs. In the process of formalising VANET communications, we also indicate its connections with the stage at which privacy threats arise. The chapter ends by comparing our formalised VANET communication stages with the privacy quantification frameworks in the literature, and highlighting the gaps in VANET privacy quantification through trajectory reconstruction research.

Chapter 4 We continue on the path of quantifying privacy by preparing a dataset for use in trajectory reconstruction models. This chapter of the thesis begins by presenting an analysis of public datasets from pilot projects conducted by the US Department of Transportation. We highlight three limitations of the public datasets, which render them insufficient in VANET pri-

vacy research with the trajectory reconstruction approach. We then present our methodology for creating our synthetic dataset, which holds three principles for privacy research using a trajectory reconstruction approach. These principles are standard compliance, appropriate vehicle density, and adequate spatial coverage. Despite the inadequacies in the public datasets, we utilise their characteristics for comparison with our synthetic dataset.

Chapter 5 We propose three trajectory reconstruction models that are based on our understanding and analysis preceding this chapter. The three trajectory reconstruction models are path history-based, Bayesian-based, and Kalman Filter-based trajectory reconstructions. These models demonstrate the implementation of privacy threats presented in Chapter 3. The path history-based trajectory reconstruction is a unique threat in VANET, to our knowledge, that has not been explored before. On the other hand, while the other two approaches have been used in the literature, we emphasised in Chapter 3 the gaps presented in the literature, which required further analysis presented in this thesis.

Chapter 6 We quantify the privacy leakage in VANET communication through the proposed trajectory reconstruction approaches. Since the user's privacy is interconnected with the adversary's success rate in the trajectory reconstruction, we quantify the privacy leakage through the evaluation of the adversary's success and failure. The evaluations are divided into three main tranches: ambiguous trajectory evaluation, pseudonym linkage evaluation, and trajectory reconstruction model evaluation. The first evaluation focuses on the problem that the adversary may face during the trajectory reconstruction process. On the other hand, the second and third evaluations focus on the success rate of the adversary, i.e., the severity of the privacy leakage in VANET communication. The evaluations in this chapter highlight the possibility of privacy threats introduced in Chapter 3 when the proposed trajectory reconstruction models in Chapter 5 are implemented.

Chapter 7 We conclude the thesis with a summary of our analysis of

privacy threats in VANET communication through trajectory reconstruction. We also discuss the implications of our findings and future research directions for privacy-preserving communication in VANETs.

Chapter 2

Background And Related Work

In this chapter, we provide background knowledge and related work to help readers understand the concepts and grasp the material presented in the following chapters of the thesis. This chapter begins with the core concept of this work, the VANET standards, by providing an understanding of the VANET standards that are central to the work in this thesis. In Section 2.2, a privacy assessment framework, LINDDUN, is presented. Then, related work on privacy quantification is presented in Section 2.3. Lastly, the chapter concludes with an introduction to the Kalman filter, which is utilised in our vehicle tracking model.

2.1 VANETs Standards

This section examines how entities in VANETs communicate by reviewing the standards that govern VANET communication. Additionally, we also review documents from the US Department of Transportation pilot projects. The two primary standards organisations involved in VANET communication are SAE International and the European Telecommunications Standards Institute (ETSI). SAE standards are primarily adopted in the United States and Canada, while ETSI standards are widely used across European nations. Despite being governed by two distinct organisations, these standards share fundamental similarities in their overall approach.

2.1.1 VANETs Communication Architecture

We will first review two VANET communication architectures, namely the Security Credential Management System and the Architecture Reference for Cooperative and Intelligent Transportation, to provide an overview of VANET communication. The Security Credential Management System is an architecture referenced in VANET literature [56–59] that describes the functionalities of VANET authority entities, their relationships with others, and the processes of user enrollment, participation, and revocation. On the other hand, the Architecture Reference for Cooperative and Intelligent Transportation [60, 61] outlines the planning, design, and implementation processes of connected vehicle systems, considering the viewpoints from multiple stakeholders, and is referenced by the US Intelligent Transportation Systems Joint Program Office.

The VANET's **Security Credential Management System (SCMS)** is an architecture that comprises a collection of authorities whose goal is establishing trust between VANET entities [56–59]. While the majority of VANET entities will be users, including public and private vehicles, it also encompasses the authority entities of VANET. To establish trust, the SCMS will be able to bootstrap VANET entities, provide certificates, manage misbehaviour reports, and revoke misbehaving entities. With the ability to provide certificates, the SCMS is considered to partially act as a Public Key Infrastructure (PKI). The key differences between SCMS's PKI functionality and the traditional PKI are the number of certificates that need to be handled and the balance between security, privacy, and efficiency [58]. Figure 2.1, taken from [59], illustrates the structure of the Security Credential Management System.

From Figure 2.1, the notable components which differentiate VANET's SCMS from traditional PKI include location obsecurer proxy, pseudonym authority, linkage authorities, and misbehaviour authority. **Location obscurer proxy** hides a device's location by altering source addresses and shuffling misbehaviour reports and certificate requests before forwarding them to other components, thereby preventing the opportunity to link network addresses to

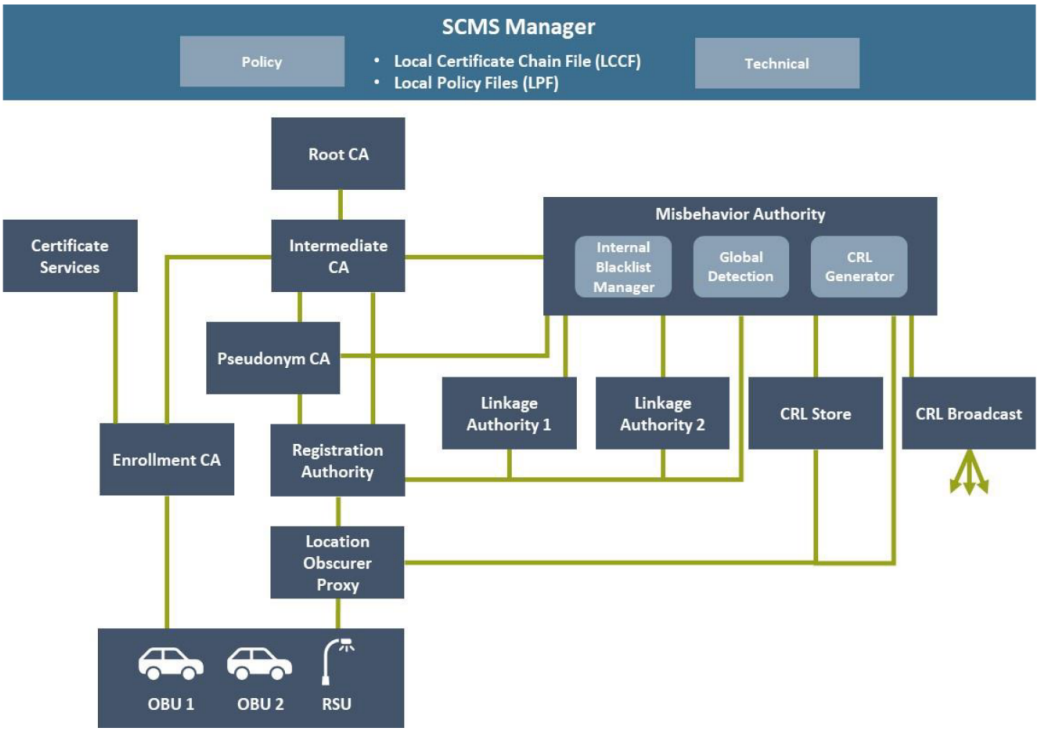


Figure 2.1: VANET SCMS structure.

device locations and preserving user privacy. **Pseudonym authority** generates pseudonym certificates for a device so that the device can use certificates to sign every message it sends. **Linkage authorities** generate linkage values between pseudonym certificates, which can be used for certificate revocation. **Misbehaviour authority** identifies misbehaviour vehicles based on misbehaviour reports. If misbehaviour is detected, the device and its certificates will be added to the certificate revocation list.

According to [58], the SCMS components work together while maintaining their separation to balance VANET's security, privacy, and efficiency. The security goal is to provide trust among VANET's entities through messages that are signed with certificates. Next, the user's privacy is achieved by separating SCMS components. The SCMS components and their functionality are designed so that the linkage between a user's identity and certificates can be revealed only if the SCMS components work together. Therefore, the SCMS components must be physically separated and owned by different organisations. As a result, a specific authority can only obtain partial information

about the user. A further consideration for a privacy-preserving mechanism is to allow VANET entities to change their signing certificate periodically, hence shortening the authority's observation period. The mechanism for pseudonym certificate changing will be discussed in 2.1.3. However, periodically changing the signing certificate means a higher demand for certificates from VANET entities and may reduce the SCMS's efficiency. The SCMS efficiency can be increased through a technique called butterfly key expansion and a consideration of the number of certificates required by VANET entities.

Another architecture in VANETs is the **Architecture Reference for Cooperative and Intelligent Transportation (ARC-IT)**, which is referenced by the US Intelligent Transportation Systems Joint Program Office (ITS JPO) [60, 61]. The architecture serves as a common language for different stakeholders to plan, design, and implement an intelligent transportation system (ITS). The ARC-IT describes the intelligent transportation system through four viewpoints, each of which can conceptually be connected to the others. The four viewpoints of ITS by ARC-IT are the enterprise, functional, physical, and communications views, each serving a distinct perspective on ITS planning, design, and implementation.

At the ARC-IT core, a *service package* is a group of services related to the ITS, categorised into multiple focused areas, such as public safety, public transportation, vehicle safety, etc. The service can be used by a physical object that belongs to the *physical* viewpoint. Additionally, the physical viewpoint also describes the physical object's associated functionalities, as well as the information flows between these physical objects, which are depicted through triples: source, information flow, and destination. Each service package, once associated with a physical object, will also be assigned a security class from 1 to 5, where security class 1 is the lowest security risk. Then, the *communication* viewpoint identifies the actual communication protocol for the communication between physical objects. The information flow in the physical viewpoint can also be described in detail by a data flow and processes in the *functional*

viewpoint. Lastly, the *enterprise* viewpoint describes the entities in the ITS, including organisations or users, their roles, and the relationship between these entities.

Let us provide an example of how the service package and the four viewpoints are connected. For instance, a 'vehicle' is a physical object from the physical viewpoint [62]. The vehicle should have a 'vehicle maintainer', which is an object from the enterprise viewpoint, whose role is to maintain the vehicle. One of the triples in which the vehicle is considered as a source is 'vehicle location and motion' [63], i.e., the vehicle sends its location and motion to other entities. From the communication viewpoint, this information flow can be achieved by following the standards described in [64]. The vehicle location and motion triple can be partially described by the 'tov-vehicle_location' dataflow from the functional viewpoints [65]. One of the services that the vehicle can use is 'Vehicle basic safety communication' [66], which belongs to the 'V2V basic safety' and other service packages. Meanwhile, the 'V2V basic safety' service package is classified as a security class 3 since it has low confidentiality, high integrity, and moderate availability [67]. The lists of device security classes, vehicle functionalities, and triples are listed in Appendix B

2.1.2 Communication in VANETs

The VANET communication is a building block for its applications, which include, but are not limited to, forward crash warning, blind spot warning, intersection movement assist, emergency vehicle warning, etc. [68]. These safety-related applications leverage incoming messages and the vehicle's current spatio-temporal information to assist the driver. For example, in a forward crash warning, the vehicle can alert the driver of a potential/imminent crash after sensing an approaching vehicle via message reception [68].

Vehicles exchange real-time spatio-temporal information with others through messages via wireless communication. A type of message that could be considered as the heartbeat message for VANET applications is *Basic Safety Mes-*

sage (BSM) and *Cooperative Awareness Message* (CAM). These two types of messages are equivalent to one another, but are backed by different standard organisations. The BSM follows the SAE standard [69], used in the United States, Canada, Australia, etc. On the other hand, the CAM follows the ETSI standard [70], used in the European Union, United Kingdom, India, etc. These messages[1] enable receivers to determine the sender's location and kinematics, enhancing situational awareness. BSMs and CAMs contain not only real-time spatio-temporal data but also historical and predicted information [69, 70]. Vehicles broadcast these messages at a frequency of 1–10 Hz to all nearby entities [21, 71]. While entities are mainly vehicles, they could also include other road users, such as cyclists, pedestrians, and roadside Units (RSUs). To minimise communication overhead, these messages are not encrypted [21]. A detailed list of fields in the message is provided in Table A.1.

Although the Basic Safety Message and Cooperative Awareness Message have different ways to represent the data fields, similar information can be observed in both types of messages. In the Basic Safety Message (BSM), information is grouped into two main dataframes: core and optional dataframes [69]. As stated in SAE J2735 (2020), the core dataframe will be presented in every BSM, while an optional dataframe is not a requirement for a valid BSM [69]. On the other hand, the Cooperative Awareness Message (CAM) does not allow 'optional', but the information may be presented as 'unavailable' [70]. Even though the field name and granularity vary between BSM and CAM, most of the information is presented in both types of messages. For example, the speed unit in BSM is 0.02 m/s, while the speed unit in CAM is 0.01 m/s. Nevertheless, both standards can represent speeds up to 163.8 m/s. There is also a conflict in BSM's compatibility between two standards from the same organisation, SAE J2735 (2020) [69] and SAE J2945/1 [21], where a 'path history' is required for disseminating the BSM in the latter, but is an optional

[1]The word *message* will be used to refer to both Basic Safety Message (BSM) from SAE and Cooperative Awareness Message (CAM) from ETSI unless stated otherwise.

data frame in the former standard.

A path history consists of a list of entries that comprise the time and location where the vehicle was in the past, and each entry in the path history is called '*path history point*'. These locations are presented as an offset of latitude, longitude, and altitude from the current location [69, 70]. The objective of including the path history in the message is to enhance the message receivers' situation awareness and collision detection [69, 71]. The receivers of the message with the path history can observe the sender's behaviour based on their past locations in the path history and their current location. In ETSI TS 102-894-2 (2018) [70], 1 to 7 path history points are presented in the message. On the other hand, SAE J2735 (2020) stated that the message will contain 1 to 14 path history points. Note that in BSM, while the location offset and the vehicle's current location are presented with the same unit granularity, the unit of time offset in path history and the unit of time in the message are different. The time unit in the core dataframe is one millisecond, and the unit of time offset is 10 milliseconds. According to the standard SAE J2945/1, the path history points are selected from the locations in previous CAMs to achieve at least the 200 metres total distance, with a minimum number of path history points, while maintaining a minimum distance error. Experiments conducted in SAE J2945/1 showed that the radii of road curvature affect the total number of path history points [21]. For example, eight path history points are required to represent a curved road, while only two are required for a straight road. Nevertheless, an experiment also showed that, with 91.3% of the time, five or fewer path history points are needed to represent 200 metres.

2.1.3 Pseudonym Changing Mechanisms

To create trust between entities in VANETs, each communicated message is signed with a certificate. Specifically to BSM and CAM, the certificate for

signing these messages is called *pseudonym certificate*[2] , which is generated by a pseudonym authority. However, using the same pseudonym certificate to sign every message will allow the adversary to group messages according to the certificate and track the users based on the locations in the grouped messages.

To preserve user privacy, both standard organisations suggest that the pseudonym certificate must be changed periodically [21, 25]. The ETSI suggested multiple certificate-changing mechanisms in the pre-standard document in April 2018 [25]. These mechanisms include constant triggering parameters, pure randomness triggering parameters, and a mix-zone based on aids from roadside units. The document also highlighted an approach from the Car-to-Car Communication Consortium (C2C-CC), in which distance and time are used as triggering conditions. This approach aims to divide the vehicle trajectory into three parts: the trip's beginning, middle, and end. Based on C2C-CC, the first certificate change trigger is after 800–1500 metres of travelling from the trip's start. The subsequent triggers are based on 800 metres of travelling and within 2–6 minutes since the last certificate changed. Similarly, SAE defined the certificate-changing triggering conditions based on distance and time [21]. In SAE, the duration of pseudonym usage must not exceed 5 minutes unless the *absolute* distance from the last certificate change is less than 2.1 kilometres.

With periodically changing certificates, many certificates must be available to vehicles to support seamless communication. The number of certificates, however, depends on the Security Credential Management System's ability to generate valid certificates and the ability of vehicles to store valid certificates. The US Department of Transportation (USDOT), which follows the guidelines from SAE standards, has set the number of certificates to be 20 per week [24]. However, the number of certificates per week could be negotiated, as seen in the New York City pilot project, where 60 certificates are available at a

[2]The word *pseudonym certificate* and *pseudonym* are used interchangeably in VANETs privacy research.

time [59]. The New York City pilot project also proposed that the pseudonym should be changed every 2 *travel kilometres* or 5 minutes, *whichever comes first*. These negotiations were proposed to protect the privacy of long-distance road users, such as taxi drivers.

The summary of standards that are referenced in this thesis is listed in Table 2.1. Note that the standards are sorted by their standard numbers.

Table 2.1: A summary of VANET standards accounted for in this thesis.

Standard Number	Synopsis
ETSI EN 302 637-2 [71]	Provide specification for Cooperative Awareness service, syntax and semantics of CAM, and how CAM should be handled.
ETSI TR 102 638 [68]	Define basic set of applications.
ETSI TR 103 415 [25]	Analysis of pre-standard on pseudonym changing.
ETSI TS 102 894-2 [70]	Provide data dictionary for VANET communication.
ETSI TS 102 940 [72]	Defines a security architecture and outlines the required functional entities.
ETSI TS 102 941 [22]	Identifies the trust and privacy management in VANET communication.
ETSI TS 103 097 [73]	Defines messages' security header, and certificate formats.
FHWA-JPO-17-461 [74]	Explains the data privacy plan adopted by Tampa pilot projects.
FHWA-JPO-19-775 [59]	Provides description of SCMS adopted by USDOT in their pilot projects.
SAE J2735 [69]	Provide data dictionary for VANET communication.
SAE J2945/1 [21]	Specifies the system requirements for VANET communication, which includes the pseudonym changing algorithm.
SAE J2945/2 [75]	Extends SAE J2945/1 standard to cover more VANET applications.

2.2 LINDDUN Framework

The framework is inspired by the STRIDE threat model [6] but focuses on privacy threats. The privacy threats in LINDDUN include **L**inking, **I**dentifying, **N**on-repudiation, **D**etecting, **D**ata disclosure, **U**nawareness, and **N**on-compliance. The privacy threats include **L**inkability, **I**dentifiability, **N**on-repudiation, **D**etectability, information **D**isclosure, content **U**nawareness, and policy and consent **N**on-compliance. Similar to STRIDE, the LINDDUN framework leverages a data flow diagram to analyse and elicit system privacy threats.

2.2.1 Privacy Threats

The privacy threats in the LINDDUN framework are derived from the privacy properties that a system should have. The privacy threats are listed as follows:

Linking is a threat where the adversary is able to identify the relation between two items of interest (IoIs), which does not need to contain the user's identity. The IoI can be pseudo-identities, contextual and non-contextual information in the message, records in a data store, and the user's actions. Linking threats may arise from revealing too much information in the communication or storing too much information in the database. Identifying whether a relation between IoIs exists will suppress the *unlinkability* privacy property. Being able to link between IoIs allows the adversary to observe, learn, and infer the user's behaviours without knowing the user's identity.

Identifying is a threat where the adversary is able to identify the user's identity. This threat may arise from unintentionally revealing the user's identity, such as through a data breach, or from implications based on information learned from other threats. Due to a known identity, this privacy threat is related to the *anonymity* and *pseudonymity* privacy properties. Identifying poses a significant threat, as it reveals the user's true identity.

Non-repudiation refers to an undeniable claim of performing actions or obtaining properties. This privacy threat arises from an insecure or unen-

crypted logging system or insufficient protection of the user's actions and identity. With a non-repudiation threat, the adversary can use the claim or the ownership against the user, such as a whistleblower. *Plausible deniability* is the privacy property related to this threat in which the user can deny performing such actions. The authors in [76] acknowledged that non-repudiation can also be considered a system requirement where actions performed by a user must be identifiable.

Detecting threat is when the existence of the IoI is revealed to the adversary. The IoI in detecting a threat can simply be a message on a wireless communication channel or the presence of a user at a specific location. Detecting may arise from the adversary's capability to eavesdrop on the communication. Since the presence of an IoI is revealed to the adversary, the 'detecting' threatens the *undetectability* privacy property.

Data disclosure occurs when information is exposed to an unauthorised entity, or unnecessary information is collected by a system. This threat also appears in the STRIDE model [6], highlighting the importance of the privacy property associated with the threat. By exposing information to an unauthorised entity or collecting unnecessary information from the user, excessive information about the user is revealed to another party. With the information being disclosed to an unauthorised party, the data disclosure suppresses the *confidentiality* privacy property.

Unawareness arises when insufficient information on how the user's data is being collected or processed is shared with the user. Consequently, users may share too much information with the system, which may lead to other privacy threats, such as linking and identifying. The counterpart of this privacy threat is *content awareness*.

Non-compliance is a privacy threat where the company does not follow its privacy policies. This privacy threat is also closely related to regulations and legislation, where the company must adhere to specific policies. By not following the regulations or their policies, the company risks being unlawful

and increases the risk of other privacy threats. Therefore, the presence of this threat signifies the violation of *policy and consent compliance* privacy property.

2.2.2 Applying LINDDUN

The process of privacy threats elicitation with LINDDUN comprises three steps [77]. It begins with a crucial step, **(1)** creating a Data Flow Diagram (DFD). This step is important since an incorrect DFD can lead to an incorrect interpretation of privacy threats that may arise. Then, **(2)** the DFD elements are mapped with LINDDUN threats to identify the susceptible element to a privacy threat. Table 2.2, taken from [77], shows the relation between the privacy threats and the DFD elements. The *X* mark in the table indicates the possible susceptibility to the threat.

Table 2.2: LINDDUN threats mapping with DFD elements.

Threat categories	Entity	Data Flow	Data Store	Process
Linking	X	X	X	X
Identifying	X	X	X	X
Non-repudiation		X	X	X
Detecting		X	X	X
Data Disclosure		X	X	X
Unawareness	X			
Non-compliance		X	X	X

Lastly, **(3)** the privacy threats are identified through three sub-activities: identify threats through threat tree patterns, document assumptions, and document the found threats.

For each privacy threat, a threat tree pattern helps identify the cause of the privacy risk by asking the threat elicitor to review the system based on the path from the root node to a leaf node in the tree. Figure 2.2 shows an example of a threat tree for the linking privacy threat. All seven threat tree

patterns, taken from [78], are listed in Appendix C.

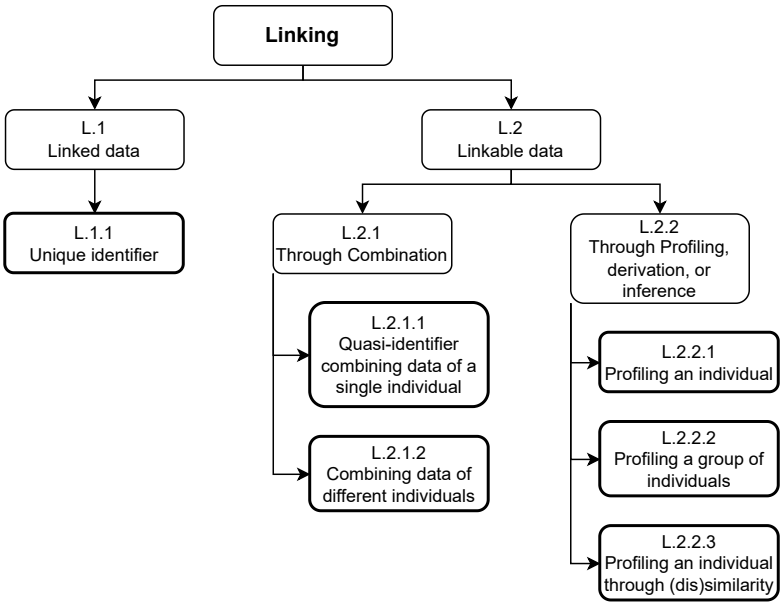


Figure 2.2: Threat tree - Linking.

Taking an example of linking privacy threat to the data flow, the threat elicitation will start from the root node *Linking* and continue the path to the leaf node. Then, for each node, the threat elicitor will ask a question: "Does the property highlighted on the node exist in the system?". For instance, at the *L.1* node, the question will be "does the data flow between an entity and a process contain linked data?". If the system possesses such a property, the threat elicitation process continues to the next level of the tree until reaching a leaf node; otherwise, go to the other child of the current parent node. Note that, on each node, any assumptions made to identify the privacy threat should be documented. Finally, once the leaf node is reached, the threat must be documented. The tutorial [77] also suggests three additional steps to address the identified threats.

These addtional steps include, **(4)** prioritise threats, **(5)** elicit mitigation strategies, and **(6)** select corresponding privacy enhancing technology.

An example of LINDDUN usage in VANETs was presented by Chah et al. in [79]. The authors used a single high-level data flow diagram to elicit the privacy threats. They identified 15 privacy threats, 3 for each of the

following: linking, identifying, non-repudiation, detecting, and data disclosure. Additionally, the works by Raciti and Bella [80, 81] mentioned LINDDUN in their work, but do not follow the threat elicitation processes by the framework. Their divergence from the LINDDUN framework is that their threat elicitation did not start with a data flow diagram. The authors identified potential threats from four sources and mapped them with LINDDUN threat tree patterns. The authors identified 56 privacy threats in [80] and 17 privacy threats in [81].

2.3 Privacy Quantification

2.3.1 Location Privacy Framework

Seminal location privacy quantification frameworks were proposed in [82, 83]. Although these frameworks are not solely specific to VANETs, they break down the concepts of location privacy and help systematically analyse related privacy issues. The analysis of location privacy can be divided into six parts, including the users, the user's actual trace, the location privacy-preserving mechanism, the observable trace, the adversary's tracking model, and privacy metrics.

Users and Actual Traces require protection to preserve the user's identity and location privacy. The users generate a vector of triplets $\langle u, r, t \rangle$ where u, r, and t represent the user, location, and time, respectively. The vector generated by a user is, therefore, the actual trace of the user, a. The authors also highlighted that a set of users forms the total set of users, $\mathcal{U}$. A set of locations can be considered as a region of interest, $\mathcal{R}$. Lastly, a set of all actual traces is represented as $\mathcal{A}$.

Location Privacy-Preserving Mechanism (LPPM) transforms the users' actual traces into observable traces to preserve location privacy. Shokri et al. highlighted that LPPM can be done in different ways based on the objective of privacy preservation. For instance, a centralised offline LPPM can be used for public dataset generation. On the other hand, users can use a decentralised

online LPPM to preserve their location privacy on the fly.

The LPPM can also be classified into two types, which are location obfuscation and identity anonymisation. Location obfuscation aims to introduce inaccuracy and lower the precision of the location information. This can be achieved by introducing noise, reducing precision, and/or hiding the location. Similarly, identity anonymisation introduces randomness to the user's identity. Therefore, the LPPM's output is $\langle u', r', t \rangle$ where u' and r' are the anonymised identity and obfuscated location, respectively.

Observable Traces are the results from LPPM and are observable by the adversary. Similar to the formation of an actual trace, a vector of $\langle u', r', t \rangle$ represents an observable trace of a user, o. Then, a set of all observable traces forms the total observable traces by the adversary, $\mathcal{O}$. Shokri et al. assumed that the adversary knows that these traces are not the users' actual traces and that the adversary also knows the LPPM. Therefore, given a known LPPM and the observable traces, the adversary aims to retrieve the users' set and actual traces.

Adversary Model describes the objectives, the knowledge of the adversary, and their tracking model. Adversary objectives can be classified into multiple types, including *tracking* of users, identifying *presence/absence* of a user in a certain location, and identifying *meeting disclosure* of two users. The adversary also incorporates noisy or incomplete actual traces as training sets to build users' mobility profiles, which will be used in the tracking model. Since the LPPM introduces randomness to a triplet $\langle u, r, t \rangle$, the tracking model probabilistically reverses the LPPM functions, using the Bayesian approach as the main concept in their work.

Privacy Metrics measure the failure of the adversary to achieve their objectives. While tracking is presented as one of the adversary objectives, the authors focused on the *presence/absence* and *meeting disclosure* objectives, which translate into localisation, aggregated presence, and meeting disclosure attacks. The privacy metric then measures the incorrectness of the attempt of

these attacks.

2.3.2 Privacy Metrics

Apart from Shokri et al. metrics, which focus on the incorrectness of the attack on location privacy, there are other privacy metrics proposed in the past few decades that are related to VANETs [8, 50, 55]. It is also worth noting that ETSI provided a set of privacy metrics in the pre-standardisation document [25]. Due to the abundance of privacy metrics, Wagner classified these privacy metrics into five groups based on the metric's objective and compared the metrics' strength [49]. These groups of privacy metrics are as follows:

Uncertainty metrics refer to metrics that quantify a level of indistinguishability between users. A higher amount of adversary uncertainty implies higher privacy preservation. Example metrics in this group include anonymity set size and entropy-based metrics.

Information gain/loss metrics aim to quantify the difference between the adversary's prior knowledge of the users and the knowledge after a privacy attack. Metrics in this category include mutual information, Pearson's correlation coefficient, and an increase in the adversary's belief.

Adversary's success probability metrics quantify the level of the adversary's success in identifying the users. This group of metrics highly depends on the parameters given to the metrics.

Error-based metrics focus on the difference between the adversary's tracking outcome and the ground truth. The measurement can be in distance-based quantification or incorrectness in identifying the user.

Time-based metrics, as the name suggests, measure the amount of time, which can come in many forms. The common approach for time-based metrics is the length of time that the adversary can track a user.

Wagner used monotonicity as a criterion to evaluate the strength of the metrics. The monotonicity measures the difference in the metric's outcomes based on different levels of the adversary's strength. The author highlighted

that, although some of the metrics may have high monotonicity, a strong desirable metric, using a combination of metrics is more desirable. The privacy evaluator can quantify the level of privacy from multiple perspectives by combining metrics from different groups. This finding can be used to support studies in [8, 50, 82] where multiple metrics have been used together for privacy quantification. The following are notable works on location privacy and VANET privacy quantification.

Ma et al. [55] are among the first works in VANET privacy quantification. They quantify the entropy between the trip's origin, destination, and the user's identity. The authors refer to their work as *"Hub and Spokes model"*, where a hub represents an individual, and each spoke represents a probability that the user made a trip from origin j to destination k. With the hub and spokes, the entropy and normalised entropy can be calculated to account for the adversary's uncertainty of a user making a certain trip. The authors also extended their model by connecting multiple hubs to form a consecutive user tracking [84]. The extended version of the hub and spokes can also recalculate the probability at the spokes based on past information to reduce uncertainty. The recalculation process can be done in three ways: averaging the probabilities, frequency-based, or Bayesian-based.

Tracking percentage and an average number of confusions are used by Emara et al. as privacy metrics in [8]. The first metric measures the maximum percentage of time-based tracking in which the adversary can correctly track the vehicles, i.e., the metric evaluates the performance of the tracking model. That means the lower the tracking percentage is, the better for the VANET users.

$$Tracking\ Percentage\ =\ \frac{\sum_v \max_t l(t,v)}{\sum_v L(v)} \tag{2.1}$$

where, expressed in the terms from previous subsection, $l(t, v)$ is the length of the time when the actual trace v is assigned to the observable trace t and $L(v)$ is the lifetime of the vehicle actual trace v. Another metric, the average number of confusions, is used as a privacy metric in [8]. The metric measures the

average number of times the vehicle tracker assigned two consecutive messages from the same vehicle to different tracks. Therefore, the higher average number of confusions reflects better privacy preservation.

$$Average\ Number\ of\ Confusions\ =\ \frac{1}{N}\sum_{i}^{N}\sum_{k}^{L(v_i)}C_{i,k} \tag{2.2}$$

$$C_{i,k} = \begin{cases} 1 & T_{k-1}(v_i) \neq T_k(v_i) \\ 0 & otherwise \end{cases}$$

where $T_k(v_i)$ is the observable trace assigned to the message of vehicle v_i at time step k, N is the number of vehicles, and $L(v_i)$ is the lifetime of the vehicle.

Lastly, the KDT-based model comprises anonymity set size, distance-based, and time-based metrics [50]. Corser et al.'s model works around the anonymity set size, in which the vehicle privacy is preserved if the anonymity set size is greater than 1, i.e., the vehicle is indistinguishable from the others. They argued that, due to the constant change in the number of vehicles, the anonymity set size should be accounted for by duration and not a single time instance. Corser's time-based metric accounts for the duration where the anonymity set size is greater than 1, which is called *"Anonymity duration (|T|)"*. Their second metric is *"Average Anonymity Set Size (K)"*, which accounts for the average number of indistinguishable vehicles in the anonymity set size over the duration that the vehicle's anonymity set size is over 1.

$$K\ =\ \frac{K_0 + K_1 + \cdots + K_{|T|}}{|T|+1} \tag{2.3}$$

Lastly, Corser's distance-based metric accounts for an *"average distance deviation (D)"*, i.e., the distance between vehicles in the same anonymity set size.

$$D\ =\ \frac{\bar{d}_0 + \bar{d}_1 + \cdots + \bar{d}_{|T|}}{|T|+1} \tag{2.4}$$

where $\bar{d}_j$ is an average distance deviation at time t_j and is calculated by

$$\bar{d}_j\ =\ \frac{1}{k_j}\frac{1}{k_j}\sum_{s=1}^{k_j}\sum_{i=1}^{k_j}p_{sij}d_{sij} \tag{2.5}$$

p_{sij} is the probability that the attacker think that entity s is their taget while entity i is the actual target and d_{sij} is the distance deviation between these entities.

Table 2.3 summarises the privacy metrics discussed in this section. The second column, *Level of Privacy*, shows the relation between the actual privacy of the users and the privacy metric's value, where an up arrow, ↑, means that a higher value of the privacy metric indicates higher privacy preservation for the users, and a down arrow, ↓, means otherwise.

Table 2.3: Privacy metrics presented in the literature.

Privacy Metrics	Level of Privacy
Incorrectness of localisation [85]	↑
Incorrectness of aggregated presence [85]	↑
Incorrectness of meeting disclosure [85]	↑
Hub and Spokes [55]	↑
Tracking percentage [8]	↓
Average number of confusions [8]	↑
Anonymity duration [50]	↑
Average anonymity set size [50]	↑
Average distance deviation [50]	↑

2.3.3 Datasets

The methods for evaluating privacy in VANETs vary between theoretical analysis, public datasets, and simulations derived from the public dataset. Ma et al. evaluated their work with their own generated dataset and theoretical analysis [55]. This method is possible in their work since their metric only requires probability pairs between identity, origin, and destination.

Usage of public datasets in evaluation can be seen in [49, 85]. The characteristics of the public dataset usage method are (1) the dataset can be used as it is, or (2) some data points in the dataset can be removed. Shokri et al. [85]

randomly chose 20 mobilities from the EPFL/mobility dataset. Wagner [49] used taxi traces in Rome to represent city traffic, where the message frequency has been reduced to 15 seconds.

The last evaluation method group uses simulation with a public dataset as the source [8, 49, 50]. The notable characteristics of this method are (1) the environment in the dataset is controllable, (2) the granularity of the data point can be increased, and (3) the dataset contains real-world characteristics. Emara et al. used the VISSIM simulator to generate their dataset, where message frequency, number of vehicles, and simulation duration can be controlled [8]. Corser et al. used the Generic Mobility Simulation Framework (GMSF) in their work, where three scenarios were generated to represent the density of the cars [50]. Another dataset used by Wagner [49] is a synthetic dataset from a highway near Madrid.

It must be noted that none of these datasets can capture every aspect of VANET communication. Most of these works either (1) do not capture all information in the message, or (2) the frequency of the messages is lower than the VANET standards. On the other hand, the real-world VANET communication datasets [86][3] , where most of the information is captured, are unsuitable for the vehicle tracking framework. These findings will be elaborated upon in Section 4.1. The datasets in [86] cannot be used because of the privacy concern for the users, where some data points were removed [74].

Table 2.4 lists publicly available vehicle trace datasets and simulated datasets used in the literature. In the *Type* column, an **R** refers to a real-world dataset, and an **S** refers to a simulated dataset. Additionally, a U refers to an urban scenario, and an H refers to a highway scenario. We can observe that real-world datasets offer a longer duration of captured data than those in simulated datasets. On the other hand, the simulated datasets offer a finer granularity than the real-world dataset and result in a closer message broadcast frequency

[3]The access to the dataset has been changed in 2025, and a request form must be submitted to gain access for the full dataset [87].

used in VANET communication.

Table 2.4: Vehicle trace datasets.

Dataset	Type	Duration (minute)	Map size	Granularity (second)	Traces	Used in
Roma/Taxi [88]	**R** *U*	30 days	$87.62 \times 101.0\ km^2$	15	320	[49]
Madrid highways [89]	**R** *H*	-	10 km	-	2500	[49]
epfl/mobility [90]	**R** *U*	30 days	$25.51 \times 52.60\ km^2$	2	500	[82]
T-Drive [91]	**R** *U*	3 months	$255.4 \times 786\ km^2$	varies	10357	[92]
GeoLife [93]	**R** *U*	3 years	varies	1–5	17621	[92]
Car steering [94]	**R** *U*	-	0.6–31.9 km	-	58	[94]
TAPAS Cologne [95]	**R** *U*	1 day	$28.94 \times 31.27\ km^2$	-	0.4 M	-
Tampa [86]	**R** *U*	545 days	$1.65 \times 1.47\ km^2$	varies	-	-
Wyomming [86]	**R** *H*	3.5 years	$90.84 \times 25.98\ km^2$	varies	-	-
Luxembourg	**S** *U*	5	$850 \times 500\ m^2$	0.1–0.5	25–195	[8]
Luxembourg	**S** *H*	5	$550 \times 500\ m^2$	0.1–0.5	20–64	[8]
Northwest Washington	**S** *U*	60	$2.93 \times 1.6\ km^2$	0.1	1425	[96]
Dulles Access Road	**S** *H*	30	22.5 km	0.1	836	[96]
Hangzhou	**S** *U*	600	$26 \times 13\ km^2$	0.3–0.5	279	[97]
Chengdu	**S** *U*	600	$23 \times 23\ km^2$	0.3–0.5	264	[97]
Beijing	**S** *U*	600	$13 \times 8\ km^2$	0.3–0.5	255	[97]

It must be noted that the privacy quantification literature usually uses only a subset of these datasets. For instance, Wagner [49] used 1800 seconds of data from Roma/Taxi [88] and 1000 seconds of data from Madrid Highway [89]. Another observation on the simulated datasets is that all simulated datasets in Table 2.4 used a real-world map to increase the realism of the datasets. However, the traces in these datasets were generated by the authors; hence, realistic traffic may not be accurately captured in these datasets used in the literature.

2.4 Kalman Filter

The Kalman filter is a recursive process for estimating, smoothing, and predicting the states of a dynamic system, and is known for its storage and com-

putation efficiency [98]. A dynamic system refers to an object, such as a vehicle or aeroplane, whose state evolves over time. The system's state is a set of values that change over time due to the system's behaviour, such as the vehicle's location. While sensors can provide measurements of these states, faulty sensors can introduce inaccuracy between the reported values and the true system state. The Kalman filter mitigates such errors by estimating the system's state from the combination of sensor data, a model of system dynamics, and known noise. The process achieves storage efficiency by representing a dynamic system with a vector that stores only its current state. For computational efficiency, it leverages matrix multiplication to estimate the system state. With the ability to estimate the system state from inaccurate sensor measurements, the Kalman filter becomes a candidate approach to be used for estimating the vehicle's state, i.e., the Kalman filter can be used as a tool for vehicle tracking.

2.4.1 Kalman Filter

Let t be the current discrete time step of a system; smoothing is a process for estimating the system's states from time step 0 to $t-1$. With the system state at $t-1$ and sensors' measurements at t, the Kalman filter can estimate the system's current state, which is considered to be its true state. The current state estimation process is called $filtering$. Lastly, the *prediction* process estimates the system's state at time $t+1$ using its current state and a transformation matrix. Figure 2.3 illustrates the Kalman filter process when it is used for estimating the system state. Note that the Kalman filter is *smoothing* the system states when the process in Figure 2.3 is used between the time 0 and $t-1$. Then, the $filtering$ process is achieved from the transition from the previous state $(t-1)$ to the measurement value. Lastly, with the delay, the system state at t becomes the previous state, and the *prediction* occurs at a transition from the previous state to the current state.

To explain the Kalman filter process, let us begin at the *Previous state* v_{t-1},

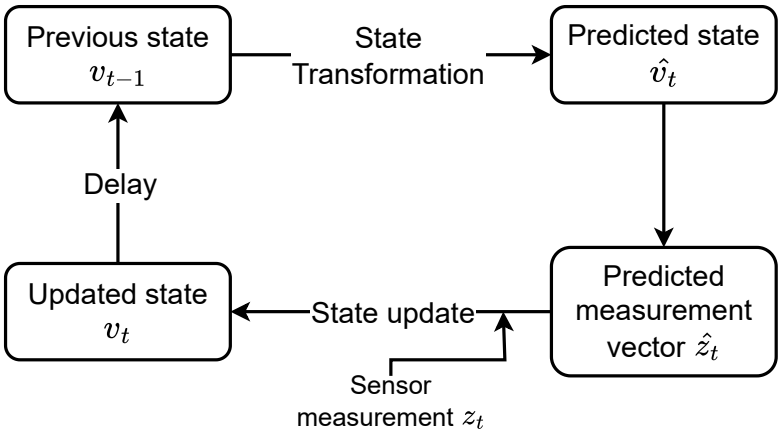


Figure 2.3: Processes in the Kalman filter.

which is the initial state of a dynamic system. The system state represents the dynamic system as a vector whose elements are the system dynamics of interest. The *Predicted state* $\hat{v}_t$ is calculated by applying a transformation matrix to the vector v_{t-1}. The transformation matrix, T, represents our understanding of how each system's dynamics change over time and how they influence the other system's dynamics. Furthermore, the Kalman filter accounts for errors that could occur during the state transformation due to process noise, w_t. The process noise represents accountability for uncertainty and unmodeled dynamics in the state transition with values drawn from a Gaussian distribution $\mathcal{N}(0, Q)$, where Q is the covariance of the process noise. The state transformation can be expressed as follows:

$$\hat{v}_t = Tv_{t-1} + w_t \tag{2.6}$$

The predicted state $\hat{v}_t$ represents our estimate of the system's state in the next time step and does not incorporate any measurements from the sensors. To incorporate the sensor measurement, we must calculate the innovation and the Kalman Gain, which will be used to update the predicted state.

The innovation is the residual between the sensor measurements z_t and the predicted sensor measurements $\hat{z}_t$. Notice that the innovation uses the predicted sensor measurements $\hat{z}_t$, not the predicted state $\hat{v}_t$. The predicted state $\hat{v}_t$ can be mapped to a predicted measurement $\hat{z}_t$ with an observation

matrix H. The observation matrix is helpful when the system state vector is larger than the measurement vector. For instance, the sensors only measured two values, but the system state also considered the kinematics that affected those values. The observation matrix H can map a predicted system state vector $\hat{v}_t$ into a predicted sensor measurement vector $\hat{z}_t$ as follows:

$$\hat{z}_t = H\hat{v}_t$$

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ \dot{x} \\ \dot{y} \end{bmatrix}$$

With the predicted sensor measurements $\hat{z}_t$, the innovation y_t can be computed as follows:

$$y_t = z_t - \hat{z}_t = z_t - H\hat{v}_t \tag{2.7}$$

Additionally, we can also calculate the covariance of the innovation, which represents the uncertainty of the predicted measurements.

$$S = HQH^T + R \tag{2.8}$$

Where Q is the covariance of processing noise, H is the observation matrix, and R is the covariance of the measurement noise.

Then, the innovation and the Kalman Gain are used to correct the predicted state. The Kalman Gain has two objectives: projecting the innovation from measurement to the system state and balancing the weight between the estimated vector and the measurement vector. The Kalman Gain can be computed as follows:

$$K_t = QH^T S^{-1} \tag{2.9}$$

Then, the Kalman Gain can be used to correct the predicted state $\hat{v}_t$ as follows:

$$v_t = \hat{v}_t + K_t y_t \tag{2.10}$$

From Equation (2.9), it can be observed that the Kalman Gain is affected by processing noise Q and measurement noise R. As a result, in Equation (2.10),

the Kalman Gain acts as a weight between trusting the predicted value $\hat{y}_t$ or the measurement value through correction with innovation. When the processing noise Q is high, the Kalman Gain becomes larger, and the corrected state weighs more towards the correction from the innovation. On the other hand, when the measurement noise R is high, the Kalman Gain becomes small, and the corrected state is weighted more towards the predicted state. Lastly, as time passes, the state v_t is delayed and becomes the state in the past v_{t-1}, in which the Kalman filter cycle can continue from Equation (2.6).

We now detail the construction of the process noise covariance matrix Q and the measurement noise covariance matrix R. The measurement noise covariance R quantifies the uncertainty in sensor measurements. Its values can typically be obtained from sensor datasheets or derived from real-time sensor performance metrics, such as reported measurement uncertainty or empirical calibration. In contrast, the process noise covariance Q models the uncertainty in the system's dynamic transformation and generally cannot be directly measured. Instead, it must be estimated and adaptively updated at each time step.

At the initial time step, Q should be estimated based on the variance of the system variable that influences other variables. For example, in a Kalman filter applied to motion estimation in a 1D Cartesian plane where location and velocity are modelled in the system state, the system state can be expressed as follows:

$$\begin{bmatrix} location & speed \end{bmatrix}^T$$

Since position evolves from velocity, and velocity evolves from acceleration, the primary source of uncertainty in such a system stems from random acceleration. Modelling uncertainty in acceleration enables the propagation of process noise to both position and velocity, capturing the true dynamic variability of the system. Therefore, the initial covariance of the process noise of this system

can be expressed as follows:

$$Q_0 = \sigma_a^2 \begin{bmatrix} \frac{\Delta t^4}{4} & \frac{\Delta t^3}{2} \\ \frac{\Delta t^3}{2} & \Delta t^2 \end{bmatrix} \tag{2.11}$$

where σ_a^2 is the variance of the acceleration and Δt is the time step duration.

With the initial covariance of the process noise, the covariance can be updated after Equation (2.10) at each time step as follows:

$$Q_{t+1} = (I - K_t H)Q \tag{2.12}$$

where K_t is the Kalman Gain and H is an observation matrix.

2.4.2 Mahalanobis Distance

Mahalanobis distance is a statistical distance metric that can be used in classification, clustering, and outlier detection [99]. A statistical distance measures the similarity or dissimilarity between two objects while also considering the certainty in the objects through variance and correlation. The Mahalanobis distance is not only a well-known method for outlier detection, but it is also often used with the Kalman filter. With the ability to take vectors as input for distance measurement, the Mahalanobis distance is a suitable metric for use with the Kalman filter, where vectors represent the system state and sensor measurements.

The Mahalanobis distance can be utilised in the Kalman filter during the track maintenance process, where a decision must be made to use a sensor measurement for updating the system state. For example, a traffic operator has detected an unknown moving vehicle on the road. While a system state representation of the unknown vehicle can be modelled and the traffic operator can constantly detect the presence of the vehicle, the traffic operator can not be certain that the sensor measurement belongs to the unknown vehicle. The Kalman filter's track maintenance process determines whether the sensor measurement corresponds to the unknown vehicle or if the unknown vehicle

has stopped and the sensor has detected a new vehicle. If the Mahalanobis distance between the predicted state and the sensor measurement is too high, the traffic operator may decide that the sensor measurement represents another vehicle.

Another reason Mahalanobis distance is a suitable distance metric for the Kalman filter is its ability to reuse the innovation and the covariance of the innovation from Equations (2.7) and (2.8). Mahalanobis distance can be calculated as follows:

$$d_{ij}^2 = y_t^T S^{-1} y_t \tag{2.13}$$

where y_t is the innovation from Equation (2.7) and S is the covariance of the innovation from Equation (2.8). The key distinction between Mahalanobis distance and Euclidean distance is the use of innovation covariance S, which normalises the distance based on the uncertainty between the predicted state and the sensor measurement.

Suppose the track maintenance decision must be made; the Mahalanobis distance will be calculated before the Kalman Gain in Equation (2.9) to decide whether to use the sensor measurement to update the predicted state. The decision to drop the sensor measurement can be achieved using a gating mechanism, where the measurement is dropped if the Mahalanobis distance exceeds a certain value. Since the Mahalanobis distance follows the chi-squared distribution, the gating value can be chosen from the chi-squared distribution. Let n be the dimension of the sensor measurement z_t used in Equation (2.7); the Mahalanobis distance between the sensor measurement z_t and the estimated sensor measurement $\hat{z}_t$ will follow the chi-square distribution with n degrees of freedom: $d^2 \sim \mathcal{X}^2(n)$. Then, the chi-square's confidence level α can be used to reject the sensor measurement. For example, if the 95% confidence level is used, the sensor measurement will be rejected when the Mahalanobis distance is larger than the chi-square confidence value: $d_{ij}^2 > \mathcal{X}_{0.95,n}^2$

Chapter 3

Analysis of Privacy in VANETs

This chapter presents our systematic analysis of privacy threats in VANETs. The chapter begins by analysing privacy threats in VANETs using the LINDDUN framework presented in Section 3.1. This analysis provides a list of privacy threats in VANET communication based on the real-world implemented architecture, the Architecture Reference for Cooperative and Intelligent Transportation (ARC-IT) [60]. The identified privacy threats are grouped into four stages of VANET communication, namely, pre-anonymisation data, privacy-preserving mechanisms, data dissemination, and the adversary model, based on the entity responsible for triggering them, as outlined in Section 3.2. We further analyse these stages of VANET communication and formalise each stage into mathematical equations for quantifying the privacy in VANETs through the means of a vehicle tracking model. In Section 3.3, we provide a compatibility analysis between the VANET privacy quantification frameworks presented in the literature and the analysis presented in Section 3.2.

3.1 Privacy Analysis with LINDDUN

While there are examples of using the LINDDUN framework in the literature for privacy threat elicitation in VANETs [79–81], these works either did not focus on identifying privacy threats specific to VANET communication [79] or did not follow the LINDDUN framework's threat elicitation process [80,

81] as highlighted in Section 2.2. Chah et al. [79] presented their work as privacy threat elicitation for connected and autonomous vehicle systems. The architecture used to build the Data Flow Diagram (DFD) did not align with the VANET architecture as presented in Section 2.1.1. On the other hand, Raciti and Bella mentioned in their work that the LINDDUN framework was used in [80, 81]. However, only 'Unawareness' and 'Non-compliance' privacy threats were listed in their work. The authors also did not provide the DFD in either piece of their work. Since creating the DFD is the first step in the LINDDUN privacy threat elicitation framework, their methodology and results are debatable. The incompleteness in the literature on applying the LINDDUN framework results in gaps in identifying privacy threats in VANET communication, which can be summarised into 3 parts.

- A data flow diagram that did not align with VANET's communication architecture.
- Only Unawareness and Non-compliance threats were analysed.
- Incomplete privacy threat documentation.

It should also be noted that the literature on applying the LINDDUN framework on VANET privacy quantification presented in Section 2.2.2 is focused on Linking privacy threat but lacks documentation according to the LINDDUN framework.

It is crucial to follow the LINDDUN privacy threat elicitation framework as the process could potentially identify undocumented threats. Recall that the LINDDUN framework has three steps for threat elicitation: (1) creating a Data Flow Diagram (DFD), (2) mapping the DFD elements with LINDDUN threats, and (3) identifying the privacy threats with threat tree patterns and documenting the threats. This section provides VANET privacy threats analysis using the LINDDUN framework with a DFD from the Architecture Reference for Cooperative and Intelligent Transportation (ARC-IT) [60]. Note that the ARC-IT provides multiple data flow diagrams based on service packages

and processes. However, this thesis will focus on the 'vehicle' physical object, as the objective is to identify privacy leakage that could occur to VANET users and explore related data flows. Our objective is to identify privacy threats in VANET communication according to the ARC-IT, which is used in the US, along with the standards governing VANET communication, as presented in Section 2.1.

We begin the section by listing the service packages [62], functionalities [100], and triple source and destination [63] in ARC-IT that are related to the *vehicle* physical object. Figure 3.1 and 3.2 show the mapping of *vehicle* and its related service packages, functionalities, and the source and destination of data flow, along with the security class notations. The figures provide complete details of the service packages that the vehicle will use, the functionalities that the vehicle must have, and the information flow from and to the vehicle. To understand the security class notations, each service package is assigned a security class, where device security class 1 has the lowest security requirements and class 5 has the highest security requirements. Each service package has one or more functionalities, which we assume carry the security class of the service package to which they are connected. Then, each functionality has one or more triples, representing data flow from one physical object to another. We also assume that the triple source and destination carry the security class from the functionality to which they are attached. Lastly, while the service package has one security class attached to it, the functionality and triple may have multiple security classes attached to them, depending on the service packages to which they are attached. The functionality and triple's security class in Figure 3.1 and 3.2 are represented with the highest security class. The full list of functionality and triple's security classes are presented in Tables B.2, B.3, and B.4.

According to the security classes in Tables B.2, B.3, and B.4, we observed that if all service packages are implemented, some of the functionalities and the physical object's security class should be considered higher than the security

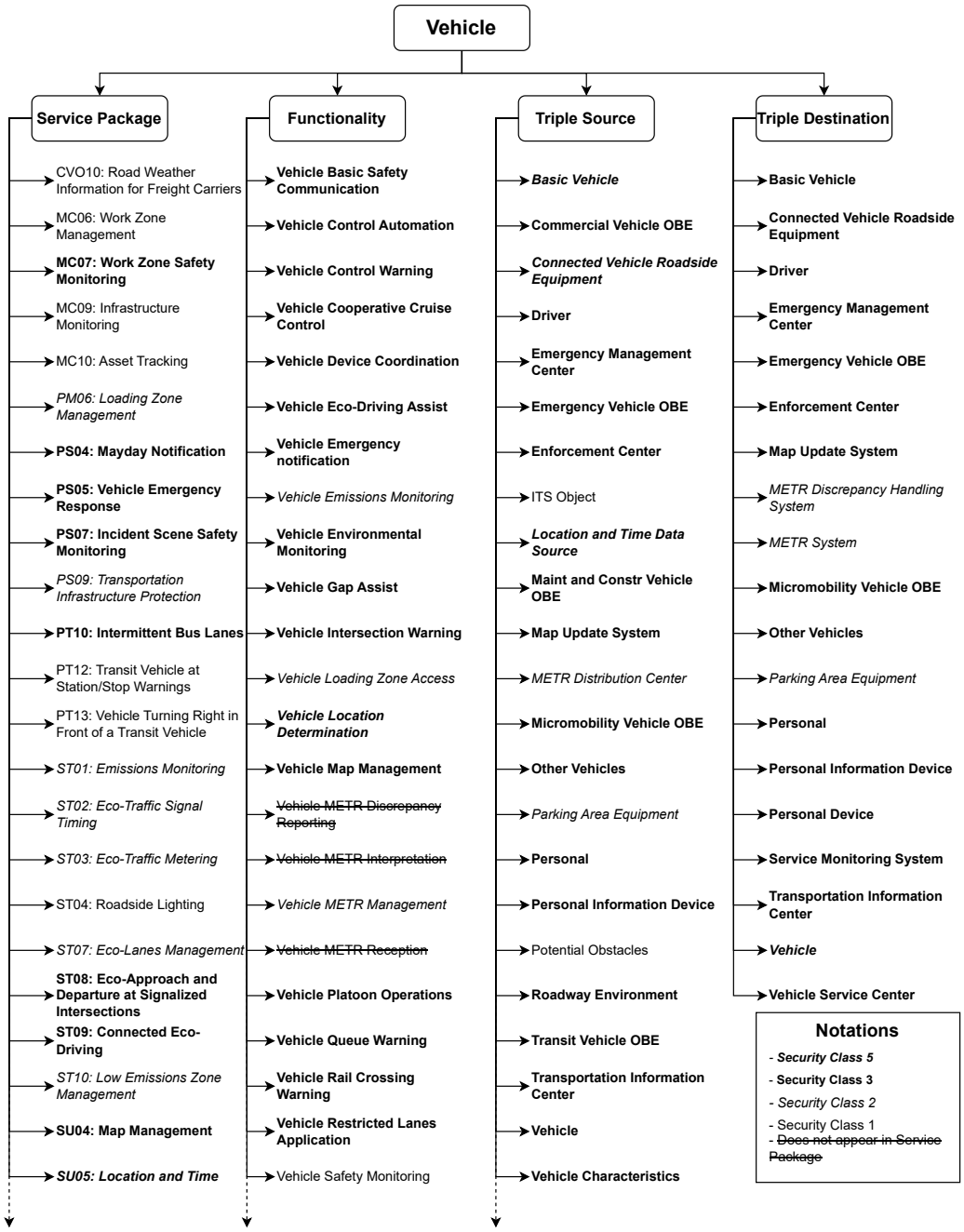


Figure 3.1: Mapping of vehicle physical object.

class assumed from the service package. For example, the *Vehicle Basic Safety Communication* functionality is used in service packages, which have a security class range of 1 to 3. By considering the highest security class, the *Vehicle Basic Safety Communication* functionality should be classified as a security class 3, rather than classes 1 or 2. Similarly, the *Basic Vehicle*, *Connected Vehicle Roadside Equipment*, and *Vehicle* physical objects should be consid-

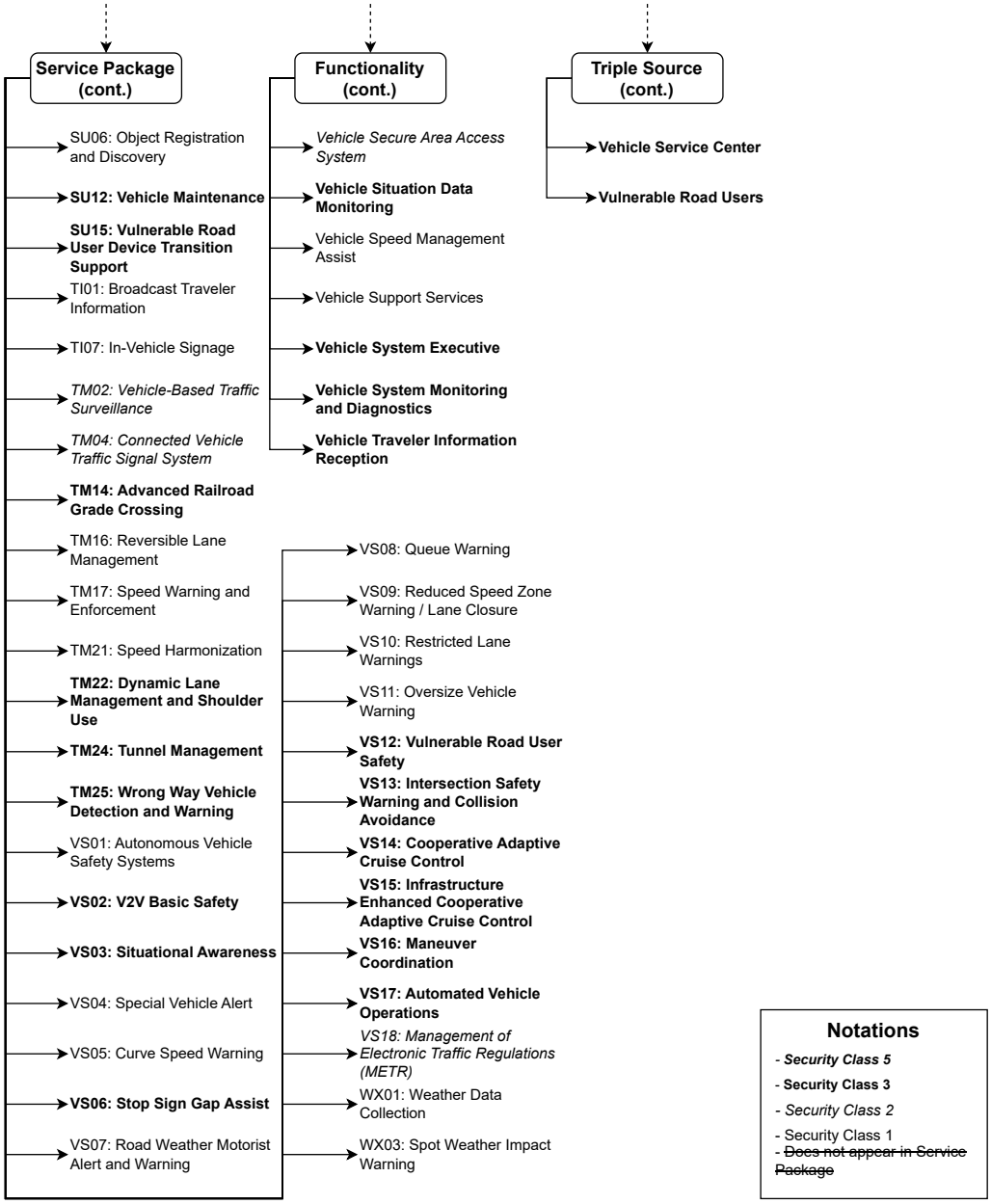


Figure 3.2: Mapping of vehicle physical object. (Cont.)

ered security class 5, as they utilise the *Location and Time* service package. Another observation is that some of the functionalities are mentioned in the vehicle's functionality [100] but have not been used by any service packages. Due to not being used by service packages, we cannot assign the related security class to these functionalities; therefore, they appear in Figure 3.1 and 3.2 as *strike-through* text. Knowing the security class can help focus on protecting service packages, functionalities, and physical objects that pose a high secu-

rity risk. Unfortunately, the ARC-IT's security class is only considered for the *data security* and does not account for the *data privacy*. This highlights the shortcomings of privacy concerns in the ARC-IT.

We are now beginning the LINDDUN privacy threat elicitation process by creating a data flow diagram based on the vehicle's interfaces diagram from ARC-IT [101]. Note that, due to a limitation of information from ARC-IT, the data flow is only represented by physical objects and the data exchanged between them. Figure 3.3 reflects diagrams in [101] and combines multiple physical objects into groups. We also combine the information exchanged between these physical objects based on the interfaces through which communication is conducted. Additionally, the human interface with the *driver* physical object is removed, as the adversary cannot intercept the information. Overall, there are five interfaces with nine groups of entities, excluding the vehicle. The interface is represented by a number on the data flow, which includes wide area wireless, short range wireless, vehicle on-board, position location interface, and environment interfaces. The separation of entity groups is determined based on the communication interface and the data flow between the vehicle and other physical objects.

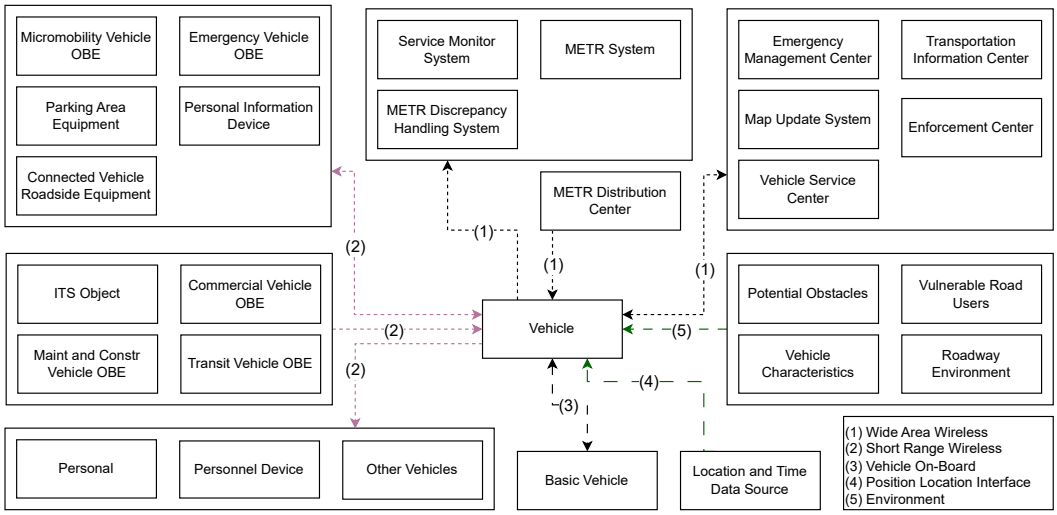


Figure 3.3: Data flow of vehicle based on ARC-IT

Let us consider Figure 3.3 with the privacy threats mapping in Table 2.2.

There will be only two data flow diagram elements to be considered, which are *entity* and *data flow*. Nevertheless, these two elements cover all privacy threats based on the LINDDUN framework.

Next, privacy threats can be identified using the threat tree patterns listed in Appendix C. We will explore privacy threats based on their triggered sequence; that is, the privacy threat that does not have another threat as a trigger will be presented first. Each privacy threat will be documented in a threat template in the Listing environment as presented in [77]. To provide consistent threat identification, the threat caption is presented as "<*Entity*>, <*Threat identifier*> <*Threat name*>" where:

- <*Entity*> is the physical object in which the privacy threat arises. The physical object is shown in Figure 3.3.
- <*Threat identifier*> is the identifier of the threat, appearing on the first row of the node in threat tree patterns in Appendix C.
- <*Threat name*> is the threat name showing after the threat identifier in the node on the threat tree patterns.

Let us walk through the first privacy threat identification, *unaware of the data that the vehicle sends*. From Table 2.2, we identify that the *vehicle*, which is an entity in a DFD, is prone to *unawareness* privacy threat.

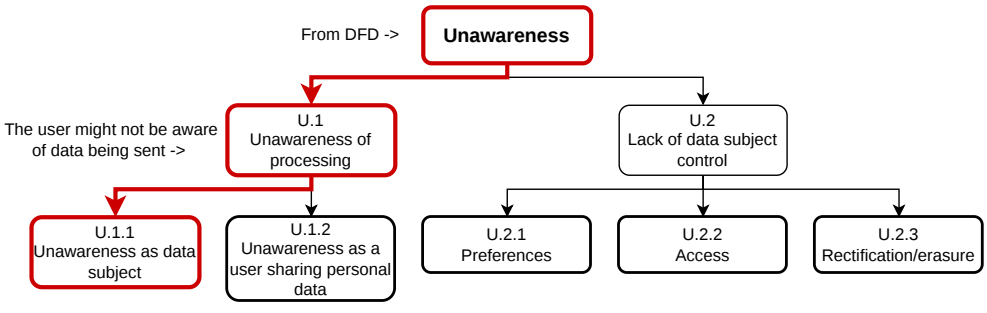


Figure 3.4: Example of using LINDDUN's privacy threat tree.

Then, we consult with the unawareness threat tree patterns in Figure 3.4, beginning at the root node. Traversing to node U.1, we assume that the driver,

who controls the vehicle, may not be aware of the information sent by the vehicle; therefore, the privacy threat in U.1 remains valid. Since the node U.1 is not a leaf node, we traverse further to its first child node, U.1.1 *unawareness as data subject*. With the previous assumption, the driver of the car can fall into the category of a data subject, thereby making the privacy threat in node U.1.1 possible. Since node U.1.1 is a leaf node, we document the threat as shown in Listing 3.1. With the *vehicle*'s privacy threat being explored, the <*Entity*> in our threat identifier is *vehicle*. The <*Threat identifier*> and <*Threat name*> are taken from the current leaf node we are at, which is *U.1.1 Unawareness as data subject*. The remainder of this subsection lists the privacy threats that we identified with a similar process as we walked through.

```
Unaware of the data that vehicle send
    Summary: To receive the benefits from VANET's applications,
    the vehicle must participate in VANET's communication,
   including occasionally broadcasting its location and
   kinematics. The driver may not be properly informed about
   the information sent by the vehicle.
    Basic Flow:
    (1) The user participates in VANET's communication,
   unknowingly the information sent by the vehicle.
    Leaf Nodes: U.1.1 unawareness as data subject.
    Root Nodes: Unawareness.
    Remarks: Although the user is willing to trade privacy for
   VANET's utilities and benefits [11], an education level
   higher than the last year of high school might be required
   in order to comprehend the privacy policies [10].
```

Listing 3.1: Vehicle, U.1.1 unawareness of processing as data subject threat.

```
Lack of control of preference over the sending data
    Summary: The driver/vehicle has no options for their
   preference on the data that must be sent and how it is being
    processed. This may include the ability to choose the
   pseudonym-changing mechanism.
    Basic Flow:
```

```
(1) Information, such as location and kinematics, must be
broadcast with the message. The driver cannot choose which
information should not be sent with the message.
Leaf Nodes: U.2.1 Preferences.
Root Nodes: Unawareness.
Remarks: This privacy threat highlights the conflict
between the system's utilities and the user's privacy.
```

Listing 3.2: Vehicle, U.2.1 lack of data subject control over preferences threat.

We put Listing 3.1 and 3.2 at the beginning of our threat elicitation since they do not have a prerequisite nor the primary adversary. However, they may constitute the following privacy threats. Listing 3.2 also highlights the beginning of privacy problems in VANETs, where the user must comply with standards to receive the benefits from the applications. The privacy threat in Listing 3.1 also corresponds to privacy concerns presented in news sources as presented in [16, 102], where the vehicle users may not be aware that their data is being collected and processed.

```
Every message in VANETs is signed.
Summary: The Basic Safety Messages and Cooperative Awareness
Messages are signed with a pseudonym certificate to ensure
data integrity. The pseudonym certificate or its digest will
also be attached to the messages.
Basic Flow:
(1) The vehicle signs the message with a certificate.
(2) The pseudonym certificate or its digest is attached to
the message.
Leaf Nodes: Nr.1.2 signed data.
Root Nodes: Non-repudiation.
Remarks: This privacy threat highlights the conflict
between the system's security and the user's privacy.
```

Listing 3.3: Data Flow, Nr.1.2 signed data threat.

```
VANETs communication is unencrypted and observable.
Summary: The Basic Safety Messages and Cooperative Awareness
Messages are observable by all entities within the range of
```

```
communication. These messages are also not encrypted due to
their relation with safety applications in VANETs [21].
Primary Adversary: Entities that receive short range
wireless interface, highlighted as (2) in Figure 3.3, e.g.,
other vehicles and connected vehicle roadside equipment, can
detect the presence of the vehicle.
Basic Flow:
(1) The vehicle broadcasts its location and kinematics to
nearby entities with BSM or CAM.
(2) The nearby entities detect the vehicle's presence
through the received message.
Leaf Nodes: D.1 observed communications.
Root Nodes: Detectability.
Remarks: Although VANET's standards state that the
receivers will not forward these messages, the public
dataset from THEA highlights that data collection and
forwarding are possible. The characteristics of this dataset
will be presented in Section 4.1.
```

Listing 3.4: Data Flow, D.1 observed communications threat.

```
Fine-grained location and kinematics are being broadcast.
Summary: Fine-grained data is broadcast to sustain safety
applications like crash avoidance systems.
Primary Adversary: Entities that receive short range
wireless interfaces.
Basic Flow:
(1) The vehicle broadcasts BSMs/CAMs to participate in
VANET communication and to use the safety applications.
(2) The adversary detects detailed information about the
vehicle, e.g., a precise location of the vehicle through the
fine-grained latitude and longitude.
Leaf Nodes: DD.1.2 data type granularity.
Root Nodes: Data disclosure.
Remarks: The full list of information and its granularity
is presented in Tabel A.1. It must be noted that there are
```

```
differences in the granularities between BSM and CAM.
```

Listing 3.5: Data Flow, DD.1.2 data type granularity threat.

```
A Message is being broadcast, at least, every second.
 Summary: The vehicle must broadcast its location and
kinematics at 1 − 10Hz, generating a tremendous amount of
vehicle data.
 Primary Adversary: Entities that receive short range
wireless interfaces.
 Basic Flow:
 (1) For every 100 ms, the vehicle checks whether a new
message should be broadcast.
 (2) The adversary receives more messages if the vehicle's
locations and kinematics change more frequently.
 Leaf Nodes: DD.2.2 frequency.
 Root Nodes: Data disclosure.
 Remarks: The broadcast frequency depends on the rate of
changes in the vehicle's location and kinematics, i.e., the
BSMs and CAMs will be broadcast more frequently when the
vehicle moves at a higher speed than when it is stuck in
traffic.
```

Listing 3.6: Data Flow, DD.2.2 frequency threat.

Privacy threats in Listing 3.3 – 3.6 can be considered, stemming from the threats in Listing 3.2 since the user has no control over their preferences in the communication. We must also note that VANET communication is designed to support the safety and efficiency of road users; hence, the system designer prioritises the system's utility and security. Therefore, the privacy threats in Listing 3.3 – 3.6 also highlight the conflict between the user's privacy and the system's utility or security. The user must accept the privacy risks to gain the benefits of VANET's safety applications or to maintain trust within the system. Although the privacy threats in Listing 3.3 – 3.6 may seem innocuous, these threats can contribute to the following privacy threats that are more severe to the user's privacy.

```
Vehicle's pseudonym as unique identifier
    Summary: The pseudonym certificate and its digest are
   unique identifiers across multiple BSMs or CAMs. The
   adversary can construct a vehicle profile based on the
   locations of the messages.
    Primary Adversary: Entities that receive short range
   wireless interfaces.
    Basic Flow:
    (1) The adversary receives messages from the vehicle and
   looks at the pseudonym certificate or the certificate digest
   .
    (2) The certificate or the certificate digest is used as an
    identifier to map with the previously known certificate.
    Preconditions: The vehicle must participate in VANET
   communication and broadcast BSM or CAM to the surrounding
   entities.
    Leaf Nodes: L.1.1 unique identifier.
    Root Nodes: Linkability.
    Remarks: Although the pseudonym certificate will be changed
    over time, messages with the same pseudonym are susceptible
    to this threat.
```

Listing 3.7: Vehicle, L.1.1 unique identifier threat.

The privacy threat in Listing 3.7 highlights the beginning of the vehicle tracking threat. As stated in the threat card, this privacy threat only focuses on a single pseudonym; therefore, tracking duration depends on the pseudonym usage, i.e., a frequently pseudonym-changing mechanism will result in a shorter duration of vehicle tracking. This also means that an entire trip will be exposed if the vehicle does not change its pseudonym during the commute. Moreover, the adversary may also learn the driving characteristics from the statistics of the set of messages with the same pseudonym.

```
Vehicle's width and length can be used as quasi-identifier.
    Summary: Since the vehicle's width and length are constants
   , they remain the same even though the pseudonym was changed
   . These constant variables can be used as the quasi-
```

```
identifier across multiple pseudonyms.
 Primary Adversary: Entities that receive short range
wireless interfaces.
 Basic Flow:
 (1) The adversary receives a message from the vehicle.
 (2) The adversary queries other messages with the same
width and length.
 Preconditions: The vehicle must participate in VANET
communication and broadcast BSM or CAM to the surrounding
entities.
 Leaf Nodes: L.2.1.1 quasi-identifier combining data of a
single individual.
 Root Nodes: Linkability.
 Remarks: This threat entirely depends on the uniqueness of
the vehicle's width and length. Therefore, a common vehicle
model can be less vulnerable to this threat than a vehicle
with a unique width and length.
```

Listing 3.8: Data Flow, L.2.1.1 quasi-identifier combining data of a single individual threat.

```
Linking two pseudonyms with path history points
 Summary: According to SAE and ETSI, the vehicle's historic
locations are attached in the messages through path history
points. The similarity between the vehicle's location and
path history points can be assessed and used to link two
different pseudonyms.
 Primary Adversary: Entities that receive short range
wireless interfaces.
 Basic Flow:
 (1) The adversary receives a message from the vehicle and
calculates its historic location with path history points.
 (2) The adversary queries other messages with the same
locations as the calculated location from step (1).
 Preconditions: The vehicle must participate in VANET
communication and broadcast BSM or CAM to the surrounding
entities.
```

```
Leaf Nodes: L.2.2.3 profiling an individual through (dis)
similarity.
Root Nodes: Linkability.
```

Listing 3.9: Data Flow, L.2.2.3 profiling an individual through (dis)similarity threat (with path history).

```
Linking two pseudonyms with calculated location and kinematics
Summary: The locations and kinematics in the messages are
used to predict the vehicle's next location and kinematics.
The similarity between the predicted location and kinematics
, and the location and kinematics in a different message of
a different pseudonym, can be used to assess the likelihood
of the two pseudonyms being the same vehicle.
Primary Adversary: Entities that receive short range
wireless interfaces.
Basic Flow:
(1) The adversary receives a message from the vehicle and
predicts the vehicle's next location and kinematics.
(2) As the adversary receives a new message, the location
and kinematics in the new message are compared with the
predicted location and kinematics from step (1).
(3) The adversary concludes that two messages are generated
 from the same vehicle if the similarity exceeds a certain
threshold.
Preconditions: The vehicle must participate in VANET
communication and broadcast BSM or CAM to the surrounding
entities.
Leaf Nodes: L.2.2.3 profiling an individual through (dis)
similarity.
Root Nodes: Linkability.
```

Listing 3.10: Data Flow, L.2.2.3 profiling an individual through (dis)similarity threat (with predicted location and kinematics).

Notice that the adversary can combine the linkability threats to create a more efficient privacy attack. The adversary can use Listing 3.7 as their main approach to link multiple messages of the same pseudonym together. Then,

the adversary may use Listing 3.8–3.10 to link different pseudonyms together, creating a longer vehicle tracking. In Listing 3.10, the approach for predicting the vehicle's next location and kinematics and associating two different pseudonyms depends on the adversary. We will explore two different prediction approaches and one approach for the threat in Listing 3.9 in Chapter 5.

```
Deanonymisation of pseudonym's owner
    Summary: Since the pseudonym certificate is unique to the
    vehicle, it can be used to identify the user who owns the
    pseudonym directly.
    Primary Adversary: Linkage authorities in the security
    credential management system.
    Basic Flow:
    (1) Two linkage authorities share the butterfly key
    expansion secrets and the pseudonyms generated for a vehicle
    .
    Leaf Nodes: I.2.1.1 pseudonym identifier.
    Root Nodes: Identifiability.
    Remarks: To identify the user's identity with pseudonyms,
    linkage authorities in the security credential management
    system must collaborate with each other.
```

Listing 3.11: Vehicle, I.2.1.1 pseudonym identifier threat.

The privacy threat in Listing 3.11 highlights an honest-but-curious adversary type where linkage authorities perform their duties correctly but also try to violate the user's privacy through de-identification. However, proving the possibility of this threat is out of the scope of this thesis.

```
Usage of locations as vehicle's quasi-identifier
    Summary: The combination of the vehicle's locations can be
    used as a quasi-identifier for the vehicle, especially the
    beginning and destination of the trip, where they can be
    implied as house and office locations.
    Primary Adversary: Entities that receive short range
    wireless interfaces in a wide range, such as connected
    vehicle roadside equipment.
    Basic Flow:
```

```
(1) The adversary collects messages sent by the vehicle.
(2) The adversary locates the starting and destination
locations of the vehicle's trip.
(3) The adversary uses the starting and destination
locations of the vehicle's trip as a quasi-identifier for
the vehicle for further deanonymisation.
Preconditions: The vehicle must participate in VANET
communication and broadcast BSM or CAM to the surrounding
entities.
Leaf Nodes: I.2.1.2 pseudonym quasi-identifier.
Root Nodes: Identifiability.
```

Listing 3.12: Vehicle, I.2.1.2 pseudonym quasi-identifier threat.

```
deanonymisation with locations
Summary: The combination of the vehicle's location,
especially the home and office, is mostly unique from person
to person.
Primary Adversary: Similar to the I.2.1.2, entities that
receive short range wireless interfaces in a wide range.
Basic Flow:
(1) The adversary collects messages sent by the vehicle.
(2) The adversary locates the starting and destination
locations of the vehicle's trip to distinguish between
vehicles.
(3) The adversary compares the known locations with the
vehicle's starting and destination locations to determine
the driver's identity.
Preconditions: The vehicle must participate in VANET
communication and broadcast BSM or CAM to the surrounding
entities, and the vehicle's trajectory is reconstructed by
the adversary.
Leaf Nodes: I.2.3 distinguishable from others.
Root Nodes: Identifiability.
Remarks: Correctly linking multiple pseudonym certificates
of the same vehicle will increase the success of the attack
```

```
on the deanonymisation of the vehicle to the driver.
```

Listing 3.13: Vehicle, I.2.3 distinguishable from others threat.

The privacy threats in Listing 3.12 and Listing 3.13 show the adversary's final objective, where the user's identity is revealed. Note that the primary adversary in most of these threat cards can be any entity that can receive the short range wireless interface, i.e., participants in VANET communication. We also observed that most of the privacy threats in VANETs stem from how the standards are implemented. For example, the fine-grained, high frequency, unencrypted data for improving VANET safety-related application utilities, and the usage of path history, where the objective is unclear, but the presence of this information can benefit the adversary in the vehicle tracking process.

While threat elicitation is valuable for identifying potential privacy threats and tracing their sources, privacy threat cards do not provide a means to measure or quantify the extent of privacy loss. We observed that privacy threats in VANETs can be condensed and contribute to vehicle tracking problems. Therefore, measuring and quantifying the privacy loss could be conducted by evaluating the adversary's vehicle tracking model. In the next section, we will transform the VANET communication scenario into mathematical forms to build the adversary's vehicle tracking model.

3.2 Formalisation of VANETs Privacy Preservation

This section analyses the VANETs communication scenario and the privacy-preserving mechanisms based on the ETSI and SAE standards. This analysis aims to develop a mathematical model that facilitates the quantification of privacy leakage.

Building on the findings from the privacy threat cards presented in the previous section, we categorise the identified threats based on the entity re-

sponsible for triggering them. These threats are grouped into four stages of the system: pre-anonymisation data, privacy-preserving mechanisms, data dissemination, and the adversary model. Although these system stages occur sequentially from a single vehicle perspective, it can be seen as parallel from multiple vehicles' perspectives. In other words, one vehicle might be collecting raw data while another vehicle is disseminating the message. These stages also correspond to the seminal location privacy quantification framework [82] where the system can be represented with $\langle \mathcal{U}, \mathcal{A}, \mathcal{LPPM}, \mathcal{O}, \mathcal{ADV}, \mathcal{METRIC} \rangle$, which are user, actual trace, privacy-preserving mechanisms, observed trace, adversary model, and privacy metric.

Table 3.1: Mapping between system stage, privacy threat cards, and elements in the location privacy quantification framework.

System Stage	Threat Cards	Framework Elements
Pre-anonymisation data	U.1.1 Unawareness as data subject I.2.1.2 Pseudonym Quasi-Identifier I.2.3 Distinguishable from Others	$\mathcal{U}$ser, $\mathcal{A}$ctual trace
Privacy-preserving mechanisms	U.2.1 Preferences L.1.1 Unique Identifier I.2.1.1 Pseudonym Identifier	$\mathcal{LPPM}$
Data dissemination	Nr.1.2 Signed Data D.1 Observed Communications DD.1.2 Data Type Granularity DD.2.2 Frequency	$\mathcal{O}$bserved trace
Adversary model	L.2.1.1 Quasi-identifier combining data of a single individual L.2.2.3 Profiling an individual through (dis)similarity	$\mathcal{ADV}$ersary model

The system stage begins with pre-anonymisation data, raw data, from the

VANET user, which should only be known to the user, $\mathcal{U}$. Since the data has not been through the privacy-preserving mechanisms, it is equivalent to the actual trace, $\mathcal{A}$. This raw data is passed through a privacy-preserving mechanism, $\mathcal{LPPM}$, which may involve obfuscation, pseudonymisation, and pseudonym-changing mechanisms. The output from this step is the BSM/-CAM that vehicles use to communicate with each other. Note that BSMs/-CAMs are the messages that can be observed by an adversary, which is equivalent to the observed trace, $\mathcal{O}$. Then, the adversary, $\mathcal{ADV}$, uses the observed BSMs/CAMs to perform vehicle tracking with the vehicle tracking model of choice. Lastly, the output of the tracking model is the adversary's belief in the pre-anonymisation data, where we can measure the privacy leakage with privacy metrics, $\mathcal{METRIC}$.

For the rest of the thesis, we follow the symbolic structure described in Shokri et al. [82]. A bold italic capital letter, $\boldsymbol{X}$, denotes a random variable. A script capital letter, $\mathcal{X}$, denotes a set of the random variables taken from $\boldsymbol{X}$. A small italic, x, is an element drawn from the set $\mathcal{X}$.

3.2.1 Pre-Anonymisation Data

Pre-anonymisation data refers to the information generated by a vehicle throughout its trip. We assume this data is exclusively known to the user who produced it; therefore, it is equivalent to the actual trace, a, and related to the user who produced the data, u, from the location privacy quantification framework.

Let $\mathcal{U} = \{u_1, u_2, \ldots, u_N\}$ be a set of VANET users in which the total number of users is N. A user u_i generates a route r_i when he uses the vehicle in commutes, and the set of all routes used by a user is $\mathcal{R}u_i$. For instance, a user u_1 generates a route r_1 as he drives from his house to his workplace and r_2 as he drives back from his workplace to his house. Note that the same trajectory but at a different time will be considered as different routes in this work, since the time of driving is also considered as information presented in the route. For example, the user u_1 will generate another route, r_3, if he drives

from his house to his workplace the next day.

The start and destination of the route are pieces of information revealed by the route. These locations correspond to privacy threat card 3.12, I.2.1.2 pseudonym quasi-identifier, where house and workplace locations are used as the quasi-identifier of the trip. As highlighted by Golle and Partridge [103], knowing both house and office locations is highly beneficial for an adversary to de-anonymise a user. De-anonymised a user with home and workplace locations is also highlighted in the threat card 3.13, I.2.3, distinguishable from others. Additionally, the same start-destination pair can be generated by different trajectories, e.g., the user, u_1, may use different roads to drive from his house to his workplace in r_1 and r_3, but they contain the same start-destination pairs: u_1's house and workplace. By accounting for all the routes created by all of the users, we have a set $\mathcal{R}$, $\bigcup_{i=1}^{N} \mathcal{R}u_i$.

During the driving, the vehicle periodically gathers *internal data* before passing it to the privacy-preserving mechanism to generate a message m_i. The message is then broadcast to every entity in the vehicle vicinity, and hence, $\mathcal{M}r_i$ is a set of all messages generated during the route $r_i \in \mathcal{R}u_i$. The user might not be aware of this process as it happens automatically; therefore, it is related to the threat card 3.1, U.1.1 unawareness as a data subject.

According to ETSI TS 102 894–2 and SAE J2735 [69,70], the message contains several spatio-temporal and kinematics information about the current state of the vehicle. Information sent in a message includes, but is not limited to, location, speed, acceleration, heading, and yaw rate. Thus, we can represent a message with a tuple, $m = \langle lat, lon, s, a, h, y, t \rangle$, where the elements in the tuple represent latitude, longitude, speed, acceleration, heading, yaw rate, and time stamp accordingly. Another piece of information that will be available in a message is the vehicle's path history. The vehicle's path history is represented by an array of *path history points*, which contains location and time. Let $\mathcal{PH}_{mi}$ be an array of path history points in a message m_i, we have $\mathcal{PH}_{mi} = [ph_1, ph_2, \ldots, ph_n]$, where ph_i is a path history point. It must be

noted that the location and time in the path history point are taken from a previously broadcast BSM/CAM.

3.2.2 Privacy-Preserving Mechanisms

Before the message is broadcast, some of the raw data is processed to preserve the user's privacy. According to the standards [21, 25, 69, 70], the privacy-preserving mechanisms include obfuscation and identity anonymisation. The obfuscation is applied to the timestamp in the message and the time offset in the path history points. The identity anonymisation is achieved through periodically changed pseudonyms.

The time included in VANET messages is represented as either secMark in the SAE J2735 standard [69] or GenerationDeltaTime in the ETSI CAM standard [71], both of which express time in milliseconds. Despite the high resolution, its range, limited to values between 0 and 65,535, constrains its capacity to represent only one minute. As a result, temporal details such as the specific date, hour, and minute are excluded from the message, contributing to temporal obfuscation. However, an adversary who intercepts the message within a minute of its generation can accurately infer the omitted temporal information by correlating it with their local clock. Therefore, while the format may appear privacy-preserving, it remains vulnerable to inference attacks in near-real-time scenarios.

Another form of temporal obfuscation in VANET messages involves reducing the granularity of time units in the time offset associated with path history points. In BSM and CAM, the path history point's time offset is represented in ten milliseconds instead of one millisecond. Additionally, both time offsets have a maximum of 65,535, which is used to represent 655.35 seconds and higher. Similarly, the spatial resolution is limited by the maximum values allowed for latitude and longitude offsets, which are restricted to 13,107.1 microdegrees. These limitations mean that the path history points can be used accurately for 655.34 seconds and within approximately 1,310 metres of the

location in the message.

The main privacy-preserving mechanism in VANETs is through the usage of periodically changed pseudonyms instead of the user's real identity. Let a set $\mathcal{PID}$ be a set of all pseudonyms available in the VANETs. A subset $\mathcal{PID}u_i$ describes a set of all pseudonyms used by a user u_i. Since all pseudonyms must be unique, we have:

$$\forall u_i \in \mathcal{U};\ \bigcap_{i=1}^{N} \mathcal{PID}u_i = \emptyset \tag{3.1}$$

According to the standards, all messages must be signed with a pseudonym certificate. To focus on the pseudonym and the pseudonym-changing mechanism, we describe p_{mi} as a pseudonym that is used for signing a message m_i. The vehicle will constantly check for the trigger of pseudonym changing before signing the message and broadcasting it. We described a detail of the pseudonym-changing mechanism in Subsection 2.1.3. By opting out of the differences between standards, the abstract pseudonym-changing mechanism can be described as follows:

$$p_{mi} = \begin{cases} g(\mathcal{PID}u_i - p_{mi-1}) & \text{Met all conditions according to the standard} \\ p_{mi-1} & \text{otherwise} \end{cases} \tag{3.2}$$

where $g(\boldsymbol{PID})$ is a function which randomly pick a pseudonym from $\boldsymbol{PID}$, and p_{mi-1} is a pseudonym which used for signing a message m_{i-1}. We can define a set of messages signed by the same pseudonym as $\mathcal{M}pid_i$, which creates a segment s_i in a route r_i. Notice that if the pseudonym-changing conditions are not met throughout the route r_i, all messages in r_i will be signed by a single pseudonym, and the result is a segment s_i equivalent to a route r_i.

These privacy-preserving mechanisms are, however, can not be specified by the user; therefore, the mechanisms are related to a threat card 3.2, U.2.1 Preferences. Using pseudonyms instead of real identity is also related to Threat card 3.7, L.1.1 Unique Identifier, and Threat card 3.11, I.2.1.1 Pseudonym Identifier.

3.2.3 Data Dissemination

We define the data dissemination stage as the phase in VANET communication where BSM and CAM messages are transmitted between entities. These messages serve as heartbeat messages, continuously broadcast by vehicles at a frequency of 1–10 Hz [21, 71], which means the maximum duration between messages is one second. The high broadcast frequency, while essential for maintaining real-time situational awareness, also introduces privacy risks, as reflected in threat card 3.6 (DD.2.2, Frequency). Furthermore, the transmitted information and its units possess fine-grained granularity, enhancing the precision of the shared information but simultaneously increasing vulnerability to privacy breaches. This observation corresponds to threat card 3.5 (DD.1.2, Data Type). As discussed in Section 2.1, each message is digitally signed to establish trust and authenticity among communicating entities. However, signing every message also introduces privacy concerns, as signatures can potentially be exploited for entity linkage, which aligns with threat card 3.3 (Nr.1.2, Signed Data). Finally, since VANET communication is designed to be observable within a certain communication range to ensure safety and interoperability, this characteristic inherently creates a risk of unauthorised observation, linking to threat card 3.4 (D.1, Observed Communications). The observable communication also renders this stage to correlate to $\mathcal{O}$bserved trace in the location privacy-preserving framework, where all messages are observed by the adversary.

Since the messages can be lost during transmission, the number of messages in the observed trace might not be equal to the transmitted messages. We can describe the relation between the observed messages set, $\mathcal{M}r'_i$, and the transmitted messages set, $\mathcal{M}r_i$, as $\mathcal{M}r'_i \subseteq \mathcal{M}r_i$. Consequently, this relation is inherited by the set of messages that are signed with a pseudonym, $\mathcal{M}pid_i$, and creates $\mathcal{M}pid'_i \subseteq \mathcal{M}pid_i$, where $\mathcal{M}pid'_i$ is a set of observed messages signed with a pseudonym, pid_i. Additionally, from the set $\mathcal{M}pid'_i$, we have f_pid_i and $last\_pid_i$, which represent the first and last message in $\mathcal{M}pid'_i$. Lastly, with a

constraint that there is no message loss, we have

$$T(f\_pid_j) - T(last\_pid_i) \leq t_{max} \tag{3.3}$$

where $T(m)$ is an absolute time in message m, and t_{max} is a maximum time between messages, which is one second.

A summary of variable notations is presented in Table 3.2. Recall that a script capital letter, $\mathcal{X}$, denotes a set, and a small italic letter, x, denotes an element from the set $\mathcal{X}$. We also combine the notation such that a script capital letter followed by a small italic letter represents a set of $\mathcal{Y}$ that is associated with an element x.

Table 3.2: Notation of variables.

Variables	Annotation
$\mathcal{U}$	Set of all users in VANETs
$\mathcal{R}$	Set of all routes generated by all VANET users
$\mathcal{PID}u_i$	Set of all pseudonym owned by a user u_i
$\mathcal{M}r_i$	Set of all messages in a route r_i
$\mathcal{M}r'_i$	Set of all observable messages in a route r_i
$\mathcal{M}pid_i$	Set of all messages that signed with a pseudonym pid_i
$\mathcal{PH}m_i$	An array of path history points in a message m_i
t_{max}	A maximum time between messages, 1 second
f_pid_i	The first message that signed with a pseudonym pid_i
l_pid_i	The last message that signed with a pseudonym pid_i

3.2.4 Adversary Model

As highlighted in the preceding subsections, the adversary in the VANET environment may include any entity capable of receiving BSMs or CAMs, such as vehicles or roadside units. Additionally, the adversary can be an honest-but-curious traffic monitoring centre that performs its intended functions correctly

but is interested in analysing the data from the reconstructed vehicle trajectories. Even though the adversary might be able to obtain all the messages, the pseudonym-changing mechanism is implemented to protect the user's privacy. However, the pseudonym-changing mechanism is known to the adversary, and it is mostly deterministic, as highlighted in Equation (3.2) conditions, which are based on distance and time.

The adversary can exploit the deterministic conditions in Equation (3.2) to reconstruct vehicle trajectories. Exploitation can be achieved by estimating and correlating information, such as the location available from the messages. Additionally, location, speed, and direction can be used to predict a vehicle's next location and kinematics, which can be used to create a similarity between two pseudonyms. Notice that predicting and correlating location and kinematics between two pseudonyms is the threat card 3.10. Moreover, other information, such as path history points and static information, such as the vehicle's width and length, can also be used as alternative trajectory reconstruction approaches, threat card 3.9 and 3.8 accordingly. The adversary can also combine multiple trajectory reconstruction approaches to enhance the result of trajectory reconstruction.

Recall the conditions for the pseudonym-changing mechanism from Subsection 2.1.3; the conditions state that the vehicle should change its pseudonym after 800 metres and within 2–6 minutes in the ETSI standard. In SAE, the duration of pseudonym usage should not exceed 5 minutes unless the absolute distance from the last location where the vehicle changed its pseudonym exceeds 2.1 kilometres. These conditions can be used by the adversary to filter a potential pair of pseudonyms that might be from the same vehicle. From the last subsection, we have f_pid_i and f_pid_{i-1} that represents the first message of pseudonyms pid_i and pid_{i-1}. We also assume that the pid_i and pid_{i-1} are used by the same vehicle. With ETSI, we have

$$T(f\_pid_{i-1}) - T(f\_pid_i) \geq 120s \wedge T(f\_pid_{i-1}) - T(f\_pid_i) \leq 360s \tag{3.4}$$

$$D(f\_pid_{i-1}, f\_pid_i) \geq 800m \tag{3.5}$$

where $T(m)$ is an absolute time in message m, and $D(m_1, m_2)$ is the travelling distance between m_1 and m_2. Similarly, in SAE, we have

$$T(f\_pid_{i-1}) - T(f\_pid_i) \geq 300s \tag{3.6}$$

$$AbsD(f\_pid_{i-1}, f\_pid_i) \geq 2100m \tag{3.7}$$

where $AbsD(m_1, m_2)$ is the absolute distance between m_1 and m_2. The adversary can use Equation 3.4 and 3.5 to filter the possible pseudonyms pair under the ETSI standard, and Equation 3.6 and 3.7 for the SAE standard.

From the location and kinematics prediction method, the adversary can leverage the information in f_pid_i and l_pid_{i-1}, where l_pid_{i-1} represents the last message with pseudonym pid_{i-1}. Let $ExpLoc(m, t)$ be a function that calculates the expected location of a vehicle in the next t seconds based on location and kinematics in the message m. Then, the adversary can use Algorithm 3.1 to find the candidate pseudonym likely to be used by the same vehicle with pid_i.

Algorithm 3.1: Find a candidate pseudonym with location

```
1  expLoc ← ExpLoc(l_pid_{i-1}, t_max);
2  minDistance ← max(distance value);
3  candidatePID ← null;
4  forall f_pid_j appears in time T(l_pid_{i-1}) + t_max do
5      diffLoc ← locationDistance(expLoc, f_pid_j);
6      if (diffLoc ≤ locThreshold) ∧ (diffLoc ≤ minDistance) then
7          minDistance ← diffLoc;
8          candidatePID ← pid_j;
9      end
10 end
11 return candidatePID
```

Two concepts are left as abstractions in Algorithm 3.1, namely *expLoc* and *locationDistance*'s implementation and *locThreshold*'s value. Firstly, the implementation of *expLoc* and *locationDistance* is left to the adversary. The *expLoc* function can use as many variables as the adversary wants, as long

as they are available in the message or can be derived from it. The function may account for only location and speed or include the vehicle's heading, acceleration, and yaw rate in the calculation. Similarly, the $locationDistance$'s implementation can use a simple Euclidean distance function or a sophisticated Haversine formula [104]. While the Haversine formula can be more accurate than the Euclidean distance function, with a maximum difference of 1 second, the two functions might not appear any different. Secondly, the $locThreshold$'s value is also left for the adversary to decide. While it is ideal that the threshold should be small, for example, within one metre, the adversary may allow more room for errors in the calculation and sensors, and have a larger threshold.

In addition to the two concepts, the algorithm only considers the difference between the expected location, $expLoc$, and the location in the message f_pid_j. A more robust adversarial model should account for the full range of available kinematic data, which may improve the accuracy of trajectory reconstruction. The adversary can enhance the algorithm by leveraging differences in speed and heading, supported by corresponding differential values in the message, namely, acceleration and yaw rate. With the additional distances from speed and heading, the condition on Line 6 in Algorithm 3.1 can be split into three sets, namely location, speed, and heading distance. On the other hand, the three distances can also be combined into a single distance, i.e., the distance between the expected message and the actual message based on the differences of location and kinematics. The combined distance will be explored in Chapter 5 through two trajectory reconstruction models, including the Bayesian and Kalman Filter approaches.

Another trajectory reconstruction that can be performed by the adversary and is unique to VANET communication is a trajectory reconstruction with path history points. This approach corresponds to the threat card 3.9. The adversary can query messages that have the same location as described in the path history and infer that two messages originated from the same vehicle.

Algorithm 3.2 finds a candidate pseudonym likely to originate from the

same vehicle as message m_i. Two conditions are used as answers for data obfuscation described in Subsection 3.2.2. Firstly, the condition in Line 2 prevents the usage of path history points with a value that exceeds their limit. This is to prevent a miscalculation of the previous location of the vehicle. Secondly, the second half of the condition in Line 6 allows a range comparison since the time offset in the path history point is represented in ten milliseconds. In addition to the two conditions, a set operation in Line 5, $M - Mpid_i$, prevents a comparison of messages with the same pseudonym as the message m_i.

Algorithm 3.2: Find a candidate pseudonym with a path history point

```
1  candidatePID ← null;
2  if pathHistoryPoint_locOffset < max(locOffset) then
3  |  prevLoc ← Loc(m_i) − pathHistoryPoint_locOffset;
4  |  prevTime ← T(m_i) − pathHistoryPoint_timeOffset;
5  |  forall m_j in M − Mpid_i do
6  |  |  if (Loc(m_j) = prevLoc) ∧ (|T(m_j) − prevTime| ≤ 10) then
7  |  |  |  candidatePID ← pid(m_j);
8  |  |  end
9  |  end
10 end
11 return candidatePID
```

Lastly, the adversary can take advantage of static information, the vehicle's width and length, that is reused across multiple pseudonyms to make an inference on two pseudonyms. While this approach is related to the threat card 3.8, since many pseudonyms can originate from different vehicles with the same brand and model, this pseudonym linking method can result in a faulty correlation. Therefore, it will be more effective to use it as a filter along with Algorithm 3.1 and 3.2 instead of a standalone trajectory reconstruction model. In other words, the vehicle's width and length will be used as additional conditions that two pseudonyms must have the same width and length

in order to originate from the same vehicle.

In summary, this section outlined four stages of VANET communication: pre-anonymisation data, privacy-preserving mechanisms, data dissemination, and the adversary model. In the first three stages, we systematically defined the communication process using formal variables and mathematical relationships, providing a structured framework for modelling data flow. These formalised components were then utilised in the adversary model, exploring potential trajectory reconstruction strategies that an adversary might employ. These strategies will be revisited in Chapter 5, where we propose the trajectory reconstruction model for quantifying privacy leakage in VANETs. Additionally, the equations (3.6) and (3.7) will also be used during our investigation of public datasets in Section 4.1. By connecting the technical characteristics of message generation, anonymisation, and dissemination with the adversary's capabilities, we highlighted how anonymised data can still be exploited to compromise user privacy.

3.3 Analysis of Frameworks in Background Literature

This section explores how existing privacy quantification frameworks align with the formalised VANET communication introduced in the previous section. The primary objective is to assess the compatibility and feasibility of applying these seminal frameworks within the specific context of VANET communication systems. To guide this evaluation, we consider four key aspects: (1) the privacy-preserving mechanisms employed, (2-3) the trajectory reconstruction models and their required inputs, and (4) the data inputs necessary for quantifying privacy loss. These criteria provide a structured basis for comparing the suitability of each framework. We focus on three seminal privacy quantification frameworks from the literature: the hub and spokes model [55, 84], the location privacy quantification framework [82], and the VANET-specific

privacy evaluation framework [8]. Through this analysis, we aim to identify which elements of these models can be adapted to the VANET context and highlight any gaps that must be addressed for effective privacy evaluation.

3.3.1 The Hub and Spokes Model

The hub and spokes model was among the earliest attempts to quantify privacy in VANETs [55, 84]. This framework assesses privacy loss by evaluating the likelihood of associating a user's identity with the origin and destination of their trip. Such an approach aligns with two specific privacy threats identified in the threat cards: the use of location as a quasi-identifier (threat card 3.12) and deanonymisation through location inference (threat card 3.13). However, being an early contribution, the hub and spokes model does not account for privacy-preserving mechanisms defined in VANET standards, such as pseudonym-changing schemes. Additionally, the model assumes that the adversary already possesses a known probability linking users to their trip origins and destinations. As a result, it bypasses the need for trajectory reconstruction as a prerequisite for privacy quantification.

While the hub and spokes model can theoretically be adapted for use in modern VANET contexts by integrating it with the formalised adversary model and reconstructing trajectories to derive identity-location probabilities, this adaptation reveals a fundamental limitation. The model does not account for the presence or effectiveness of privacy-preserving mechanisms implemented in VANET standards, such as pseudonym-changing schemes. Because it does not incorporate these mechanisms into its analysis, the model cannot evaluate whether such techniques succeed or fail in protecting user privacy. As a result, any privacy quantification based on the hub and spokes model may deliver a misleading interpretation of privacy levels in VANETs. Specifically, it risks overestimating adversarial success because it does not simulate how pseudonym changes or other protective strategies affect trajectory reconstruction. Therefore, the hub and spokes model is ineffective for accurately assess-

ing privacy in current VANET systems. Its inability to reflect the dynamic privacy-preserving practices defined by current standards limits its utility in real-world evaluations.

3.3.2 Location Privacy Quantification Framework

Although the framework by Shokri et al. [82] has provided valuable direction for formalising aspects of VANET communication in the previous section, the framework shows several incompatibilities when applied to VANET-specific privacy quantification. These limitations arise across three key areas: the privacy-preserving mechanisms, the adversary model, and the required inputs for adversarial inference. Firstly, the framework is proposed for general location privacy and uses location obfuscation in the privacy-preserving mechanism. This approach is fundamentally incompatible with VANETs, where location accuracy, specifically precise longitude, latitude, and altitude, is essential for safety-critical applications. Introducing obfuscation in location data could compromise VANETs' core function, ensuring timely and accurate communication for collision avoidance and traffic efficiency.

On the other hand, identity anonymisation in this framework aligns with the concept of VANETs, where the pseudonym is randomly chosen from the available pool. However, the trigger conditions for the identity-changing mechanism in VANETs are deterministic. In practice, VANETs often follow standardised triggering conditions for pseudonym changes—such as time, distance, or location-based criteria—which introduce inherent linkability between consecutive pseudonyms used by the same vehicle. This structured behaviour is described in Equations (3.4) – (3.7), and is absent in the generalised identity model of the original framework. Although using random functions in location obfuscation and identity anonymisation is innocuous, the effects of randomness in the privacy-preserving mechanism can be observed in the adversary model.

Next, using randomised mechanisms in location obfuscation and identity anonymisation within the framework undermines its ability to model realistic

adversaries in VANET environments. In VANET, the adversaries can exploit predictable patterns in pseudonym-changing mechanisms and broadcast kinematics as described by threat card 3.9 linking two pseudonyms with path history points and threat card 3.10 linking two pseudonyms with calculated location and kinematics. By relying on randomness rather than the actual structured design of VANET protocols, the adversary model in the framework lacks the resolution needed to simulate realistic privacy attacks and requires a training set to reverse the probabilistic functions effectively. Therefore, the framework's adversary model is ineffective for evaluating trajectory reconstruction under the constraints and behaviours mandated by VANET standards.

Despite the previously identified incompatibilities, the privacy quantification concepts and metrics introduced in the framework remain valuable for assessing privacy loss in VANETs. Specifically, the framework centres on adversary incorrectness, an approach that quantifies privacy by measuring how often the adversary fails to infer sensitive information. This yields three core privacy metrics: incorrectness of localisation, incorrectness of aggregated presence, and incorrectness of meeting disclosure. These metrics rely on comparing the output of a tracking or inference model against ground-truth data to measure privacy risk. While this approach does not directly incorporate the specific privacy-preserving mechanisms in VANET standards, and adopting these privacy metrics into our work will not be advantageous in accounting for these mechanisms, the underlying concept of adversary incorrectness offers a robust and generalisable foundation. Incorporating this concept into VANET privacy analysis can help highlight the limitations of an adversary's tracking capability.

3.3.3 VANETs Privacy Quantification

Although the privacy quantification proposed by Emara et al. [8] is directly related to VANETS, multiple aspects in their work can be changed to align with our formalised VANETs communication. Firstly, the privacy-preserving

mechanism in their work assumed that the vehicles change their pseudonym in every message. This aspect of the privacy-preserving mechanism is not aligned with the standards, which require certain conditions to trigger the vehicle's pseudonym to change. Two effects on the adversary model can be presumed from their mechanism, including the conditions in Equations (3.4) – (3.7) can not be leveraged, and that more computation must be made by the adversary. Since every message uses a different pseudonym, it is obvious that the relationship between two pseudonyms described in Equations (3.4) – (3.7) will not hold. Changing the pseudonym in every message also prevents the formation of a set of messages with the same pseudonym; therefore, the adversary has to spend more computation to group these messages. This also means that the threat card 3.7, the vehicle's pseudonym as a unique identifier, is ignored in the work.

Secondly, although Emara et al.'s trajectory reconstruction model aligns conceptually with Algorithm 3.1, as described in our formalised adversary model, there is clear potential for improvement by integrating a broader set of kinematic attributes available in VANET messages. Emara et al.'s model primarily focuses on location information, which provides a foundation for tracking. However, standard VANET messages, BSM and CAM, contain additional kinematic data such as speed, heading, acceleration, and yaw rate, which offer temporal and directional context. Incorporating these variables can improve the accuracy and robustness of the adversary's trajectory reconstruction capabilities. For example, sudden changes in the vehicle's speed and heading can be detected with acceleration and yaw rate. Moreover, static vehicle characteristics, such as width and length, are included in the message and can serve as filters to eliminate pseudonym candidates that do not match the observed vehicle's physical dimensions. The added filtering layer helps narrow the candidate set of pseudonyms during tracking and makes the adversary model more precise.

Similar to the previous model, Emara et al.'s use of tracking percentage

as a measure of privacy loss provides a useful and intuitive assessment of adversarial effectiveness. Unlike earlier frameworks that primarily focus on the adversary's incorrectness, particularly the incorrectness of linking pseudonyms in this work, tracking percentage directly quantifies the degree to which an adversary can successfully reconstruct an entire vehicle's trajectory. The tracking percentage offers a complementary perspective by highlighting the accuracy and completeness of trajectory reconstruction rather than the adversary's capability. This insight is particularly valuable for an adversary, as a high tracking percentage indicates the ability to correlate different message fragments and pseudonyms to a single user over time. This can lead to successfully deidentifying the user's identity, which aligns with threat card 3.13, deanonymisation with locations.

3.4 Summary

This chapter presented an analysis of Vehicular Ad Hoc Network (VANET) communication from two perspectives: privacy threat elicitation using the LINDDUN framework and formalisation of VANET communication into a structured model. Through applying the LINDDUN privacy threat elicitation framework, we identified 13 privacy threats specific to VANETs. These threats were then mapped across four stages of VANET communication that structure our formalisation: pre-anonymised data, privacy-preserving mechanisms, data dissemination, and adversary model. Building on a seminal location privacy quantification framework, we developed a mathematical formalisation of VANET communication, establishing relationships between a user's identity and the broadcast data. This formalisation was designed to be compatible with ETSI and SAE standards and supports privacy quantification through clear and structured variables. We further proposed example adversary algorithms capable of reconstructing vehicle trajectories using available information from VANET messages. These algorithms illustrate how real-world adver-

saries could exploit standard-compliant data flows to undermine user privacy. Finally, we evaluated the compatibility between our four-stage communication model and existing privacy quantification frameworks in the literature. This evaluation revealed several limitations in current models, particularly regarding their assumptions about pseudonym changes, use of randomness, and adversary capabilities. We identified areas where these models could be improved to better reflect realistic VANET environments and enhance the adversary's accuracy and robustness.

Chapter 4

VANETs Dataset And Simulated Data

Acquiring and evaluating datasets is a critical step in the quantitative analysis of privacy loss, as the quality and structure of the data directly influence the validity of privacy evaluations and adversary modelling. We use the following three key characteristics that the datasets should have for privacy quantification research to analyse the datasets.

1. Standards Compliance: The dataset must follow VANET communication standards (e.g., ETSI and SAE), specifically regarding message structure, broadcast frequency, and pseudonym usage. This ensures the data accurately reflects what is transmitted in real VANET systems and supports compatibility with the formalised models developed in earlier chapters.

2. Realistic Vehicle Density: The number of vehicles in the dataset must be sufficient to mimic realistic traffic conditions. Vehicle density is essential for evaluating the adversary's capability to reconstruct vehicle trajectories and quantify privacy in VANETs.

3. Adequate Spatial Coverage: The dataset should cover a large geographical area to capture diverse driving behaviours and urban layouts. A broad coverage area also enables analysis of long-range tracking attempts.

These characteristics are selected to reflect the gaps in the dataset characteristics presented in Table 2.4. As this thesis focuses on privacy leakage quantification in VANETs, it is crucial that the datasets follow VANET communication standards to properly assess the system. According to Table 2.4, the real-world datasets [88–91, 93, 94] used by the literature did not capture our first criterion. On the other hand, the synthetic datasets used by the literature, which followed VANET communication standards, showed a lack of realistic vehicle density [97] and adequate spatial coverage [8, 96].

This chapter begins with an analysis of two real-world VANET datasets [105, 106] that were not previously used in VANET privacy leakage quantification literature in Section 4.1. Then, we extract characteristics of these real-world datasets to compare them with our additional synthetic dataset, ensuring that the message's characteristics in the synthetic dataset match those of the real-world datasets. These characteristics include the number of pseudo-identities per minute and the characteristics of the path history in a message. As highlighted earlier, the synthetic datasets used in the literature did not provide realistic vehicle density and spatial coverage; we present the process for developing a synthetic dataset, which will serve as the primary dataset in this thesis, in Section 4.2. We also include comparisons between the synthetic dataset and the real-world dataset based on the characteristics we observed in Section 4.1.

4.1 Public Datasets

This section analyses the public datasets published by the US Department of Transportation (USDOT) [86]. While there are three available datasets, one of the three datasets, the New York City dataset [107], is not included in this thesis since the Basic Safety Message (BSM) is not included in their published dataset. The other two datasets used in this section are from the Wyoming Department of Transportation [19] and the Tampa Hillsborough Expressway Authority (THEA) [18]. Apart from the full collection of the dataset at [86],

example datasets can be accessed at [105, 106].

We first identify the general characteristics of these datasets. Table 4.1 summarises the general characteristics of these datasets. Note that the word pseudo-identity is used instead of pseudonym here to highlight the field name in a BSM. While the pseudo-identity is a field name [69] and a pseudonym is a certificate for message signing [21], this information is equivalent since the pseudo-identity remains the same and changes only when the pseudonym is changed [21]. To identify the last three characteristics in Table 4.1, we first group the BSMs according to their pseudo-identity. For each pseudo-identity, we count the number of BSMs in the group and calculate the travelling distance and time duration.

Table 4.1: General characteristics of Wyoming and Tampa Basic Safety Messages datasets.

Characteristics	Wyoming dataset	Tampa dataset
Duration of dataset	1,332 days	545 days
Coverage area	$90.84 \times 25.98\ km^2$	$1.65 \times 1.47\ km^2$
Average pseudo-identities per day	80	6,660
Average BSMs per day	142,702	885,340
Average BSMs in a pseudo-identity	3,487	127
Average distance in a pseudo-identity	54.81 km	0.36 km
Average duration in a pseudo-identity	2.5 hours	89 seconds

We can observe that the Wyoming dataset contains a larger coverage area than the Tampa dataset. This is because the Wyoming dataset was collected in a highway area [19], whereas the Tampa dataset was collected in a city area [18]. It should also be highlighted that the average distance used by a single pseudo-identity in Wyoming is longer than the condition that trig-

gers the pseudonym-changing mechanism based on the SAE's standard [21], highlighted in subsection 2.1.3. This observation leads to our analysis of the inadequacies in real-world datasets in the following subsection.

4.1.1 Limitations of Public Datasets

As discussed in Section 3.1 regarding the VANET's privacy threats, VANET users are susceptible to being tracked by adversaries. We further discussed two algorithms for vehicle tracking in Algorithms 3.1 and 3.2, where the adversary leverages the information in the message for tracking the vehicles. Having access to public datasets can greatly support privacy quantification by simulating privacy attacks through trajectory reconstruction, especially when these datasets align with the three key perspectives outlined at the beginning of this chapter. However, the public datasets exhibit three limitations that undermine their effectiveness for privacy research. The first limitation appears in the Wyoming dataset, where the vehicles may not follow the pseudonym-changing mechanism based on the SAE standards. The second limitation is that duplicated data points can be observed in the Tampa dataset, each with a different pseudo-identity. Lastly, the third limitation involves the Tampa dataset, where privacy concerns from the data publisher require some information to be removed, scrubbed, and sanitised before publication. The following subsection provides our analysis that reveals these limitations and explains how these aspects can be limitations in VANET privacy research.

Firstly, it appears that the vehicles in the Wyoming dataset may not follow the SAE's pseudonym-changing mechanism standards. We follow up on the average travel distance characteristic in the Wyoming dataset by identifying the number of pseudo-identities that violate the pseudonym-changing mechanism. As described in subsection 2.1.3, the SAE J2495/1 describes that the pseudonym must be changed after it has been used for 5 minutes, and the absolute distance from the last pseudonym changed is longer than 2.1 kilometres [21]. Since the distance presented in Table 4.1 is the travelling distance,

we recalculate the absolute distance between the locations in the first and last BSMs with the same pseudo-identity.

Figure 4.1 displays the percentage of pseudo-identities from the Wyoming dataset regarding the total number of pseudo-identities in the year displayed on the x-axis, with line colours representing the conditions for the pseudonym-changing mechanism. The blue line represents the pseudo-identities that were used for more than five minutes. The black line represents the pseudo-identities with an absolute distance between their first and last location longer than 2.1 kilometres. Lastly, the percentage of pseudo-identities that meet both of these conditions is shown in a red line. We observed that 16.20% of the overall pseudo-identities during the data collection period are used for more than 5 minutes and with an absolute distance of more than 2.1 kilometres.

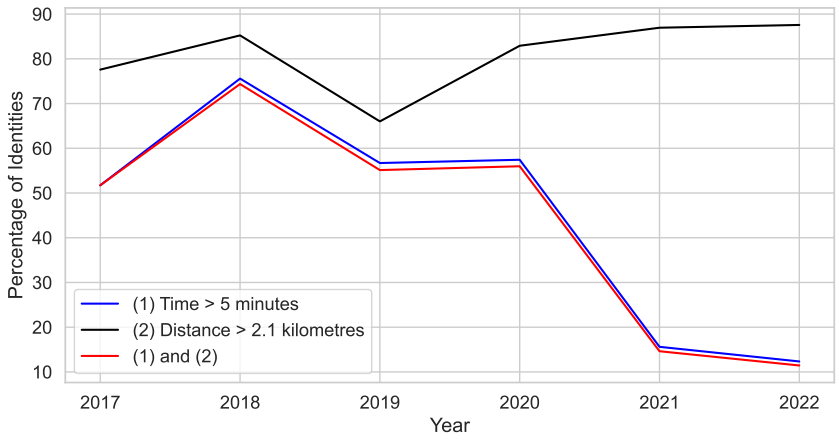


Figure 4.1: Percentage of identities with: duration more than 5 minutes (blue), absolute distance more than 2.1 kilometres (black), or both (red).

Since the Wyoming dataset represents a highway scenario, most identities will have their absolute distance over 2.1 kilometres, and the pseudonym-changing conditions tend to be influenced by the usage duration more than the absolute distance. We can observe that the percentage of identities that met both conditions decreases over the course of data collection and drops sharply between 2020 and 2021, when the current version of the SAE J2495/1 standard was published. Nonetheless, this number is still higher than 10%, equal to

4,668 identities in 2022. Having both conditions at the same time **violates the pseudonym-changing conditions outlined in the SAE J2945/1 standard**, which mandates identity changes based on time and distance thresholds to preserve user privacy. The absence of a compliant pseudonym-changing mechanism has two critical implications. First, it indicates that the dataset does not follow the VANET standards, which are essential for evaluating privacy risks according to the VANET communication standards [21]. Second, without periodic pseudonym changes, trajectory reconstruction models that rely on this mechanism to simulate adversarial behaviour cannot be applied and assessed. Consequently, this limitation prevents meaningful privacy quantification based on standard-compliant VANET mechanisms, undermining the dataset's utility for privacy research.

The second limitation that compromises the effectiveness of the dataset is **the presence of duplicated data points between different vehicle identities in the Tampa dataset**. We observed that approximately 7.99% of BSMs have exact duplicates regarding location and kinematics, but are associated with different pseudo-identities. On first thought, it might appear that there could be multiple cars at the same time and place; however, that is not the case here, as the granularity of the location unit is around one centimetre. Such a small difference will not be able to hold two vehicles at the same time. Additionally, other kinematics, such as the vehicle's speed, heading, and timestamp, also have the same values in duplicate messages. Therefore, this anomaly may result from faulty sensors or data processing errors, posing a significant challenge for adversarial trajectory reconstruction models. Such duplication could affect the tracking algorithms and reduce the validity of the effort for privacy quantification through trajectory tracking, as it creates confusion when pseudonyms must be linked, i.e., when two candidate pseudonyms have the same locations and kinematics. While a practical workaround is to filter out these duplicated BSMs during preprocessing, when assessed at the identity level, these duplications affect an average of 39.15% of

pseudo-identities daily. This large amount of duplication impacts the reliability of the dataset, as nearly half of the vehicle identities may be distorted or unusable for accurate privacy evaluation.

Lastly, the Tampa dataset presents a limitation due to **data sanitisation implemented to protect participants' privacy in the connected vehicle pilot project**. As outlined in their data privacy plan (FHWA-JPO-17-461) [74], the dataset was intentionally stripped of Personally Identifiable Information (PII) and Sensitive PII. To achieve this, vehicle identifiers were replaced with random IDs presented in each message. While anonymisation is essential for protecting participants' privacy, it introduces a significant drawback for privacy research: removing ground truth from data. The absence of ground truth makes it impossible to validate trajectory reconstruction results. Even if an adversary model successfully reconstructs vehicle trajectories, there is no authoritative reference to assess the accuracy of those reconstructions. This reduces the dataset's utility for evaluating the effectiveness of privacy-preserving mechanisms and adversarial capabilities within VANETs.

To summarise, both of the public datasets contain limitations that render them ineffective for use with our VANETs privacy leakage quantification method. In the Wyoming dataset, our analysis shows that the vehicles that generated the BSMs did not apply the pseudonym-changing mechanism according to the SAE standards. The standard non-compliance behaviour makes the dataset unusable to measure the effectiveness of the standard pseudonym-changing mechanism. For the Tampa dataset, not only did it contain duplicated data points, but some of the data points had been sanitised and removed before they were published. Such limitations can render our trajectory reconstruction ineffective. Additionally, the ground truth of both datasets is not available, which makes it impossible to cross-check the correctness of the results from the trajectory reconstruction model. As a result, we generate our own synthetic dataset to evaluate the privacy leakage in VANETs when the standards are properly implemented. The process for data generation will be

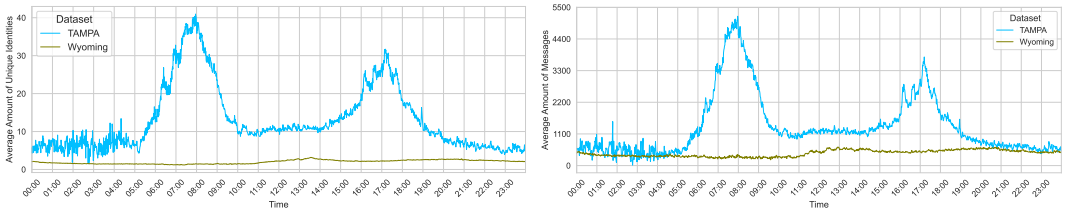
discussed in Section 4.2.

4.1.2 Characteristics of VANETs in Public Datasets

While the simulation tools we used, as will be presented in Section 4.2, can simulate substantially appropriate mobility data for privacy leakage quantification, they cannot fully capture the essence of VANET communication, leaving some information omitted from the simulation results. Therefore, understanding the message characteristics from real-world datasets can be used for improving the realism of the synthetic dataset, thereby bringing privacy leakage quantification with the synthetic dataset closer to real-world scenarios. Despite the limitations in these datasets, we can still make use of the datasets by analysing their characteristics and using them for a comparison with a synthetic dataset. The characteristics we are interested in, which should not be affected by the inadequacy in the dataset discussed in the previous subsection, are the number of pseudo-identities throughout the day and the characteristics of the path history. The number of pseudo-identities throughout a day will help us understand the traffic behaviour between the city-based scenario (Tampa) and the highway scenario (Wyoming). On the other hand, the characteristics of path history could provide insight into this information, as path history poses a unique privacy threat to VANET communication, as discussed in Threat Card 3.9.

Figure 4.2 shows the amount of (a) average number of unique pseudo-identities and (b) average number of Basic Safety Messages (BSMs) per minute from both datasets. We can observe that the Tampa dataset has more unique identities and BSMs than the Wyoming dataset. The Tampa dataset also exhibits the typical rush hour peak behaviour, whereas the Wyoming dataset remains mostly flat throughout the day.

The peak hour pattern in the Tampa dataset reflects the characteristics of an urban environment, where vehicle activity is closely tied to daily commuting routines, particularly during morning and evening rush hours. In contrast,



(a) Unique identites

(b) Total BSMs

Figure 4.2: Average number of unique identities and Basic Safety Messages (BSMs) per minute.

the Wyoming dataset, which represents a highway scenario, does not indicate this peak-hour trend. Additionally, Figures 4.2 (a) and (b) show a strong correlation between the number of unique pseudo-identities and the number of BSMs, as highlighted by their similar patterns. This correlation is expected, as according to the standard [21], each vehicle broadcasts BSMs at a rate of 1 to 10 Hz. Therefore, the number of BSMs should be approximately 1 to 10 times larger than the number of pseudo-identities. Since Figure 4.2 (a) and (b) show the number of pseudo-identities and BSMs per minute, the expected number of BSMs becomes 60 to 600 times larger than the number of pseudo-identities. It is worth noting that, according to the graphs, the number of BSMs is only approximately 120 times larger than the number of pseudo-identities, indicating that the BSM broadcast rate is approximately 2 Hz in the real world.

The next characteristic that we can analyse is the number of path history points in a message. Figure 4.3 illustrates the percentage of BSMs based on the number of path history points included in each message across both datasets. There is a notable difference between the datasets regarding how path history is represented, with the Wyoming dataset generally having few path history points. In the Wyoming dataset, nearly 50% of the BSMs include only two path history points, and most messages contain fewer than five path history points. Conversely, the Tampa dataset exhibits a wider distribution of path

history points. The number of points in BSMs ranges from 2 to 14, with over 30% of the messages containing 15 path history points. Note that the peak at 15 path history points is a result of the limitation of the number of path history points in a message [21].

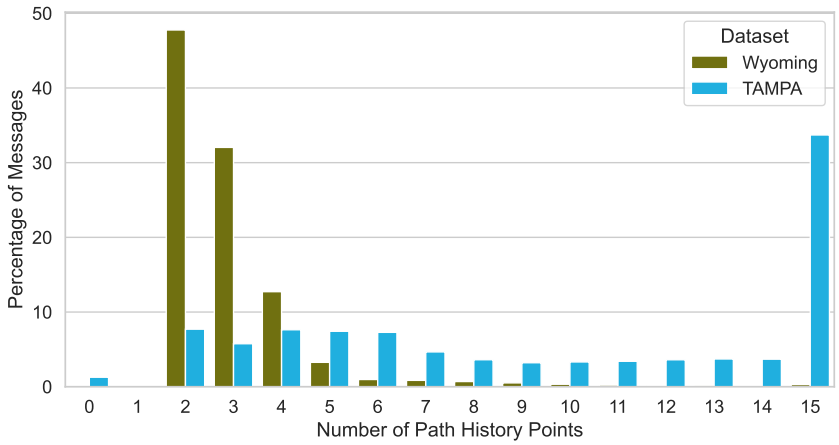


Figure 4.3: Number of path history points in a message.

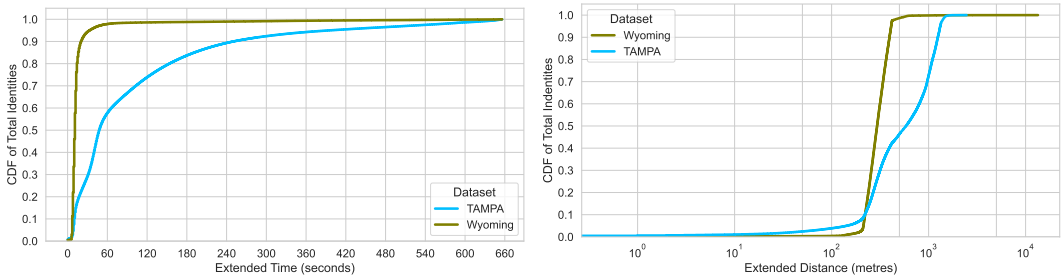
The difference in the distribution of path history points across the Wyoming and Tampa datasets may reflect the impact of road scenarios on the number of path history points. According to the SAE-J2945/1 standard, the vehicle should use a minimum number of path history points to represent a minimum of 200 metres in the past [21]. Vehicles typically travel at higher speeds in the Wyoming dataset, which primarily reflects highway driving conditions. As a result, they can cover 200 meters in a relatively short period, requiring fewer path history points to meet the minimum standard. This explains the concentration of BSMs with only 2 to 4 points in this dataset. On the other hand, the Tampa dataset represents urban driving environments, which may involve lower speeds and frequent stops; therefore, covering a minimum distance of 200 meters may take a longer duration than the Wyoming scenario, resulting in a larger number of path history points. As a result, the observed differences in path history point distributions are likely influenced by the contextual driving environments represented in each dataset.

The last characteristic we are interested in from the datasets is the distance

and time in the past from the first BSM sent by the pseudo-identity, as revealed through path history points. In this analysis, we first group the BSMs with their pseudo-identity, representing sets of BSMs broadcast by the vehicle. Then, we retrieve the locations and times in the past of the pseudo-identity from the location and time in the first BSM used by the pseudo-identity and the location and time offsets in the path history points. With the retrieved location and time, we can reconstruct the vehicle's past trajectory and calculate the absolute distance and duration of the recovered trajectory. We name this information as *"extended absolute distance"* and *"extended time"* from the first BSM broadcast by a pseudo-identity.

Figure 4.4 shows the cumulative distribution of all pseudo-identities with (a) the extended time and (b) the extended absolute distance from the first BSM broadcast by a pseudo-identity. We can observe that almost every pseudo-identity in Wyoming has been exposed for less than one minute in the past, whereas approximately 40% of the pseudo-identities in Tampa have been exposed for more than one minute. For the extended distance, almost every pseudo-identity in Wyoming is exposed with a path history of less than 500 metres; however, more than 50% of pseudo-identities in Tampa have an extended distance of more than 500 metres. Nevertheless, path history can be exposed for up to almost 11 minutes in both datasets, representing the maximum limit of time offsets in the path history point [69].

This insight correlates with Figure 4.3, where the number of path history points in a BSM in Wyoming is much less than those in Tampa; therefore, we can observe sharp inclines of the percentage of pseudo-identities at a low extended time and distance in the Wyoming dataset. However, the graph is more gradually inclined over an extended time in the Tampa dataset, suggesting the variation in commute time in the urban area. We could also observe two sharp inclines in the Tampa dataset in the extended distance graph. This may suggest two driving behaviours: driving in the city area (the sharp incline between 200 and 500 metres) and driving between the city area and the out-



(a) Extended time in the past

(b) Extended distance in the past

Figure 4.4: Comparison of the cumulative distribution of identities with the information contained in path history.

skirts of the city (the sharp incline around 1 kilometre). Nevertheless, it should be a concern that the path history can expose the user to up to 11 minutes in both datasets, with 2 kilometres in an urban scenario and 13 kilometres in a highway scenario.

This section provides our analysis of the inadequacy of VANET public datasets for privacy quantification based on trajectory reconstruction evaluation. At the beginning of the chapter, we raised three important aspects that the dataset should have for the privacy evaluation using the trajectory reconstruction approach. These key perspectives include standard compliance, realistic vehicle density, and adequate spatial coverage. Neither the Wyoming nor the Tampa datasets have met all of these perspectives, rendering them inadequate for use in our evaluation. Additionally, due to concerns about the privacy of participants in the pilot projects, neither of these datasets provides the ground truth of vehicle trajectories, which is essential information for evaluating the correctness of the trajectory reconstruction model. Despite their inadequacies, we are able to analyse three characteristics that can be used for comparisons with the upcoming synthetic dataset. These characteristics include the number of pseudo-identities throughout the day, the number of path history points in a message, and the extended time and distance from the first message broadcast by a pseudo-identity exposed by path history.

4.2 Synthetic Dataset

In the previous section, we explored the characteristics of real-world VANET datasets and identified several limitations that render them unsuitable for privacy research in VANETs using trajectory reconstruction approaches. In this section, we turn to an alternative approach: using synthetic datasets for quantifying VANET privacy. Prior studies have widely adopted the use of synthetic datasets, particularly when real-world data is unavailable, or a more controlled dataset is required [8, 49, 50]. Simulation offers a controlled environment that allows us to model standard-compliant behaviour and evaluate privacy-preserving mechanisms under configurable conditions.

It must be noted that the synthetic dataset can be purely simulated or based on a real-world dataset. In the first scenario, the simulation is usually presented with a Manhattan-style city grid [50]. The vehicles in such a simulation arrive and leave the area in a controlled manner. On the other hand, the simulation can be based on a real-world dataset [8, 49]. This type of simulation uses the vehicle locations from the real-world dataset and allows the vehicles to freely drive from point A to point B. The arrival of vehicles into the simulation can be set to reflect the time at which the vehicle was presented in the real-world dataset. The benefit of the second type of simulation is that it transforms a real-world dataset into a synthetic one for a more controllable scenario. This can also be used to simulate and capture VANET communication while maintaining the vehicles' mobility to the real-world characteristics.

4.2.1 Simulation Setup

VANET simulation tools can generally be categorised into two main types: mobility simulation and network simulation. Mobility simulators, such as the Simulation of Urban Mobility (SUMO) models the physical movement of the vehicles, while network simulators, such as OMNeT++ [108], simulate the communication between vehicles and infrastructure. These two simulation

types are typically integrated into a combined framework that enables real-time communication between the mobility and network components [109,110]. Such integration ensures synchronisation of vehicle behaviour and message exchange, providing a more realistic and holistic simulation environment. In this work, we employed the Artery framework [110], which connects SUMO for mobility simulation with OMNeT++ for network simulation, with the TAPAS Cologne mobility dataset [95,111]. Additionally, we developed a custom C++ module to simulate pseudonym-changing mechanisms, allowing us to evaluate privacy-preserving strategies under standard compliance.

Two commonly used frameworks for VANET simulation are Veins and Artery [109, 110]. While Veins is more widely adopted in the research community, this thesis employs the Artery framework due to its built-in support for the ETSI protocol stack, which enables the simulation and transmission of Cooperative Awareness Messages (CAMs). Artery's open source also offers additional flexibility in modifying the underlying code. Since CAMS and BSMS share similar data structures, we can modify the code in our local environment to simulate specific information about Basic Safety Messages (BSMs). We also modified the code to log all messages that the vehicles sent and received.

With vehicle density and spatial coverage as the requirements for the dataset, the Artery framework, using SUMO's *scenario feature*, allows us to define simulation parameters, including the number of vehicles, their routes, and the road network map. Being able to control such simulation parameters makes the Artery framework suitable for configurable VANET simulations. For our simulation scenario, we employed the Traffic and Parking Simulation Cologne (TAPASCologne) mobility dataset [95,111]. This dataset provides 24 hours of mobility data based on a detailed and realistic map of Cologne, Germany, covering an area of approximately $28.94 \times 31.27 km^2$. According to the scenario documentation on the SUMO website [95], the dataset supports up to 1.2 million vehicle trips, assuming each vehicle performs three daily trips. However, the site suggests that only one-third of the total trips should be used

in the simulation to avoid a wide range of traffic jams in the scenario. This constraint helps simulate a manageable number of traffic flows while maintaining the usability of the synthetic dataset for evaluating VANET privacy and communication performance. Figure 4.5 shows the vehicles' locations, which resemble the road network of the city of Cologne.

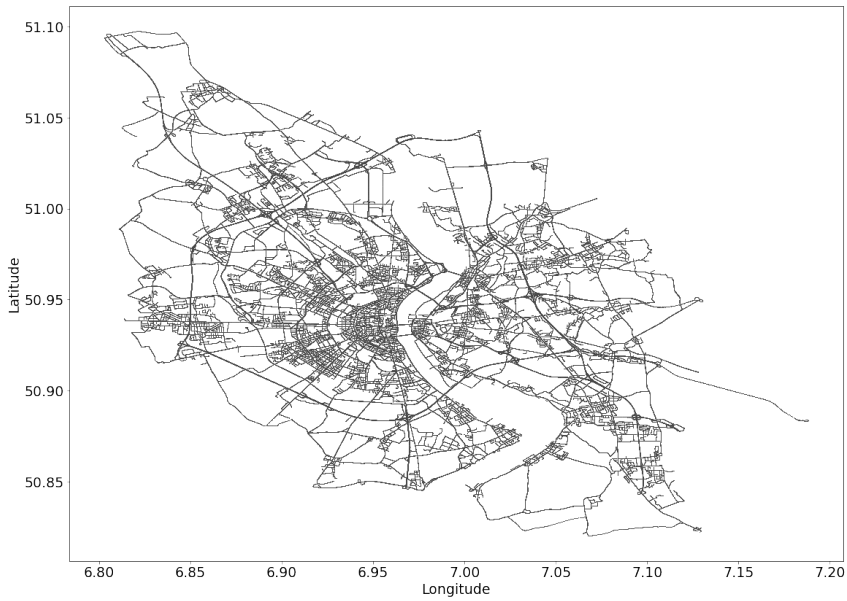


Figure 4.5: Plot of vehicles' locations, resembling Cologne's road network.

Having a large number of vehicles is advantageous because it reflects a realistic number of vehicles on the road and may highlight the difficulty adversaries face during the trajectory reconstruction process. However, it can also result in high resource consumption during the simulation process. It is worth noting that the simulation must be performed sequentially to ensure that the vehicle's locations do not collide with those of others, and that the individual vehicle states are maintained in memory. Moreover, each message broadcast by the vehicles must be computed to determine its reachability to all other vehicles. As the number of vehicles increases, the time required to simulate one second in the synthetic dataset also increases since every vehicle's movements and message delivery are computed sequentially. The larger number of simulated vehicles also increases memory requirements to store individual vehicle states. Additionally, our attempt to log every message sent or received

introduces further computational overhead, slowing the simulation considerably. Consequently, the simulation framework reached its memory limit under the large number of simulated vehicles, even when using only one-third of the traffic flows recommended by the SUMO website [95]. Under such limitations in the Artery simulation framework, we are able to simulate two chunks of the scenario: one from midnight to 6 a.m. and another from 5:30 a.m. to 6:30 a.m.

4.2.2 Simulation Process

To address the performance issues encountered with the Artery framework, we adopted **an alternative simulation approach using only SUMO** to simulate vehicle mobility. In this setup, we recorded each vehicle's location, speed, and heading at one-second intervals (maximum interval between messages on both ETSI and SAE standards [21, 72]). Similar to the Artery framework, SUMO supports scenario-based simulation, allowing us to define the road network, vehicle routes, and traffic conditions through input scenario files. Listing 4.1 shows examples of how trips can be specified for SUMO, where each line represents a trip. Notice that only the vehicle's identification, trip start time, and start/stop locations are specified, not the route the vehicle will take.

```
 <trip id="40_61_0" depart="3210.00" from="22598787#1" to
="-28480408#8" type="pkw"/>
 <trip id="19_61_0" depart="3246.00" from="24518572" to
="-230569303#0" type="pkw"/>
 <trip id="26_61_0" depart="3252.00" from="-30767370#1" to
="-25151717#2" type="pkw"/>
 <trip id="38_61_0" depart="3266.00" from="132409803#0" to
="-25320986#4" type="pkw"/>
 <trip id="48_61_0" depart="3276.00" from="5004993#0" to
="159682142#0" type="pkw"/>
```

Listing 4.1: Example of routes file for SUMO.

Listing 4.2 shows examples of the simulation results from SUMO. Each line in the result file logs the vehicle's identity, time, location, heading, speed,

and acceleration, accordingly. Notice that the vehicle's identity in Listing 4.1 correlates with the first trip specified in Listing 4.1.

```
40_61_0 3210.0 69623720 509279780 2114 0 0
40_61_0 3211.0 69623560 509279600 2114 241 24
40_61_0 3212.0 69623300 509279320 2114 387 14
40_61_0 3213.0 69622880 509278860 2114 630 24
40_61_0 3214.0 69622340 509278250 2106 835 20
```

Listing 4.2: Example of simulation results from SUMO.

Table 4.2 compares simulation results generated using Artery and SUMO, focusing on messages per second, total vehicle trips, and total messages generated. In the initial time window from midnight to 6:00 a.m., both simulation tools produced comparable results, indicating consistency in vehicle mobility and message generation during periods of low traffic volume. However, a noticeable difference can be observed during the subsequent simulation window, from 6:00 to 6:30 a.m. Despite having a 30-minute warm-up period (5:30 to 6:00 a.m.) in the Artery simulation to establish traffic flow, the output still failed to reach the levels observed in the SUMO simulation for the same period. The difference in the second simulation highlights that VANET simulations must be continuous to maintain realistic traffic dynamics and communication behaviour. Interruptions or segmented simulation runs—especially during peak traffic hours—can disrupt vehicle interactions and prevent the system from reaching a steady state, leading to inaccurate or incomplete simulation results.

From Listing 4.2: an example of simulation results for SUMO, it must be noted that multiple information from BSM/CAM are missing from the simulation's results. The crucial information indicated as important to the trajectory reconstruction is the vehicle's size and path history. Moreover, SUMO only simulates the vehicle's mobility; therefore, pseudonym-changing mechanisms have not been applied to the dataset. This missing information and these mechanisms are patched by our developed C++ code.

Table 4.2: Characteristics comparison between synthetic datasets.

Characteristics	Method	00:00–06:00		06:00–06:30	
		Artery	SUMO	Artery	SUMO
Messages per sec	Min	1	1	5	2763
	Mean	237	369	2949	4121
	Median	72	72	3268	3971
	Max	2761	2767	3788	6108
Total Trips		12427	12976	10600	17840
Total Messages		6776811	6794776	5520884	7418302

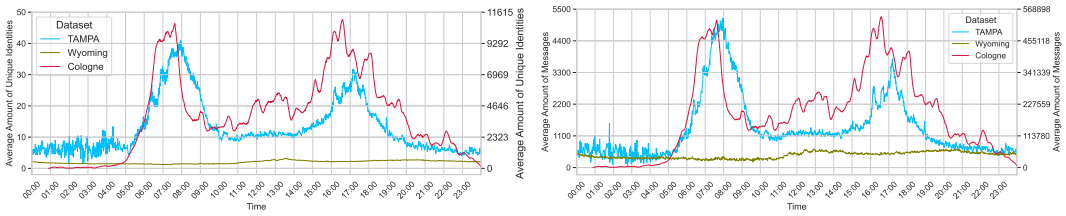
First, we randomly assigned 25 vehicle sizes to the identities in the simulation result. Although the size is randomly assigned, we assigned a weight to each vehicle's size to avoid a uniform distribution of the vehicle's size. Then, we generated path history points with the algorithm from SAE-J2495/1 [21] where the total path history points are limited to 15 points. Lastly, we applied pseudonym-changing mechanisms based on Equation (3.2) to the simulation results. The results of these steps are two datasets: one follows the ETSI standard, and the other follows the SAE standard.

4.2.3 Simulation Results

Comparing simulation results with real-world datasets is essential for evaluating the characteristics of the synthetic dataset. This comparison helps identify similarities and differences, allowing us to assess how well the synthetic data reflects actual VANET behaviour. As outlined at the beginning of this chapter, we expect each dataset to meet three key criteria: compliance with VANET standards, realistic vehicle density, and adequate spatial coverage. In the previous section, we noted that our simulation could only produce a subset of message contents due to performance limitations. To support trajectory recon-

struction, we supplemented the simulation data by manually patching critical fields that were missed during the simulation. This section evaluates the degree to which the synthetic dataset aligns with real-world datasets, considering the characteristics explored at the end of the previous section.

Figure 4.6 compares the number of unique identities and BSMs per minute between the real-world datasets (Tampa and Wyoming) and the synthetic dataset (Cologne). Note that, due to a large difference in the number of unique identities between the real-world dataset and the synthetic dataset, the real-world datasets will use the left y-axis of the graph, and the synthetic dataset will use the right y-axis. Overall, we can observe that the synthetic dataset follows the characteristics of urban density presented in the Tampa dataset.



(a) Unique identites

(b) Total BSMs

Figure 4.6: Comparison of the average number of unique identities and Basic Safety Messages (BSMs) per minute between real-world and synthetic datasets.

With the objective of generating more vehicles, the synthetic dataset achieves a much larger number of pseudo-identities and BSMs than those in Tampa and Wyoming. We must also note that, when compared with the density of pseudo-identities, the Tampa dataset has a higher density of pseudo-identities than the Cologne dataset. This might be because the Tampa dataset only focused on the city centre area, while the Cologne dataset contains empty space in the north-east and south-west, as illustrated in Figure 4.5.

Next, Figure 4.7 shows the percentage of messages with respect to the number of path history points within a message. A stark difference can be observed

between the three datasets, where most of the messages in our synthetic data contain a maximum of 15 path history points.

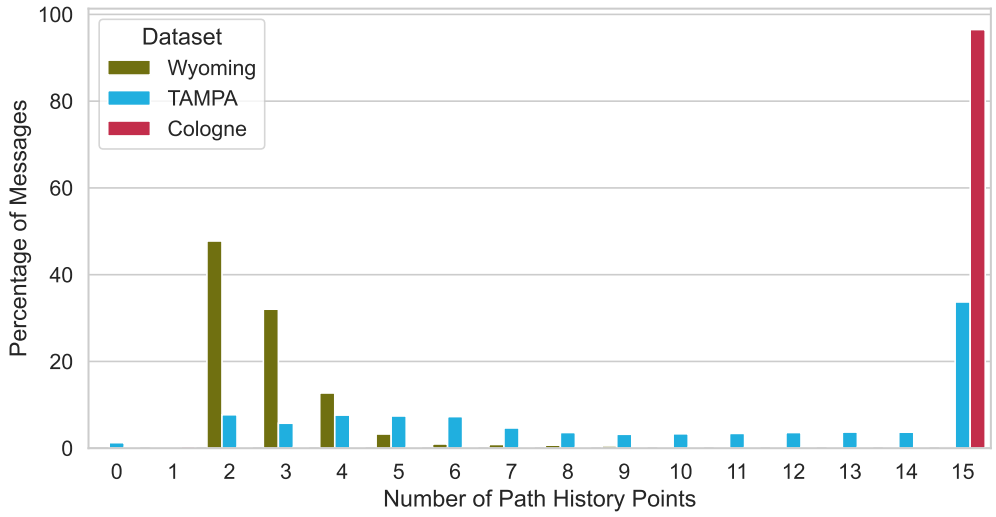
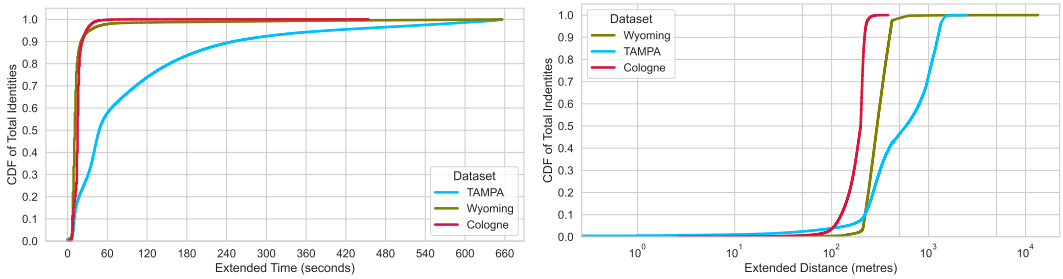


Figure 4.7: Comparison of the number of path history points in a message between real-world and synthetic datasets.

The difference in the number of path history points in a message between the real-world and synthetic datasets is unexpected, as we followed the algorithm from SAE-J2495/1 [21]. This difference may be influenced by map topology and driving behaviour, affecting how frequently new path history points are generated. Nevertheless, the Cologne and Tampa datasets share the common behaviour of being in a city area and having a maximum percentage of 15 path history points. Let us compare the path history points information with Figure 4.8, which compares the extended time and distance from the message's location. While the number of path history points in Cologne appears similar to Tampa, the temporal and spatial extensions align more closely with the Wyoming dataset, representing a highway scenario.

Based on Figures 4.7 and 4.8(a), we can infer that path history points are added more frequently in the Cologne scenario compared to both the Tampa and Wyoming datasets. This conclusion is supported by two observations:

- The Cologne dataset achieves the same number of path history points in less time than the Tampa dataset, indicating that points are logged at



(a) Extended time in the past

(b) Extended distance in the past

Figure 4.8: Comparison of the cumulative distribution of identities with the information contained in path history between real-world and synthetic datasets.

shorter intervals.

- Compared to the Wyoming dataset, Cologne accumulates more path history points over the same time duration, further reinforcing the idea of a higher logging frequency.

These patterns suggest that in the Cologne simulation, path history points are generated at a finer temporal resolution, which could be due to the differences in the path history calculation algorithm.

The differences in path history point frequency and extent have important implications for trajectory reconstruction, particularly when using Algorithm 3.2, which relies on path history data. A higher number of path history points provides more data points to match between current and past identities, thereby increasing the likelihood of linking them. Furthermore, greater temporal and spatial extensions can expand the window in which two identities may be correlated.

4.3 Summary

This chapter highlighted three reasons why the two public datasets collected from VANET pilot projects [105, 106] are inadequate for our work: noncompliance with the standards, duplicated data points, and privacy concerns from

the data publisher. Despite their inadequacy as a primary dataset for privacy quantification, we extract three characteristics from these datasets to compare with our synthetic dataset. These characteristics are the number of identities and messages throughout the day, the number of path history points in the messages, and information that leads to privacy leakage contained in the path history.

Then, we introduced a synthetic dataset that was simulated from a real-world vehicle mobility dataset, TAPASCologne [111], as an alternative dataset for privacy quantification. Although some characteristics of our synthetic dataset do not align with real-world datasets, it achieves the three goals we are looking for: standard compliance, a high number of vehicles, and large spatial coverage. Table 4.3 is provided as a chapter summary to compare multiple viewpoints between datasets.

Table 4.3: Comparison between real-world datasets and synthetic datasets per day.

Characteristics	Datasets		
	Wyoming	Tampa	Cologne
Dataset type	real-world	real-world	simulation
Scenario	highway	city	city
Message type	BSM	BSM	BSM & CAM
Coverage area (km^2)	90.84×25.98	1.65×1.47	28.94×31.27
Identities (Min)	1	72	1M (BSM)
Identities (Max)	676	43834	4.2M (CAM)
Messages (Min)	2	2.12×10^4	-
Messages (Max)	1.33×10^6	9.49×10^6	260×10^6

Chapter 5

Trajectory Reconstruction Models

Recalling our analysis in Section 3.1, we highlighted that the locations where users visit can be used to deanonymise their identity. Even though privacy-preserving mechanisms are implemented, the adversary can use multiple VANET characteristics to increase the chance of the user's deanonymisation process by leveraging these characteristics in the reversal of the privacy-preserving mechanisms. We further showed two algorithms for reconstructing a vehicle's trajectory in Section 3.2, one of which is unique to VANETs with path history information, and another that is commonly used in the literature. We also highlighted in Section 3.3 that the trajectory reconstruction models in the literature did not account for all information in the message, which may result in underestimating the adversary.

This chapter presents three distinct trajectory reconstruction approaches. These methods simulate the capabilities of an adversary attempting to link pseudonyms and reconstruct vehicle trajectories based on message content in VANET communication. The first approach leverages path history data, as described in Algorithm 3.2. This method is unique to VANET communication, as the path history is presented in Basic Safety Messages (BSMs) and Cooperative Awareness Messages (CAMs), and provides a temporal sequence of past

locations. The second and third approaches leverage a comparison between a predicted value and the actual values of a vehicle's location and kinematics. These approaches are formalised in Algorithm 3.1, and aim to assess the adversary's ability to link the pseudonyms by forecasting movement patterns.

5.1 Methodology

In Section 3.2, we formalised the privacy threat from trajectory reconstruction attacks into two approaches: path history and kinematics estimation. While the path history approach has not been previously used in the literature, the kinematics estimation approach can be recognised as a similar approach taken by the literature. Our analysis in Section 3.3 identified that the location privacy quantification framework in [82], which was designed for general location privacy, had several incompatibilities with VANETs privacy quantification. Nevertheless, the Bayesian concept could potentially offer benefits in handling sudden changes in vehicles' kinematics through the randomness in the Bayesian inference and be used in the kinematics estimation. On the other hand, Emara et al.'s work [8] was not entirely aligned with the standards, where the vehicle's kinematics were not considered as part of their calculation. Adjusting the related matrices in the Kalman filter, such as the vehicle's dynamic representation and state transformation matrices, can serve to address this gap in modelling the vehicle's movement. As a result, the gaps in the literature allow us to explore three trajectory reconstruction models, including the path history, Bayesian, and Kalman filter models.

It must be noted that each of the selected models has its own strengths and probable limitations. The path history approach is the only model that relies on historical data for linking the pseudonyms. This method potentially makes the path history approach the most reliable model for pseudonym linkage, as it does not require estimation for future location and kinematics. On the other hand, heavily relying on the path history points can also be considered

a limitation of the model, as the model will be unusable if the path history point is not present in the message.

For the Bayesian and Kalman filter models, these models used a kinematics estimation approach for linking pseudonyms; therefore, the model's strengths and weaknesses are reflected in its ability to estimate future kinematics and its ability to calculate the distance between two pseudonyms. In the Bayesian approach, as the vehicle's kinematics are represented with distributions, the distance between two pseudonyms is represented with probability, with additional randomness. Such representation gives the Bayesian approach the strength to account for fluctuating changes in the vehicle's kinematics, more specifically, changes that occur from human behaviour. The randomness can also become a weakness for the model, as it could affect deterministic predictions of location and kinematics when there are no sudden changes. On the other hand, the Kalman filter should perform well in such a scenario as most of its computation relies on deterministic functions; hence, creating the strength of the model. Finally, with the lack of randomness, the Kalman filter's weakness is that it might not perform well when kinematics that can change quickly, such as speed and heading, are accounted for as part of the location and kinematics estimation.

With the expected strengths and weaknesses of each model presented, we will proceed to the design and calculations used by each model. We design the proposal trajectory reconstruction models such that they accept the same input, the communication messages, and have a similar output format: a pair of candidate pseudonyms that will be linked together. Figure 5.1 illustrates the workflow for the three trajectory reconstruction approaches.

We assume that the trajectory reconstruction occurs offline, where the adversary collects the messages into a dataset before performing the trajectory reconstruction. Each output pair of pseudonyms represents the adversary's belief that the pseudonyms are used by the same vehicle. Then, lists of pseudonym pairs are combined into a trajectory, which can be performed

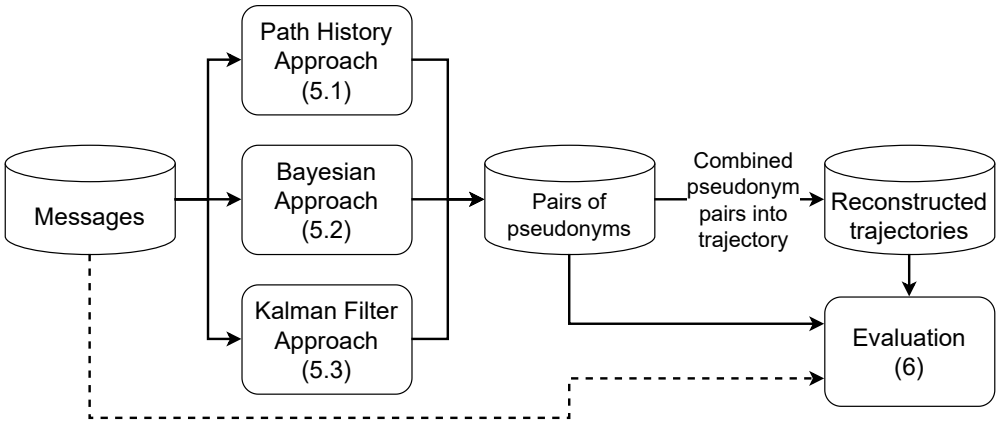


Figure 5.1: Methodology for privacy leakage comparison across all trajectory reconstruction models.

regardless of the approach used to create the list of pseudonym pairs. The same set of privacy metrics can be used for privacy quantification, as all approaches will have similar outputs, allowing comparisons between trajectory reconstruction approaches. The evaluations of these approaches will be discussed in Chapter 6. For the consistency between the trajectory reconstruction methods, Table 5.1 summarises the parameters that will be used across three methods in this chapter.

Table 5.1: Notations and variables list.

Variable	Symbol	Units	Values
Timestamp	t	ms	0 – 6535
Pseudonym	pid	-	-
Latitude	lat	0.1 µ°	-900M – 900M
Longitude	lon	0.1 µ°	-1800M – 1800M
Semi-Major Confidence	$mjConf$	1 cm	0 – 4095
Semi-Minor Confidence	$mnConf$	1 cm	0 – 4095
Heading	$head$	0.1 °	0 – 3601
Heading Confidence	$headConf$	0.1 °	1 – 127

Continued on next page

Table 5.1: Notations and variables list. (Continued)

Variable	Symbol	Units	Values
Yaw Rate	yaw	0.01 °/s	-32766 – 32767
Speed	$speed$	cm/s	0 – 16383
Speed Confidence	$speedConf$	1 cm/s	1 – 127
Acceleration	acc	0.1 m/s^2	-160 – 161
Vehicle's width	$width$	cm	0 – 1023
Vehicle's length	$length$	cm	0 – 4095
The first messsage with pseudonym i	f_pid_i	-	-
The last messsage with pseudonym i	l_pid_i	-	-
Expected value of variable var	$\hat{var}$	-	-

5.2 Path History Approach

Path history provides a vehicle's specific movement information in the past and is included in Basic Safety Messages (BSMs) and Cooperative Awareness Messages (CAMs). Although it is mainly used to estimate the lane position between the sender and receivers, as described in Section 2.1 and making it seem innocuous, it can be misused to reveal the sender's recent travel path, as shown in Figure 4.4. Additionally, our Threat card 3.9 describes that a path history may carry a more serious privacy concern. An attacker can use this information to link two different pseudonyms by considering a shared location and time between the path history point and the vehicle's location in the past message. This section focuses on Algorithm 3.2, which is used to find possible pseudonym pairs based on path history points. We also explain how an attacker might use this method to link identities.

5.2.1 Methodology

Recalling Algorithm 3.2, the algorithm iterates through one path history point at a time to identify its candidate pseudonym. While it is possible to iterate through all path history points and messages to find all the pseudonym pairs, the methodology presented in this section leverages the benefit of offline processing by finding the pseudonym pairs through database querying. Figure 5.2 illustrates the processes for finding pseudonym pairs with path history, where this figure provides a deeper look into the '*Path History Approach*' rectangle in Figure 5.1.

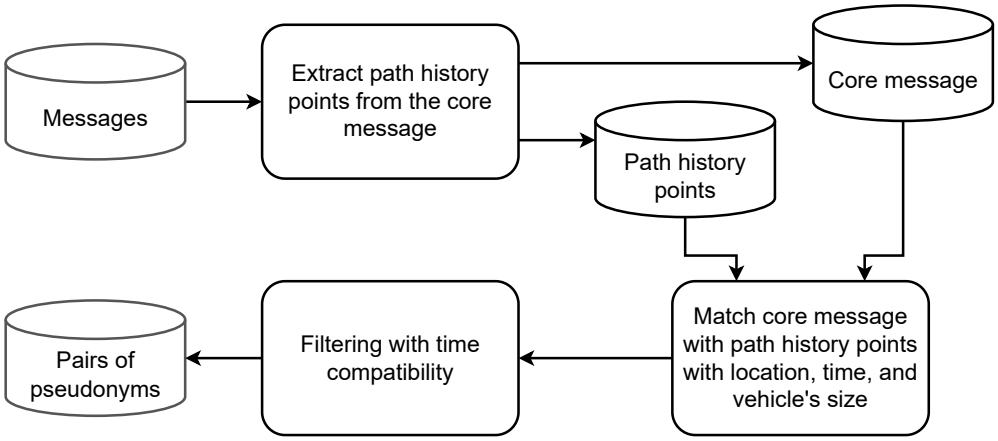


Figure 5.2: Methodology for establishing pseudonym pairs with path history.

The trajectory reconstruction process begins by separating the messages into two datasets: core messages and path history points. The core messages dataset contains the vehicle's current location, time, and size, presented in the message. The phrase 'core message' is taken from the SAE-J2735 standard [69], which represents the current location and kinematics of the vehicle. The path history points dataset is created by first extracting the data from the path history points array in each message. Then, for each path history point, the vehicle's past location and timestamp are calculated by subtracting the vehicle's current location and timestamp from the location and timestamp offsets presented in the path history point. These processes are illustrated in Algorithm 5.1.

Algorithm 5.1: Separate a messages dataset into core messages and path history points datasets.

```
Input: M;                                   /* messages dataset */
Output: coreDataset;                        /* core messages dataset */
Output: phDataset;                          /* path history points dataset */
1  coreDataset ← null;
2  phDataset ← null;
3  forall m_i in M do
4      coreTuple ← < pid, t, lat, lon, width, length >;
5      add coreTuple to coreDataset;
6      forall ph_i in Path History Array of m_i do
7          if ph_i locationOffset < max(locOffset) then
8              prevT ← t − (ph_i timeOffset × 10);
9              prevLat ← lat − ph_i latitudeOffset;
10             prevLon ← lat − ph_i longitudeOffset;
11             tuple ← < pid, prevT, prevLat, prevLon, width, length >;
12             add tuple to phDataset;
13         end
14     end
15 end
16 return coreDataset, phDataset
```

Next, the core messages and path history points datasets are joined using location, time, and vehicle size as joining keys to identify potential links between a path history point and a core message. Since the location in the core message and the location offset in the path history point share the same unit, using the location and the vehicle's size length as a joining key is a straightforward process, as both the path history point and the core message must have the same values in these fields. However, using the time as a joining key requires additional consideration. Since the time offset unit in path history points is ten times coarser than that in core messages, two records are consid-

ered a match if their timestamps differ by at most 10 milliseconds. It is worth noting that a path history point and a core message of the same pseudonym can be linked together and create a 'self-linking'. Since the path history point and the core message use the same pseudonym, this phenomenon will not be useful to the adversary and can increase computational overhead in the linking process. To prevent self-linking, any pair of joining results with the same pseudonym on both the path history point and the core data will be excluded from the results. The list of conditions that must be true for considering a core message and a path history point to be from the same vehicle is as follows:

- $location_m = location_{ph}$
- $size_m = size_{ph}$
- $|time_m - time_{ph}| \leq 10$
- $pid_m \neq pid_{ph}$

Since path history points are calculated from a vehicle's past locations and timestamps, each point should ideally correspond to a single, unique vehicle. However, we observed that some path history points can be linked to multiple core messages, introducing ambiguity into the trajectory reconstruction process. This issue can arise from two main reasons: sensor inaccuracies and differences in time resolution. First, faulty or imprecise sensors can cause small errors in the reported location. As a result, two different vehicles may appear to be at the same location simultaneously, leading to false matches between a single path history point and multiple core messages. Evidence of this can be seen in our real-world dataset analysis, where 7.99 % of the messages share identical location and kinematic values with at least one other message. Second, our matching process allows a tolerance of up to 10 milliseconds between the recovered timestamp from the path history point and the timestamp in the core message. Although this time window is relatively small, it could permit multiple core messages to fall within the same matching range for a single path history point, contributing to ambiguity.

To address the issue of matching ambiguity, we introduce a filtering technique that eliminates implausible pseudonym pairs based on their active periods. The active period of a pseudonym is a period of time during which a pseudonym is being used, and is defined by the timestamps of its first and last observed messages. Since the vehicle can only use one pseudonym at a time, the active periods of pseudonyms in a vehicle's trajectory should not overlap. This tautology can be leveraged to rule out invalid pseudonym linkages. Let $f\_pid_i\_time$ and $l\_pid_i\_time$ denote the timestamps of the first and last messages associated with the pseudonym pid_i, respectively. A pair of pseudonyms pid_i and pid_j is considered an invalid match if either of the following conditions holds:

1. $f\_pid_i\_time \geq f\_pid_j\_time \wedge f\_pid_i\_time \leq l\_pid_j\_time$
2. $l\_pid_i\_time \geq f\_pid_j\_time \wedge l\_pid_i\_time \leq l\_pid_j\_time$
3. $f\_pid_i\_time \geq f\_pid_j\_time \wedge l\_pid_i\_time \leq l\_pid_j\_time$
4. $f\_pid_i\_time \leq f\_pid_j\_time \wedge l\_pid_i\_time \geq l\_pid_j\_time$

The outcome of path history-based pseudonyms linkage is a set of pseudonym pairs that have a path history point matched with a core message, suggesting a potential link between different pseudonymous identities. As depicted in Figure 5.1, the following process combines the pseudonym pairs into the trajectory, which will be the same for all trajectory reconstruction models.

While the path history-based model is unique to VANET communication, the following two approaches are commonly used in privacy quantification in VANETs. However, as pointed out in Section 3.3, the approach taken in the literature usually omits some information in the message; therefore, the adversary can be underestimated. We will examine the Bayesian approach in the next section.

5.3 Bayesian Approach

This trajectory reconstruction approach relies on inferring the probability that two pseudonyms are used by the same vehicle and associating pseudonym pairs based on that probability. Therefore, the approach is divided into two sub-models: the inference and data association models. The inference model estimates the probability between pairs of pseudonyms. The second sub-model, the data association model, matches pseudonyms into pairs according to the probabilities from the inference model. Each pseudonym pair represents the adversary's belief that the same vehicle uses the pseudonyms. Figure 5.3 shows the workflow for this approach. Similar to the path history approach, the Bayesian approach takes a set of messages as input and outputs a set of pseudonym pairs. The dashed boxes highlight the two sub-models in this approach, where the upper box is the inference sub-model, and the lower box is the data association model.

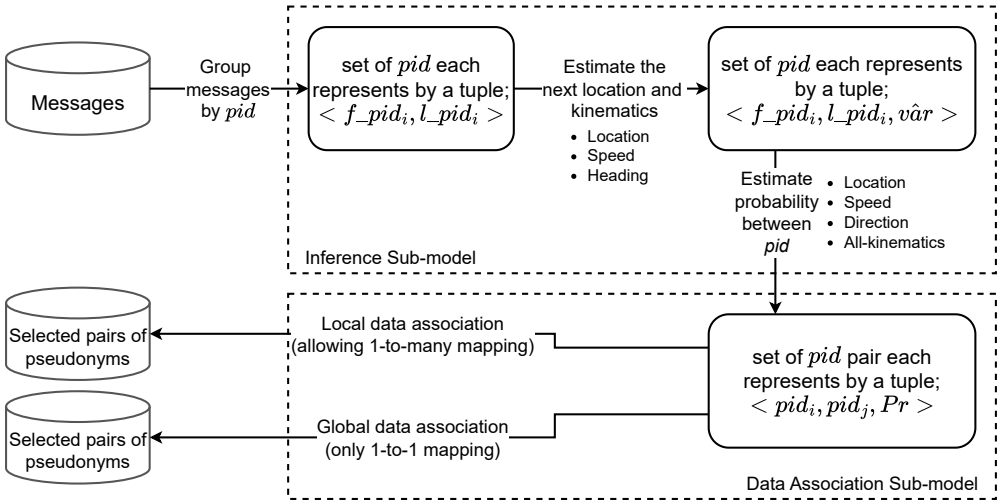


Figure 5.3: Methodology for Bayesian-based trajectory reconstruction.

5.3.1 Inference Sub-model

The objective of the inference model is to compute a probability score for every pair of pseudonyms, indicating the likelihood that both pseudonyms are used by the same vehicle. The probability is estimated using the distance

between the expected location and kinematics from one pseudonym and the location and kinematics of another pseudonym. Specifically, the predicted location and kinematics are derived by extrapolating from the state of the location and kinematics in a message. Although it is possible to perform this extrapolation for each message within a pseudonym's lifespan, the process can be optimised by computing the expected state using only the final message of each pseudonym. This simplification reduces computational overhead for the adversary while still providing a representative estimate for matching. With the extrapolation's simplification, each pseudonym can be represented with a tuple $< f\_pid_i, l\_pid_i >$, which consists of the first and last message signed by the pseudonym. Additionally, while the probability can be established between any pseudonym pair, it should only be established between the pid_i and pid_j only when the following conditions are all met:

1. $f\_pid_j\_time < l\_pid_i\_time$
2. $(f\_pid_j\_time - l\_pid_i\_time) \leq 1$ second

The first condition ensures that the first message from pid_j is received after the last message of pid_i. The second condition leverages the VANETs domain knowledge to reduce the computation further by limiting the maximum duration between these messages to be within 1 second, the maximum time duration between two consecutive messages.

For the kinematics that will be used to calculate the probability, our Bayesian inference incorporates most of the information in the messages into the distance calculation to establish the probability between two pseudonyms. Our approach of using all information in the message will be compared with the framework in the literature that may underestimate the adversary by omitting some information in the message, such as speed and heading, as pointed out in section 3.3. The inference model computes four probabilities Pr_{loc}, Pr_{head}, Pr_{speed}, and Pr_{AKs} where Pr_{AKs} is the probability of all-kinematics (AKs). Each probability represents how likely it is that the two pseudonyms were used

by the same vehicle based on its location, heading, speed, and kinematics. It must be noted that the use of Pr_{loc} is equivalent to the inference model used by the framework in the literature.

For speed and heading, the expected values can be calculated from the following equations:

$$\hat{speed}_{i+1} = speed_i + 10acc_i l \tag{5.1}$$

$$\hat{head}_{i+1} = (head_i - yaw_i l) \mod 3600 \tag{5.2}$$

where l is the time different between l_pid_i and f_pid_j. Note that the acceleration is multiplied by 10 in (5.1) to change the acceleration's unit to cm/s. In (5.2), since the heading's unit is in 0.1°, the expected heading is modulo by 3600 to maintain the heading within its value range. To estimate the expected location, latitude and longitude in degree units are first transformed into Radian units.

$$(lat'_i, lon'_i) = \left(\frac{lat_i \pi}{180}, \frac{lon_i \pi}{180}\right) \tag{5.3}$$

Then, we calculate the travel distance and Earth radius ratio, d, based on the speed *speed*, acceleration *acc*, time difference l, and Earth radius R in Equation 5.4. The estimated latitude and longitude are calculated from Equation 5.5 and 5.6, respectively.

$$d = \frac{speed_i l + \frac{1}{2} acc_i l^2}{R} \tag{5.4}$$

$$\hat{lat'_{i+1}} = \sin^{-1}\left(\sin\left(lat'_i\right)\cos\left(d\right) + \cos\left(lat'_i\right)\sin\left(d\right)\cos\left(head_i\right)\right) \tag{5.5}$$

$$\hat{lon'_{i+1}} = \tan^{-1}\left(\frac{\sin\left(head_i\right)\sin\left(d\right)\cos\left(lat'_i\right)}{\cos\left(d\right) - \sin\left(lat'_i\right)\sin\left(\hat{lat'_{i+1}}\right)}\right) + lon'_i \tag{5.6}$$

Then, latitude and longitude in radians are converted back to degrees.

$$(\hat{lat_{i+1}}, \hat{lon_{i+1}}) = \left(\frac{\hat{lat'_{i+1}} 180}{\pi}, \frac{\hat{lon'_{i+1}} 180}{\pi}\right) \tag{5.7}$$

While it is possible to use the expected values directly for distance-based matching, we enhance this approach by incorporating additional information from the message to better model uncertainty. Rather than relying solely on point estimates, we represent the expected kinematic values as a normal distribution, using confidence measures derived from the message. This probabilistic

model captures the adversary's belief about the vehicle's probable location and kinematics at the next timestamp, allowing for a more informed and flexible inference process. To represent the expected location and kinematics with a normal distribution, the calculated expected value is used as the distribution's mean, representing the highest probability or adversary's belief of the expected value. The distribution variance reflects the adversary's uncertainty in the estimation, capturing the likelihood that the actual kinematic change may deviate from the expected rate, either faster or slower. This uncertainty is encoded in the distribution's tails, where probability decreases as the value moves further from the mean. For scalar kinematic attributes such as speed and heading, the confidence value reported in the message is directly used as the variance of the corresponding normal distribution. In the case of location, which involves two-dimensional uncertainty, the message provides separate confidence values along the semi-major and semi-minor axes. To maintain a simplified one-dimensional representation for distance-based matching, we use the semi-major axis confidence as the variance of the location distribution. Each kinematic can be represented as follows:

$$\mathcal{N}\hat{lat}_{i+1} \sim Normal(\hat{lat}_{i+1}, mjConf_i)$$

$$\mathcal{N}\hat{lon}_{i+1} \sim Normal(\hat{lon}_{i+1}, mjConf_i)$$

$$\mathcal{N}\hat{speed}_{i+1} \sim Normal(\hat{speed}_{i+1}, speedConf_i)$$

$$\mathcal{N}\hat{head}_{i+1} \sim Normal(\hat{head}_{i+1}, headConf_i)$$

Lastly, the probability between pid_i and pid_j is calculated by the cumulative density function (cdf) of pid_j's kinematic value with respect to pid_i's normal distribution of the expected value.

$$Pr_{lat} = cdf(lat_j|\hat{lat}_{i+1}, mjConf_i) \tag{5.8}$$

$$Pr_{lon} = cdf(lon_j|\hat{lon}_{i+1}, mjConf_i) \tag{5.9}$$

$$Pr_{speed} = cdf(speed_j|\hat{speed}_{i+1}, speedConf_i) \tag{5.10}$$

$$Pr_{head} = cdf(head_j | \hat{head}_{i+1}, headConf_i) \tag{5.11}$$

Then, the location and all-kinematics probabilities are derived as follows:

$$Pr_{loc} = Pr_{lat} \times Pr_{lon} \tag{5.12}$$

$$Pr_{AKs} = Pr_{speed} \times Pr_{head} \times Pr_{loc} \tag{5.13}$$

Then, four tuples are generated to represent the likelihood of two pseudonyms being used by the same vehicle based on the distance between location and/or kinematics.

- Location-based: $< pid_i, pid_j, Pr_{loc} >$
- Speed-based: $< pid_i, pid_j, Pr_{speed} >$
- Heading-based: $< pid_i, pid_j, Pr_{head} >$
- All-kinematics-based: $< pid_i, pid_j, Pr_{AKs} >$

To demonstrate the possible outcomes from the inference model, an example scenario with mock-up outcomes from the inference model is provided below. Let us assume that the adversary is able to capture messages from an unknown number of vehicles. The adversary is able to group these messages based on the pseudonym certificate on the messages, and identify that six pseudonyms were used within the captured messages: $pid_0 - pid_5$. Furthermore, based on the first and last messages signed by the pseudonyms, the adversary can separate these pseudonyms into two groups, where pseudonyms in the first group, $pid_0 - pid_2$, have the same timestamp on the last message, and pseudonyms in the second group, $pid_3 - pid_5$, have the same timestamp on the first message. The adversary can identify the likelihood that two pseudonyms are used by the same vehicle by using the inference sub-model to calculate the probability between each pair of pseudonyms across the two groups. Since the duration of the pseudonyms in the first group ended before the pseudonyms in the second group, the pseudonyms in the first group become

a target pseudonym, *target pid*, and the pseudonyms in the second group become candidates for each target pseudonym. Table 5.2 shows the results from the inference model from the example scenario. Let a function $Pr(i, j, var)$ describe the probability of a candidate pseudonym j being used by the same vehicle as pseudonym i with respect to the kinematic var as the distance calculation parameter. The bold numbers depict the highest probability among the $Pr(i, *, var)$ list. These tuples are passed to the next sub-model to select a set of the most likely pairs of pseudonyms.

Table 5.2: Example of the inference model results.

Target pid_i	*Candidate* pid_j	$Pr(i, j, var)$			
		speed	*head*	*loc*	*AKs*
pid_0	pid_3	**0.72**	**0.68**	**0.25**	**0.15**
	pid_4	0.13	0.14	0.12	0.08
	pid_5	0.15	0.18	0.04	0.02
pid_1	pid_3	0.08	0.16	0.17	0.25
	pid_4	**0.83**	**0.68**	**0.53**	**0.45**
	pid_5	0.09	0.16	0.05	0.20
pid_2	pid_3	0.05	**0.41**	0.18	0.05
	pid_4	0.25	0.39	**0.25**	0.16
	pid_5	**0.70**	0.20	0.21	**0.18**

5.3.2 Data Association Sub-model

The objective of the data association model is to identify a pseudonym pair by selecting a candidate pseudonym pid_j that is most likely to belong to the same vehicle as a given target pseudonym pid_i. This selection process, associating pid_j to pid_i, leverages the probability scores produced by the inference model. We implemented two approaches for the data association model, including local and global data associations. The difference between the two approaches

is the number of target pseudonyms the adversary accounts for at a time. The local data association approach considers a single target pseudonym, while the global data association approach considers all target pseudonyms together. Recalling the results from the inference model in Table 5.2, the local data association will split the table according to the target pseudonyms and consider three rows at a time, namely rows 1–3, 4–6, and 7–9. On the other hand, the global data association will consider every row before associating the pseudonym pairs.

In the **local data association approach**, the adversary focuses on one target pseudonym at a time. The local data association selects a candidate pseudonym with the highest probability from the $Pr(i, *, var)$ set. Another condition is added to the consideration, where the probability should be greater than or equal to 0.05, since a lower probability can be insignificant. For example, suppose the adversary chooses the pseudonym pid_0 as their target pseudonym and uses location as the tracking parameter; they will only consider the probability in the *loc* column and only consider rows 1–3; therefore, the pid_3 will be the most suitable pair for pid_0. By considering only a portion of the probability set, the adversary can select a pseudonym pair with low computational complexity. The local data association can be expressed as follows:

$$\begin{aligned} &Pr(i, j, var) = max(Pr(i, *, var)) \\ &\wedge\ Pr(i, j, var) \geq 0.05 \end{aligned} \tag{5.14}$$

When both conditions in Equation (5.14) are satisfied, the adversary establishes a pair between pid_i and pid_j, indicating a belief that both pseudonyms belong to the same vehicle. However, in some cases, the target pseudonym pid_i may remain unpaired. This occurs when no candidate pseudonyms are available, or all candidates yield a probability $Pr(i, *, var) < 0.05$, indicating insufficient evidence to support a reliable match.

A key limitation of the local data association method is its potential to associate a single candidate pseudonym pid_j with multiple target pseudonyms

pid_i, resulting in one-to-many mappings. This issue arises because the method treats each target pseudonym independently during the pairing process without considering global consistency across the probability set. For example, as shown in Table 5.2, the local data association approach processes rows 1–3, 4–6, and 7–9 separately for the target pseudonyms pid_0, pid_1, and pid_2, respectively. The method produces consistent one-to-one associations based on speed or all-kinematics tracking parameters. However, when using heading or location as the tracking parameter, the independent processing can lead to ambiguous one-to-many pairings. This issue is evident when candidate pseudonym pid_5 is linked to both pid_0 and pid_2, as it has the highest association probability in both cases under heading-based tracking. A similar one-to-many association occurs with location-based tracking, where candidate pid_4 is paired with both pid_1 and pid_2. These examples highlight the inherent drawback of the local data association approach and emphasise the need for a globally consistent data association approach.

The **global data association approach** addresses the limitations of the local data association by evaluating all probabilities $Pr(*, *, var)$ collectively before associating a pair between pseudonyms. Unlike the local approach, which considers each target pseudonym independently, the global method examines the full set of probabilities to ensure consistency and avoid conflicting matches. For instance, when using location as the tracking parameter, the global method simultaneously evaluates the entire *loc* column in Table 5.2 to determine the most plausible pseudonym pairings. However, it is important to note that each probability $Pr(i, j, var)$ is computed solely from the kinematics in the first message of candidate pseudonym pid_j based on its similarity to the estimated kinematics of target pseudonym pid_i and does not incorporate information about how likely pid_i is to match with other candidates, nor how likely pid_j is to match with other target pseudonyms. As a result, the probability of $Pr(pid_0, pid_3, loc) = 0.25$ may not equal $Pr(pid_2, pid_4, loc) = 0.25$; therefore, the whole set of $Pr(*, *, loc)$ must be normalised to highlight the similarity

and difference between these probabilities. Using the same method as [8], this thesis used the nearest neighbours probabilistic data association for our global data association approach, and the probability can be recalculated as follows.

$$T_{i,var} = \sum Pr(i, *, var) \tag{5.15}$$

$$M_{j,var} = \sum Pr(*, j, var) \tag{5.16}$$

$$Pr(i, j, var) = \frac{Pr(i, j, var)}{T_{i,var} + M_{j,var} - Pr(i, j, var)} \tag{5.17}$$

Equation (5.15) accounts for the probabilities of all candidates pid_j that can be associated with the target pid_i. Similarly, Equation (5.16) accounts for the probabilities of all targets pid_i with which the candidate pid_j can be associated. Lastly, Equation (5.17) normalises the probability $Pr(i, j, var)$ according to its original probability that the candidate pid_j is used by the same vehicle as the target pid_i, the overall probability that the candidate pid_j can be associated with, and the overall probability that the target pid_i can be associated with.

To overcome the drawback of the local data association approach, the global data association adds additional conditions where each pseudonym can be paired once as a target pseudonym and a candidate pseudonym. Therefore, the global data association's pairing conditions can be expressed as follows.

$$\begin{aligned} & Pr(i, j, var) = max(Pr(*, *, var)) \\ & \wedge\ Pr(i, j, var) \geq 0.05 \\ & \wedge\ pid_i \text{ has not been paired as a target pseudonym} \\ & \wedge\ pid_j \text{ has not been paired as a candidate pseudonym} \end{aligned} \tag{5.18}$$

Since the first condition in Equation (5.18) selects the highest probability from the set $Pr(*, *, var)$, a straightforward and effective pseudonym pairing approach begins by sorting all probability values in descending order. The global data association algorithm then iterates through this sorted list, evaluating each candidate pair in order of decreasing probability. This approach satisfies the first condition and facilitates efficient evaluation of the third and

fourth conditions, where pseudonyms can be verified against previously established pairs. The global data association can be expressed in pseudocode as in Algorithm 5.2. While the global data association can achieve a one-to-one mapping between a target and a candidate pseudonym, recalculating probabilities and sorting the list of probabilities requires higher computation. Moreover, the process cannot be performed on the fly since all probabilities must be calculated and sorted before the pairing process.

Algorithm 5.2: Associating pseudonym pairs with global data association approach.

```
1  pseudonymPairs ← null;
2  targetPIDs ← null;
3  candidatePIDs ← null;
4  sortedProbabilities ← sort(Pr(∗, ∗, var), descending);
5  forall Pr(i, j, var) in sortedProbabilities do
6      if Pr(i, j, var) > 0.05 then
7          if pid_i ∉ targetPIDs ∧ pid_j ∉ candidatePIDs then
8              add (pid_i, pid_j) to pseudonymPairs;
9              add pid_i to targetPIDs;
10             add pid_j to candidatePIDs;
11         end
12     end
13     else
14         break;
15     end
16 end
17 return pseudonymPairs;
```

In summary, the Bayesian approach demonstrates how an adversary can exploit Threat 3.10 by linking two pseudonyms based on calculated location and kinematics. This approach is structured into two sub-models: the infer-

ence and data association models. Through Bayesian inference, the inference model estimates the probability that a candidate pseudonym belongs to the same vehicle as a target pseudonym. Two data association approaches, each with strengths and weaknesses, were presented for assigning a candidate to the most plausible target pseudonym. The local approach is computationally efficient but may lead to one-to-many mappings, introducing ambiguity. In contrast, the global approach produces one-to-one associations by considering all probability estimates simultaneously at the cost of increased computational complexity. In the next section, we will explore another approach for estimating the probability between target and candidate pseudonyms, which can be used as a substitution for the inference sub-model.

5.4 Kalman Filter Approach

The Kalman filter is a well-established method for estimating and predicting the state of a dynamic system. It can be used similarly to the inference sub-model in the Bayesian approach previously presented in the last section. While the Kalman filter has been used in the literature for quantifying VANETs privacy [8], the literature has not utilised all available information in the state estimation. We revisit the Kalman filter model to compare our approach with the approach based on the literature in [8], where most of the information in the message will be utilised for state estimation. Additionally, we follow the approach taken in the previous section, where state estimation is applied to the last message used by a pseudonym, rather than estimating the vehicle state throughout the pseudonym's duration. Figure 5.4 shows the overall process of the Kalman filter approach taken in this thesis. Similar to the previous approaches, the Kalman filter approach takes a message set as input and outputs a list of pseudonym pairs. Moreover, it consists of two sub-models: state estimation and data association sub-models. The similarity in the data association's input and output allows it to be reused from the Bayesian data

association sub-model approach.

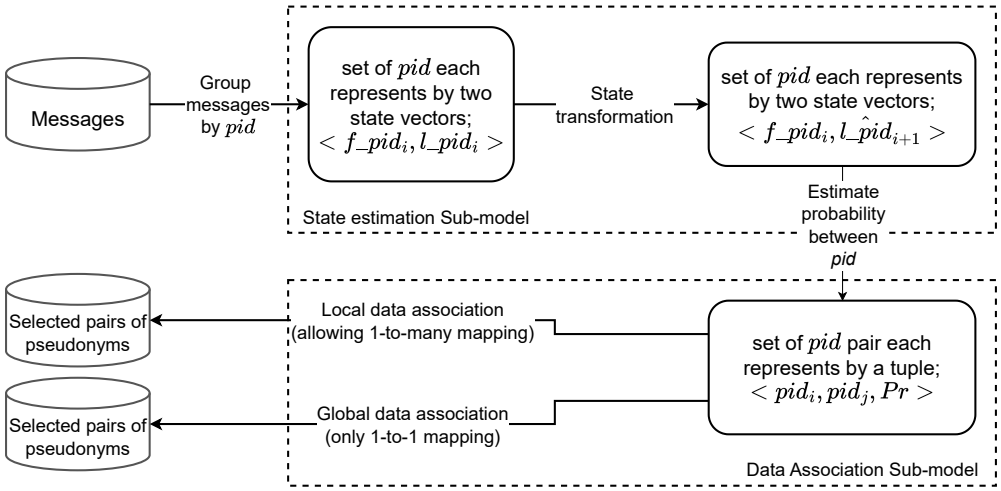


Figure 5.4: Methodology for Kalman filter-based trajectory reconstruction.

The difference between the Kalman filter and Bayesian approaches lies in the techniques for estimating the vehicle's next state and calculating probability. Unlike the Bayesian method, the Kalman filter relies on matrix operations for *state transformation*. This matrix-based formulation allows the Kalman filter to efficiently and structuredly model the vehicle's kinematics and uncertainties. Furthermore, because the vehicle's state is represented as a vector, the probability between a target and a candidate pseudonym can be calculated using matrix-based techniques. We will begin this section by explaining how the Basic Safety Message (BSM) and Cooperative Awareness Message (CAM) are mapped to a vector representation. Then, we will explain the state transformation and estimation phase and how the probability is calculated using the matrix operation approach.

5.4.1 Dynamic System Representation

Before we explain our approach for vehicle state estimation and probability calculation, we will outline our approach for constructing the Kalman filter's necessary vectors and matrices. The Kalman filter models the vehicle and its kinematics using a state vector, where each element represents a specific kinematic state of the vehicle. These elements capture the information of interest

and enable the state estimation of future states through matrix operations. To ensure consistency with the Bayesian approach, the state vector in our model includes key vehicle attributes: location, speed, heading, and the rate of change in those kinematics. Note that, in this section, the bold text will be used to describe a vector or a matrix. The state vector can be represented as follows:

$$\mathbf{state} = \begin{bmatrix} lat & lon & speed & head & acc & yaw \end{bmatrix}^T$$

Since the required values are presented in the message, they can be directly extracted from the last message used by a pseudonym. Note that all variables used in Equations (5.1) to (5.7) are represented in the state vector to reflect the similarity between the Kalman filter's state prediction and the Bayesian's state estimation.

Given that the Kalman filter operates across two domains, the system state space and the measurement space, we define two additional sensor measurement vectors: the location measurement vector and the all-kinematics measurement vector. In a typical Kalman filter process, these sensor measurements are used to update the system state after the state transformation. However, in this thesis, we utilise the sensor measurement vectors to calculate the distance between the predicted system state and the actual sensor measurements, which will be described in subsection 5.4.3. The location measurement vector, $\mathbf{z_{loc}}$, and the all-kinematics measurement vector, $\mathbf{z_{aks}}$, are defined as follows:

$$\mathbf{z_{loc}} = \begin{bmatrix} lat & lon \end{bmatrix}^T$$

$$\mathbf{z_{aks}} = \begin{bmatrix} lat & lon & speed & head \end{bmatrix}^T$$

Having two measurement vectors enables comparison between two privacy quantification approaches: (1) the location-based approach commonly used in previous VANET privacy literature and (2) the all-kinematics-based approach proposed in this thesis. Since the measurement vector will be used for comparing with the predicted state, it is essential to note that the values in the

measurement vector are taken from the first message used by a pseudonym, rather than the last message as used in the state vector.

Recalling Equation (2.9), the Kalman Gain calculation, two matrices are used for weighting between the predicted state and the sensor measurement: the covariance of processing noise Q and the covariance of measurement noise R. Although the covariance of process noise can be calculated by smoothing the system state throughout the pseudonym duration, for simplicity of calculation, we estimate the covariance Q similarly to Equation (2.11), which is used for initialising the covariance. Since acceleration and yaw rate are not affected by other variables in the system state, and can influence other variables, we used the variance of these variables to construct the covariance of the process noise.

$$\mathbf{Q} = \begin{bmatrix} \frac{\Delta t^4}{4}\sigma_a^2\sigma_y^2 & 0 & \frac{\Delta t^3}{2}\sigma_a^2 & \frac{\Delta t^3}{2}\sigma_y^2 & 0 & 0 \\ 0 & \frac{\Delta t^4}{4}\sigma_a^2\sigma_y^2 & \frac{\Delta t^3}{2}\sigma_a^2 & \frac{\Delta t^3}{2}\sigma_y^2 & 0 & 0 \\ \frac{\Delta t^3}{2}\sigma_a^2 & \frac{\Delta t^3}{2}\sigma_a^2 & \Delta t^2\sigma_a^2 & 0 & 0 & 0 \\ \frac{\Delta t^3}{2}\sigma_y^2 & \frac{\Delta t^3}{2}\sigma_y^2 & 0 & \Delta t^2\sigma_y^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & \Delta t\sigma_a^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & \Delta t\sigma_y^2 \end{bmatrix} \tag{5.19}$$

where σ_a^2 and σ_y^2 are the variances of acceleration and yaw rate, respectively. For the covariance of measurement noise, we use the confidence values of each variable from CAM/BSM to construct the covariance R. Since there are two measurement vectors, $\mathbf{z_{loc}}$ and $\mathbf{z_{aks}}$, two covariances of measurement noise are defined to be used with the corresponding measurement vector.

$$\mathbf{R_{loc}} = \begin{bmatrix} mjConf & 0 \\ 0 & mjConf \end{bmatrix} \tag{5.20}$$

$$\mathbf{R_{aks}} = \begin{bmatrix} mjConf & 0 & 0 & 0 \\ 0 & mfConf & 0 & 0 \\ 0 & 0 & speedConf & 0 \\ 0 & 0 & 0 & headConf \end{bmatrix} \tag{5.21}$$

5.4.2 State Transformation

The Kalman filter utilises state transformation to estimate the vehicle state for the next time step by using its current state and a transformation matrix. In this thesis, the state transformation will be used as a prediction for the vehicle's next state since there are no further sensor measurements to correct the state estimation. The transformation matrix describes how each element in the system state influences the other elements for extrapolating into the next time step. The influence of acceleration and yaw rate on others can be observed and reproduced from the Equations (5.1) – (5.2). However, since the changes in latitude and longitude are nonlinear transformations, the Equations (5.3) – (5.7) will not be able to fit into the matrix operation. To maintain consistency in location estimation between the Kalman filter and the Bayesian approach, we calculate changes in latitude and longitude separately from the Kalman filter's state estimation according to the Equations (5.3) – (5.7). Then, we replace the states of latitude and longitude with the results from Equation (5.7) after Equation (5.22). The transformation matrix is described as follows:

$$\mathbf{T} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 10 & 0 \\ 0 & 0 & 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Recall from the equation (2.6), the vehicle's next state can be calculated as follows:

$$\hat{\mathbf{state}} = \mathbf{T} \cdot \mathbf{state} + \mathbf{wt} \tag{5.22}$$

where **wt** is a vector of process noise taken from a white Gaussian distribution $\mathcal{N}(0, Q)$, which represents the uncertainty in the prediction, and Q is the covariance of the process noise.

Recalling the Kalman process in Section 2.4, the Kalman filter will update the vehicle's state based on incoming sensor measurements. This process uses

the predicted state $\mathbf{\hat{state}}$, the sensor measurement $\mathbf{z}$, the process noise $\mathbf{Q}$, and the measurement noise $\mathbf{R}$ using the Equations (2.7)to (2.10). However, in this thesis, these steps *are ignored* since the $\mathbf{\hat{state}}$ is calculated from the last message used by a pseudonym; hence, there are no sensor measurements to be used in the updating process. It is worth noting that, by ignoring the state update process, the process noise $\mathbf{Q}$ will not be updated, and maintain its initial value throughout the calculation.

5.4.3 Probability Estimation

Similar to the Bayesian approach, the probability that two pseudonyms are used by the same vehicle is calculated based on the distance between the predicted "vehicle state" of one pseudonym and the "vehicle's state" in the other pseudonym. With Kalman filter, the predicted vehicle's state is the estimated system state $\mathbf{\hat{state}_i}$ calculated from the Equation (5.22), and the vehicle's state from another vehicle is the sensor measurements $\mathbf{z_{loc}}$ or $\mathbf{z_{aks}}$ defined in Subsection 5.4.1. Firstly, the estimated system state needs to be converted to the estimated sensor measurement $\hat{\mathbf{z}}$, which has the same size as the sensor measurement vector. With two sensor measurement vectors defined in Subsection 5.4.1, we define two estimated sensor measurements as follows:

$$\hat{\mathbf{z_{loc}}} = \mathbf{H_{loc}} \cdot \mathbf{\hat{state}} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \mathbf{\hat{state}} \tag{5.23}$$

$$\hat{\mathbf{z_{aks}}} = \mathbf{H_{aks}} \cdot \mathbf{\hat{state}} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \mathbf{\hat{state}} \tag{5.24}$$

Then, the innovation vector can be calculated, which represents the differences between the kinematics in the estimated sensor measurement of pseudonym i and the kinematics in the actual sensor measurement from pseudonym j in a vector format.

$$\mathbf{y}_{ij} = \mathbf{z}_j - \hat{\mathbf{z}}_i \tag{5.25}$$

Note that we drop the subscriptions *loc* and *aks* from the sensor measurement vector, as the calculations for location-based and all-kinematics-based are the same. The subscriptions *loc* and *aks* will be added if there is a difference in context between them. In addition to the innovation, the covariance of the innovation can also be calculated. The covariance represents the uncertainty of the innovation when both the covariance of the process noise and the covariance of the measurement noise are considered.

$$\mathbf{S}_{ij} = \mathbf{H}\mathbf{Q}_i\mathbf{H}^T + \mathbf{R}_j \tag{5.26}$$

The observation matrix $\mathbf{H}$ is defined in Equations (5.23) and (5.24), the process noise covariance is defined in Equation (5.19), and the measurement noise covariance is defined in Equations (5.20) and (5.21).

Next, the Mahalanobis distance between the estimated sensor measurement and the actual sensor measurement, d_{ij}^2, is calculated. As explained in subsection 2.4.2, the Mahalanobis distance is often used with the Kalman filter for track maintenance, which includes the decision to assign a measurement to an existing track, create a new track, and end an existing track. In our case, the Mahalanobis distance is used to estimate the closeness between the predicted sensor measurement and the actual sensor measurement.

$$d_{ij}^2 = \mathbf{y}_{ij}^T\mathbf{S}_{ij}^{-1}\mathbf{y}_{ij} \tag{5.27}$$

It is important to note that when computing the Mahalanobis distance between a target pseudonym i and other candidate pseudonyms, the estimated sensor measurement and process noise covariance associated with pseudonym i remain constant. As a result, the Mahalanobis distance d_{ij}^2 between pseudonyms i and j is primarily influenced by the sensor measurement and measurement noise covariance of pseudonym j. Additionally, suppose the innovations $\mathbf{y}_ij$ and $\mathbf{y}_ik$ are identical; the resulting Mahalanobis distances d_{ij}^2 and d_{ik}^2 can still differ due to variations in the corresponding measurement noise covariance matrices.

To align with the Bayesian approach, the Mahalanobis distance is converted into a probability that reflects the likelihood that candidate pseudonym

j corresponds to the same vehicle as the target pseudonym i. Although the Mahalanobis distance follows a chi-squared distribution, which is commonly used for binary decision-making tasks such as gating in Kalman filtering, it is less suited for ranking multiple candidates. For this reason, instead of using the tail probability from the chi-squared distribution, we use the Gaussian likelihood function, which provides a more appropriate and interpretable measure for ranking candidate pseudonyms based on their similarity to the predicted state [112, 113]. The Gaussian likelihood between a target pseudonym i and the candidate pseudonym j can be calculated as follows:

$$Pr(i, j, var) = \frac{e^{-d_{ij}^2/2}}{(2\pi)^{k/2}\sqrt{|\mathbf{S}_{ij}|}} \tag{5.28}$$

where $|\mathbf{S}_{ij}|$ is the determinant of the innovation covariance and k is the size of $\mathbf{y}_{ij}$, which are 2 and 4 for $Pr(i, j, loc)$ and $Pr(i, j, aks)$, respectively. Since the end product of this process is similar to the inference sub-model in the Bayesian approach, it is possible to apply the data association sub-model from Subsection 5.3.2 to decide which candidate pseudonym should be associated with the target pseudonym.

5.5 Summary

This chapter presented our contribution to three trajectory reconstruction models for measuring privacy leakage in VANET communication. Our three trajectory reconstruction models include the path history-based, the Bayesian approach, and the Kalman filter approaches. These three approaches correspond with the privacy threats we identified in Section 3.1, where the path history-based model corresponds to Listing 3.9, and the Bayesian and the Kalman filter approaches correspond to Listing 3.10. As depicted in Figure 5.1, the common objective between these models is to identify a pair of pseudonyms that are likely to be used by the same vehicle. Then, the adversary can extend from a pair of pseudonyms into a list of pseudonyms that represents the vehicle's trajectory. We designed our trajectory reconstruction models to use the

same set of inputs (a set of messages used in VANET communication) and to have the same output format (pairs of pseudonyms), allowing each model to serve as a substitute for the other models presented in this section. While our three approaches for trajectory reconstruction can be used interchangeably, the underlying approaches make each model unique.

The path history-based approach (Section 5.2) introduces a novel method of trajectory reconstruction using historical spatial-temporal data embedded within messages. This approach uses a path history point, the past spatial-temporal information on the message, to match with the location information in the preceding message; hence, creating a pair of pseudonyms when those messages are signed with different pseudonyms. To our knowledge, this method of creating a pair of pseudonyms has not been explored in the literature of VANET privacy quantification.

The Bayesian approach (Section 5.3) explores the feasibility of probabilistic reasoning for trajectory reconstruction. While the Bayesian method has already been explored in the context of location privacy in the literature [82], it has not yet been applied in the context of VANET communication, where frequently broadcasting precise locations is a key aspect of the system (see further discussion in Section 3.3). Our Bayesian model relies on an inference sub-model, which estimates the probability that two pseudonyms were used by the same vehicle. This sub-model is built on the concept of updating the adversary's belief about the probability of two pseudonyms. Then, the data association sub-model selects the most likely pair of pseudonyms from the list of probabilities generated in the inference sub-model.

The Kalman filter approach (Section 5.4) revisits the use of such techniques for trajectory reconstruction in VANETs. Although the Kalman filter approach has been used in the context of VANETs' privacy quantification in the literature [8], we extend their model by incorporating pseudonym-changing mechanism standards and leveraging other kinematic quantities in the messages as part of the vehicle's state. As a result, we proposed a Kalman filter

approach in which the vehicle's state is defined by its location, speed, heading, acceleration, and yaw rate. The Kalman filter approach uses matrix operations to predict the vehicle's state and estimate the probability of two pseudonyms, which resembles the inference submodel in the Bayesian approach. Because it produces a similar output, the Kalman filter can also use the same data association submodel as the Bayesian approach.

Each model demonstrates a distinct methodological approach while sharing the same conceptual goal in which to leverage information in VANET communication to reconstruct vehicle trajectories. The choice of model may depend on the adversary's expertise or computational constraints, as well as the relative performance of each method. The preference for one approach over the others can also be determined by the model's performance, which we will evaluate in the next chapter.

Chapter 6

Evaluations

In the previous chapter, we developed three trajectory reconstruction models to simulate privacy attacks on VANET communication, including the path history-based, the Bayesian approach, and the Kalman filter models. Since the ability to accurately reconstruct a vehicle's trajectory reflects a low level of privacy preserved in the communication, the privacy of the users and the model's efficiency are two sides of the same coin. Similarly, a higher level of incorrectness in the trajectory reconstruction model may imply a higher level of privacy preserved in the system.

Given the known complementary relationship between the user's privacy and the efficiency of the trajectory reconstruction model, this chapter presents the evaluations of the three trajectory reconstruction approaches presented in Chapter 5. The main objective of this chapter is to translate our theoretical privacy threats analysis presented in Chapter 3 into quantifiable numbers, where we can compare privacy preservation/loss across the three proposed models, the standards used in the pseudonym-changing mechanisms (SAE and ETSI), and the information available to the adversary (location-based and all-kinematics-based). It must be noted that, throughout the evaluations in this chapter, the '*Location-based*' metrics are equivalent to the approach taken by the literature in VANETs privacy quantification, and the '*All-kinematics-based*' metrics are our proposed approach for leveraging all information in the

message. Our evaluation is designed to reflect the incremental steps of the trajectory reconstruction process, where we can highlight the privacy leakage that occurred in each step of trajectory reconstruction.

We begin this chapter by evaluating an ambiguous trajectory that may occur during the usage of path history reconstruction, or in the local data association in Bayesian and Kalman filter approaches in Section 6.1. These analyses account for the concerns from the adversary's perspective, while the path history-based model and the local data association approach are used for trajectory reconstruction. Then, recalling the common methodology between the three approaches in Figure 5.1, Section 6.2 presents evaluations of the first output from trajectory reconstruction models, the pairs of pseudonyms. In Section 6.3, we present evaluations of the reconstructed trajectories to emphasise the adversaries' capability and to reflect the potential privacy leakage in VAENTs' communication.

6.1 Ambiguous Trajectories

One important issue that may arise in the path history trajectory reconstruction and local data association in both the Bayesian and Kalman filter approaches is ambiguous trajectories. In the path history approach, the ambiguous trajectories issue arises when a single path history point corresponds to multiple potential messages, or conversely, when multiple path history points align with a single message. As discussed in Section 5.2, such ambiguity may result from sensor errors or differences in the time unit between the path history point's time offset and the core message timestamp. Meanwhile, in the local data association, we pointed out in subsection 5.3.2 the potential side effects where two target pseudonyms are linked with a single candidate pseudonym.

Firstly, let us define the term 'ambiguous trajectories'. The ambiguous vehicles' trajectories are those that appear as a single pseudonym at some point in the trajectory, i.e., two or more trajectories converge, or a trajectory

diverges at one point. A shared pseudonym is not ideal, as two vehicles should not appear at the same location and time under the same pseudonym. As stated earlier, ambiguity arises when a pseudonym is linked to two or more pseudonyms. These links between pseudonyms can be associated based on the matching between the vehicle's current location and time in a message and the location and time retrieved from the path history offset, or the Equation (5.14) for the local data association sub-model. Additionally, specific to the path history approach, in order to define that the trajectories are ambiguous, the linked pseudonyms must have temporal overlap. This additional condition is required because a long path history may be matched with more than one pseudonym, while these pseudonyms are used by the same vehicle, one after another.

Figure 6.1 illustrates the distinction between a normal trajectory and the ambiguous trajectories with the path history reconstruction approach. A solid black line depicts a timeline of the vehicle based on timestamps in the messages. A letter on the solid black line represents the pseudonym used by the vehicle during that time. A dashed blue line shows a timeline of the vehicle based on its path history. A horizontal dotted line with diamonds depicts a matched point between the timestamp in a message and a timestamp based on a path history point. Since there is no overlap of solid black lines between pseudonyms A, B, and C, Figure 6.1 (a) shows a normal trajectory even though pseudonym C can be linked with two pseudonyms. On the other hand, since there is an overlap of solid black lines between pseudonyms D and E, and these pseudonyms can be linked with a pseudonym F, Figure 6.1 (b) depicts ambiguous trajectories. In this scenario, the three pseudonyms, D, E, and F, will be considered ambiguous.

It is worth noting that the ambiguous trajectories highlight the problem that the adversary may face during the trajectory reconstruction. Whether the ambiguity arises from the faulty sensor or the adversary's incapability, a general perception of ambiguous trajectories is that high ambiguity in the

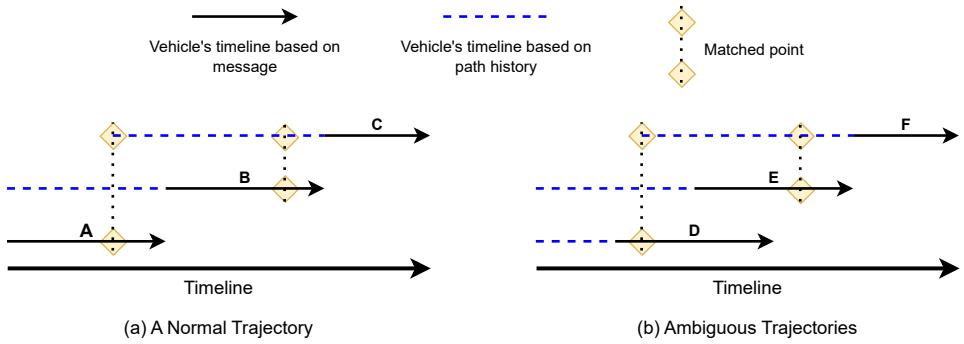


Figure 6.1: Timelines of a normal trajectory and an ambiguous trajectory.

reconstructed trajectories could indicate high privacy preservation for users in VANET communication. In this section, we evaluate the number of ambiguous trajectories without considering the correctness of the pseudonyms' linkage. The correctness of the pseudonyms' linkage will be evaluated in the next section.

6.1.1 Ambiguous Trajectories in Path History Approach

With the term ambiguous trajectories defined, we will now evaluate the ambiguity in the reconstructed trajectory from the path history approach. We first apply the process of creating pseudonym pairs with the path history methodology as depicted in Figure 5.2. Note that the "filtering with time compatibility" only discards a "target-candidate" pair of pseudonyms. This condition does not apply to the time compatibility between two target pseudonyms or two candidate pseudonyms; hence, the ambiguous trajectories may occur during the trajectory reconstruction phase depicted in Figure 5.1.

To quantify the effect of ambiguous trajectories, we will account for the number and percentage of pseudonyms that are present in these trajectories. The number of affected pseudonyms is chosen instead of the number of trajectories since the total number of pseudonyms can be extracted from the dataset. On the other hand, to account for the number of trajectories, the ground truth of the trajectory must be available. Additionally, without the

requirements for ground truth, we could also use the public dataset in this evaluation. Table 6.1 summarises the percentage of pseudonyms that appear on the ambiguous routes compared to the overall pseudonyms.

Table 6.1: Percentage of pseudonyms that appear on ambiguous trajectories in the path history trajectory reconstruction approach.

Matching Type	Dataset	Total Pseudonyms	Ambiguous Pseudonyms	
			Number	%
Exact match	THEA	3.6M	767,555	21.15%
	Cologne (CAM)	4.2M	0	0%
	Cologne (BSM)	1M	0	0%
Ranged match	THEA	3.6M	1,069,525	29.47%
	Cologne (CAM)	4.2M	0	0%
	Cologne (BSM)	1M	0	0%

We observe a significant difference in the number of ambiguous pseudonyms between the real-world dataset and the synthetic datasets, where the synthetic datasets contain no ambiguities. We could infer that the ambiguous pseudonyms after performing trajectory reconstruction with path history were caused by inaccurate or faulty sensors in the real-world data collection and the difference in temporal units between the core message and the path history point. The first observation is concluded from the problem in the public dataset highlighted in Section 4.1, where the THEA dataset contains duplicate datapoints. On the other hand, in the synthetic datasets, there are no duplicate datapoints and the location in the message is accurately measured; therefore, the path history points can be accurately matched to the vehicle's location in the past. Based on the differences in the temporal units observation, since the messages in the synthetic dataset are generated at 1 Hz, the rate of message generation nullifies the effect of the differences in temporal units between the core message and the path history points. The nullified effect

from the difference in temporal units causes the range-matching in synthetic datasets to have the same results as the exact match approach. On the other hand, since the THEA dataset contains the effect of differences in temporal units, the number of ambiguous pseudonyms increases in the range-matching approach when compared with the exact-matching approach.

The differences in the number of ambiguous pseudonyms between the real-world dataset and the synthetic datasets highlight the adversary's challenge to use the trajectory reconstruction with path history in the real-world application. The adversary has to decide between accepting the ambiguity in the trajectory reconstruction results or discarding 20–30% of the pseudonyms in order to achieve accurate reconstructed trajectories. Nevertheless, it is important to evaluate the results of the reconstructed trajectory before deciding whether the path history point can be used effectively for trajectory reconstruction; i.e., how much privacy leakage can be caused by the path history points. Before moving on to the evaluation of the trajectory reconstruction models, we present another evaluation of the ambiguous pseudonyms that occur in the local data association in both Bayesian and Kalman filter trajectory reconstruction approaches.

6.1.2 Ambiguous Trajectories in Local Data Association

Similar to the ambiguity in the path history trajectory reconstruction approach, the ambiguous trajectory in the local data association approach occurs when two target pseudonyms have the same candidate pseudonym. We described the potential ambiguity as the local data association's limitation along with the explanation of Equation (5.14). It is worth noting that this issue can occur on both the Bayesian and Kalman filter approaches with local data association.

Table 6.2 shows the percentage of pseudonyms that appear as ambiguous from the local data association approach. We separately evaluate the number of ambiguous pseudonyms according to the trajectory reconstruction approach

(Bayesian or Kalman filter), the synthetic message types that were used during the evaluation (BSM or CAM), and the parameter for probability estimation (location-based or all-kinematics-based).

Table 6.2: Percentage of pseudonyms that appear on ambiguous trajectories in the local data association approach.

Models	Message Type	Kinemetics	Affected Trajectories	Ambiguous Pseudonyms	
				Number	%
Bayesian	BSM	Location	157148	71462	6.84
		All-kinematics	157288	71570	6.85
	CAM	Location	47293	23175	0.55
		All-kinematics	22252	22252	0.53
Kalman filter	BSM	Location	3946	1958	0.18
		All-kinematics	5642	2800	0.27
	CAM	Location	5756	2865	0.07
		All-kinematics	8370	4162	0.10

We could observe that, in general, the Kalman filter approach has fewer ambiguous pseudonyms compared with the Bayesian approach. Without considering correctness, we could assume that the Kalman filter approach has a better ability to distinguish between candidate pseudonyms than the Bayesian approach. A clear distinction between candidate pseudonyms in the Kalman filter arises from constantly updating the process noise covariance $\mathcal{Q}$. While the Bayesian approach uses the confidence value taken from the last message to calculate the probability between two pseudonyms (see Equations (5.8) to (5.13)), the Kalman filter approach has the process noise covariance to account for the confidence of the expected values. Since the process noise covariance is constantly updated with each received message, the confidence values in the Kalman filter approach become more accurate than those in the Bayesian approach. However, constantly updating the covariance will cause the Kalman

filter approach to go through all messages instead of using only the last message from the pseudonym, which creates processing overhead compared to the Bayesian approach.

An additional observation from Table 6.2 is that the number of affected trajectories is higher than twice the number of ambiguous pseudonyms, suggesting that there are ambiguous trajectories that comprise more than two ambiguous pseudonyms. This highlights the struggle to distinguish between candidate pseudonyms in both the Bayesian and Kalman filter approaches in a clutter scenario. Lastly, despite having more pseudonyms, the percentage of ambiguous pseudonyms is lower with CAM (ETSI standards) than with BSM (SAE standards).

6.2 Evaluations on Pseudonyms Linkage

Identifying the relation between pseudonyms is the first step to trajectory reconstructions and corresponds to the linkage privacy threats identified in our threat cards 3.8 – 3.10. From the adversary's perspective, the keys to successfully reconstructing trajectories are not only the correctness of pseudonyms linkage identification, but also the complete set of all the links between every pseudonym pair. Let's consider an example where, by constantly applying the pseudonym changing mechanism while driving, a vehicle uses three pseudonyms to hide its real identity in one trip. To exploit the threat 3.13 in this example, the adversary must be able to identify both links between the three pseudonyms in order to find the origin and destination of the trip. These elements for a successful trajectory reconstruction can be measured from the information retrieval perspective.

Let $\mathcal{L}$ be the set of all links between pseudonyms in the dataset, and $\mathcal{L}_r$ be the set of links that the adversary identifies. A set of links between pseudonyms identified by the adversary can be classified into two groups: links that exist in $\mathcal{L}$ and links that are not in $\mathcal{L}$. These groups of identified links correspond

to the true positives TP and false positives FP in the information retrieval concept, where

- TP represents the identified links that exists in $\mathcal{L}$; $TP = \mathcal{L} \cap \mathcal{L}_r$
- FP represents the identified links that are not in $\mathcal{L}$; $FP = \mathcal{L}_r - \mathcal{L}$

Additionally, there could be links in $\mathcal{L}$ that are discarded due to (1) its weak representation of the relationship between pseudonyms, or (2) there is a link with a stronger relationship. Recalling the inference process in both the Bayesian and Kalman filter models, the relationship between pseudonyms is represented with a probability, where a high probability indicates a strong relationship between pseudonyms, and a low probability indicates a weak relationship. Then, during the data association process, all of the probabilities are considered and are used to decide which pseudonyms should be linked together. In the first discarded case, we decided that the link with a probability lower than 0.05 should be discarded since the relationship between pseudonyms is insignificant. On the other hand, in the second case, the discarded links may occur when a target pseudonym has a better candidate to pair with. In other words, there is a link with a higher probability, which represents a stronger relationship with the target pseudonym. This process of discarding a link during the data association process is discussed in Equations (5.14) and (5.18). When the discarded link is a correct link between pseudonyms, it can be viewed as the false negative FN in the retrieval process; $FN = \mathcal{L} - TP$.

Given $|X|$ is the number of links in the set X, we can measure the precision, recall, and F1-score of the links retrieval process as follows:

$$Precision = \frac{|TP|}{|TP| + |FP|} \tag{6.1}$$

$$Recall = \frac{|TP|}{|TP| + |FN|} \tag{6.2}$$

$$F1 - score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{6.3}$$

The precision highlights the accuracy of the links identified by the adversary, i.e., the identified links are all correct if the precision is 1. While the adversary

can identify only one correct link between the vehicle and achieve the highest precision, the recall evaluates the portion of links between pseudonyms from $\mathcal{L}$ which the adversary identifies. Lastly, the F1-score combines both precision and recall into one metric to evaluate the accuracy and completeness of the identified links. A general interpretation of these metrics is that a higher value, maximum at 1, represents a higher success in trajectory reconstruction, which hinders the user's privacy. It must be noted that, since the correctness of identifying the links between pseudonyms requires a ground truth, the following evaluations in this section will exclude the public dataset and be solely based on the synthetic dataset generated in Section 4.2.

Tables 6.3 and 6.4 show the evaluations of precision, recall, and F1-score of links between pseudonyms from three trajectory reconstruction approaches separated by the standards for pseudonym-changing mechanisms. Additionally, in Bayesian and Kalman filter approaches, two data association models—local and global data associations—are presented in the tables. All three trajectory reconstruction models are implemented based on the process presented in Chapter 5 with the synthetic dataset presented in Section 4.2. It is worth noting that the evaluations of "location-based" trajectory reconstruction are equivalent to the approach taken in [8], where only locations are used in the probability estimation. On the other hand, the "all-kinematics" approach is our proposed methodology for trajectory reconstruction, where location, speed, and heading are used in the probability estimation. The bold text highlights a higher value when comparing the location-based and all-kinematics trajectory reconstruction approaches.

Overall, the evaluation shows that all trajectory reconstruction approaches, except for the local data association with the Bayesian approach applied to the BSM dataset, achieve F1-scores above 90%. Such consistently high performance highlights a significant privacy risk in VANET communications, as it indicates that adversaries are highly capable of identifying the relation between pseudonyms regardless of the method used. In particular, the path

Table 6.3: Precision, Recall, and F1-score of retrieved links between pseudonyms on Basic Safety Messages (SAE standard).

Models	Data Assoc.	Kinemetics	Precision	Recall	F1-score
Path History			0.9999	0.9293	0.9633
Bayesian	Local	Location	0.6966	**0.9999**	0.8211
		All-kinematics	**0.6997**	0.9988	**0.8230**
	Global	Location	0.9341	0.9968	0.9644
		All-kinematics	**0.9441**	**0.9979**	**0.9702**
Kalman filter	Local	Location	**0.9912**	**0.9947**	**0.9929**
		All-kinematics	0.9857	0.9829	0.9843
	Global	Location	**0.9959**	**0.9950**	**0.9955**
		All-kinematics	0.9924	0.9835	0.9879

history-based reconstruction, which represents a unique privacy threat specific to VANETs, demonstrates comparable precision, recall, and F1-scores to the other approaches. This result highlights the potential for adversaries to exploit path history points with the same effectiveness as with the other trajectory reconstruction models. When comparing location-based and all-kinematics approaches, the results are nearly indistinguishable, suggesting that the usage of additional kinematic information in the probability estimation process does not necessarily improve the identification of relationships between pseudonyms. Finally, a cross-comparison of results across different standards (SAE & ETSI) shows only borderline differences in performance, indicating no differences in the pseudonym-changing mechanisms when it comes to avoiding the linkage between pseudonyms.

We can observe a trend in the Bayesian and Kalman filter models, where all location-based trajectory reconstruction approaches are slightly better than all-kinematics approaches in the Kalman filter model. On the other hand, in the Bayesian model, eight out of twelve metrics favour the all-kinematics ap-

Table 6.4: Precision, Recall, and F1-score of retrieved links between pseudonyms on Cooperative Awareness Messages (ETSI standard).

Models	Data Assoc.	Kinemetics	Precision	Recall	F1-score
Path History			0.9999	0.9362	0.9670
Bayesian	Local	Location	0.9545	**0.9991**	0.9763
		All-kinematics	**0.9578**	0.9969	**0.9770**
	Global	Location	0.9978	**0.9993**	**0.9985**
		All-kinematics	**0.9981**	0.9941	0.9961
Kalman filter	Local	Location	**0.9937**	**0.9923**	**0.9930**
		All-kinematics	0.9923	0.9752	0.9837
	Global	Location	**0.9984**	**0.9931**	**0.9957**
		All-kinematics	0.9980	0.9762	0.9870

proach more than having only location for trajectory reconstruction. This subtle difference between the Bayesian and Kalman filter models highlights the effects of having more information in the trajectory reconstruction models. Since the probability estimation in the Bayesian model is based on separate probability estimations for each kinematic (Equations (5.8)–(5.11)) before combining into a single probability (Equation (5.13)), the changes in one kinematic have only a small effect on the final probability. On the contrary, the Kalman filter model relies on the state transition where a change in one kinematics has more effect on the other (Equation (5.22)) when compared with the Bayesian model. This difference between the Bayesian and Kalman filter models results in a different rate of change in the probability between the target and candidate pseudonyms when more information is added to the models. As a result, in the Kalman filter model, the precisions with all-kinematics tend to be lower than those with the location-based approach, since more information creates higher confusion and makes it indistinguishable between the correct and incorrect candidate pseudonyms. It is also worth noting that hav-

ing more information, an all-kinematics approach, will also likely decrease the probability between the target and the candidate pseudonyms and result in a smaller number of links between pseudonyms being identified. Therefore, the recall rates with the all-kinematics approach tend to be lower than those in the location-based approach.

Another observation is the significant difference in the precision rates between the SAE and ETSI standards on the Bayesian model with a local data association approach. We can observe that the precision rates in the SAE standard (Table 6.3) are lower than 70%, which makes it 25% lower than those in the ETSI standard in Table 6.4. To explain this behaviour, we must first examine the recall rates in the Bayesian model with the local data association model in the SAE standard, where these recall rates are among the highest compared to other methods. The higher recall rate indicates that the model is able to account for most of the links between pseudonyms. It also implies that the model is less likely to discard the incorrect links between pseudonyms; therefore, creating a larger number of false positives. Secondly, the vehicles in the SAE standard use fewer pseudonyms per trip when compared with the ETSI standard. As a result, there are fewer links between pseudonyms in the SAE standard when compared with the ETSI standard. When combining these two observations, the Bayesian model with local data association in the SAE standard produces a higher ratio of false positives to the total number of links between pseudonyms than those in the ETSI standard. Therefore, this higher ratio between incorrect links and the total number of links has compromised the precision rates in the SAE standard, resulting in a significant difference compared to the ETSI standard.

In addition to the adversary's capability to link two pseudonyms, we can also evaluate their confidence in the linking process. Both the Bayesian inference model and the Kalman filter not only identify pseudonym pairs but also assign a probability indicating the likelihood that the two pseudonyms belong to the same vehicle. These probabilities can be used to compute "nor-

malised entropy", which quantifies the level of uncertainty in the adversary's inference. A higher normalised entropy indicates greater uncertainty in the linking decision, while a lower value suggests more confidence in linking the pseudonyms together. It is important to note that normalised entropy measures the confidence level of the adversary, independent of the actual correctness of the linkage. Thus, a high entropy could still lead to an incorrect link between pseudonyms, just as a low entropy could correspond to correct ones. Given x is the target pseudonym that the adversary is trying to find another pseudonym to link with, and $CPID$ is a set of candidate pseudonyms for the target pseudonym, the normalised entropy is calculated using the full set of probabilities between the target pseudonym and a candidate pseudonym and the maximum achievable entropy, as shown below:

$$Normalised\ Entropy\ (NH) = \frac{-\sum_{j \in CPID} Pr(x,j) \log\left(Pr(x,j)\right)}{Maximum\ Entropy} \tag{6.4}$$

$$Maximum\ Entropy = \log\left(|CPID|\right) \tag{6.5}$$

where $|CPID|$ is the total number of candidate pseudonyms that can be linked with the target pseudonym.

Table 6.5 shows the average normalised entropy (NH), adversary's uncertainty, in the pseudonym linking process. We separately calculate the normalised entropy according to the trajectory reconstruction model, the message type according to the standard for communication, the data association approach, and the variable for probability estimation. It is worth noting that a high normalised entropy indicates a high uncertainty during the pseudonyms linkage process (data association sub-model). Therefore, a model with lower average normalised entropy would be a preferred choice for the adversary, as it represents a higher confidence in the pseudonym linkage process. Also, since the path history-based trajectory reconstruction approach does not provide a probability between the pseudonym pair, it is excluded from this evaluation.

Overall, the average normalised entropy values are very low across all ap-

Table 6.5: Average normalised entropy in the pseudonyms linking process.

Models	Message Type	Data Assoc.	Kinematics	Average NH
Bayesian	BSM	Local	Location	0.0270
			All-kinematics	0.0087
		Global	Location	0.0483
			All-kinematics	0.0087
	CAM	Local	Location	0.0063
			All-kinematics	0.0016
		Global	Location	0.0199
			All-kinematics	0.0037
Kalman filter	BSM	Local	Location	0.0703
			All-kinematics	0.0395
		Global	Location	0.0596
			All-kinematics	0.0307
	CAM	Local	Location	0.0337
			All-kinematics	0.0157
		Global	Location	0.0285
			All-kinematics	0.0116

proaches, considering that the range of normalised entropy is between 0 and 1. A low normalised entropy indicates that the adversary has high confidence in associating the candidate pseudonym with the target pseudonym, regardless of the correctness of the outcome. It also indicates a significant distinction between the best candidate pseudonym and the other candidates. It is worth noting that the Bayesian model consistently produces lower normalised entropy than the Kalman filter model. This indicates that there is higher distinguishability in the probability in the Bayesian model compared to the Kalman filter model. This outcome aligns with our observation from the last evaluation, where the Kalman filter model has greater sensitivity to

changes in kinematics than the Bayesian model. As a result, the Kalman filter approach produces a less distinguishable link between pseudonyms and results in a higher normalised entropy than that in the Bayesian model. The distinguishability in the Kalman filter can be improved through the global data association approach, which reduces its entropy values by reassigning probabilities after considering the weights across all pseudonym links. On the other hand, the normalised entropy in Bayesian's global data association becomes higher than its local data association's normalised entropy. Since the global data association approach considers the probability of all links, in the case of the Bayesian model, it tends to smooth the distribution and reduce the difference between candidates, resulting in a more uniform probability distribution and, consequently, higher entropy.

Another observation is that the normalised entropy values for the location-based approaches are generally higher than those for the all-kinematics approaches. The lower entropy in the all-kinematics setting is expected, as incorporating additional kinematic information, such as speed and heading, introduces more constraints into the probability estimation, which increases the distinction of the probability distribution and increases the adversary's confidence when selecting a candidate pseudonym. At first glance, this observation appears inconsistent with our previous evaluation, where adding more information can increase indistinguishability between the pairs of pseudonyms. However, normalised entropy reflects the adversary's confidence in associating the pseudonyms pair, not the correctness of the chosen link. Therefore, using more kinematics information can lead the adversary to form a more confident, yet not necessarily correct, association between pseudonyms.

6.3 Evaluations on Reconstructed Trajectory

Although the previously discussed success rate of pseudonym linkage has demonstrated the adversary's ability to identify pseudonym pairs from the same ve-

hicle, the relationship between two pseudonyms may not fully capture the privacy threats in VANET communication. Once pseudonyms have been linked into pairs, without considering the correctness of the link, an adversary can aggregate these pairs into sets representing individual reconstructed trajectories. Each set of pseudonyms outlines a vehicle's path over time, enabling the adversary to infer the starting point from the first location of the earliest pseudonym and the destination from the final location of the last pseudonym. As highlighted in Threat Card 3.12, this origin-destination pair can serve as a quasi-identifier for the trip. Moreover, if the trajectory reconstruction is fully corrected, the resulting path may reveal sensitive locations visited by the driver. These frequently include personal points of interest such as the driver's home or workplace. This form of privacy leakage aligns with the concern outlined in Threat Card 3.13, where identifying home and work locations can ultimately lead to the deanonymisation of the driver.

Another significant privacy threat arises when an adversary is able to deanonymise a single pseudonym within a set of linked pseudonyms. This deanonymisation may occur through the reuse of a previously identified pseudonym or access by an insider adversary, as described in Threat Card 3.11. Once one pseudonym in the reconstructed trajectory is deanonymised, it can compromise the anonymity of the entire pseudonym set. This linkage enables the adversary to infer additional sensitive information, such as unexpected start and stop locations, even when these locations do not correspond to the driver's home or workplace. When combined with a fully reconstructed trajectory, this threat further undermines location privacy and increases the risk of user deanonymisation.

Given the identified privacy threats, it becomes clear that a fully reconstructed trajectory significantly increases the risk of privacy breaches. Conversely, user privacy in VANETs relies heavily on the adversary's inability to accurately reconstruct vehicle trajectories. Even a single missing or incorrect pseudonym linkage can lead to inaccurate interpretations of a vehicle's path,

including its origin and destination location of the trip. Our evaluation of the reconstructed trajectory begins by assessing each trajectory reconstruction model's capability to correctly identify trip origins and destinations. While origin and destination identification capture only partial information about the overall trajectory, the correct identification of these locations may lead to a deanonymisation of a pseudonym. Consequently, as previously discussed, a single deanonymised pseudonym may lead to the deanonymisation of a larger number of pseudonyms.

We continue to use the same metrics as the previous subsection, precision, recall, and F1-score, to highlight the adversary's ability to retrieve the trip's origin and destination from the dataset of the messages. In this evaluation, a pseudonym without a preceding pseudonym is considered the first pseudonym of the trajectory; consequently, the first message broadcast by this pseudonym is considered the beginning of a trip. Conversely, a pseudonym without a succeeding pseudonym is accounted as the last pseudonym, and the last message of this pseudonym is considered the trip's destination.

Given a set of all trajectories $\mathcal{T}$ in the ground truth, a set $\mathcal{O}$ represents a set of origins or destinations of these trajectories. A set of the adversary's reconstructed trajectories is represented as $\mathcal{T}_r$; consequently, a set $\mathcal{O}_r$ represents a set of trips' origins or destinations identified by the adversary. Lastly, a set $\mathcal{T}_r'$ represents a set of the adversary's reconstructed trajectories where all pseudonyms in the set belong to the same vehicle, and a set $\mathcal{O}_r'$ contains the origins or destinations extracted from these trajectories. Figure 6.2 visualises the relations between the sets $\mathcal{O}$, $\mathcal{O}_r$, and $\mathcal{O}_r'$ as well as the TP, FP, and FN classes of origins or destinations.

According to the classification illustrated in Figure 6.2, the true positive TP, false positive FP, and false negative FN can be explained, regardless of a fully reconstructed trip or a partially reconstructed trip, as follows:

- TP represents the identified origins or destinations from the correctly reconstructed trajectories $\mathcal{O}_r'$ that exist in $\mathcal{O}$; $TP = \mathcal{O} \cap \mathcal{O}_r'$

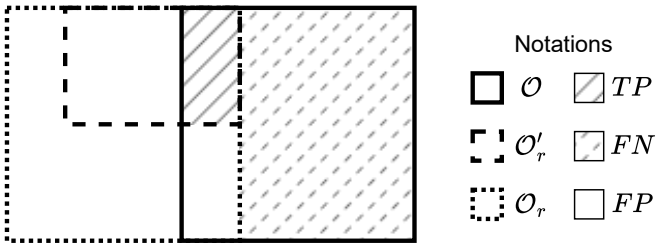


Figure 6.2: Visualisation of the origins (or destinations) classification.

- FP represents the identified origins or destinations from the reconstructed trajectories excluding the set of TP; $FP = \mathcal{O}_r - TP$
- FN represents the unidentified origins or destinations from $\mathcal{O}$ including those in the incorrectly reconstructed trajectories; $FN = \mathcal{O} - \mathcal{O}_r$

It must be noted that the set of correctly identified origins or destinations with an incorrectly reconstructed trajectory, $(\mathcal{O}_r - \mathcal{O}'_r) \cap \mathcal{O}$, is also classified as a false positive. Even though these locations belong to the correct set of origins or destinations, they are classified as false positives because the incorrect pseudonym in the set may result in a misinterpretation of the user's identity. Let us consider these cases from an example of reconstructed trajectories illustrated in Figure 6.3. A subscripted capital letter represents a vehicle, and the number that follows such a letter represents the pseudonym used in the vehicle's trip. For instance, the vehicle A used two pseudonyms throughout its trip, pid_{A1} and pid_{A2}. The line of each circle represents whether three groups of pseudonyms: the first pseudonym (solid line), the intermediate pseudonym (dotted line), and the last pseudonym (dashed line). As a consequence, the circle with a solid line contains the trip's origin, and the circle with a dashed line contains the trip's destination. A line between pseudonyms represents the link that the adversary associates during the data association process, where a solid line represents a link between the pseudonyms from the same vehicle, and a dashed line represents a link between pseudonyms from different vehicles.

In this example, the adversary is able to reconstruct three trajectories. However, the first trajectory consists of two trips from different vehicles, where

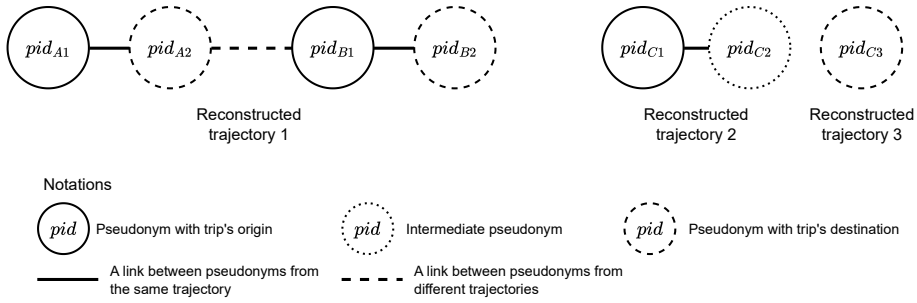


Figure 6.3: Example of reconstructed trajectories.

the second and the third trips are partial trajectories of the trip made by vehicle C. When the trip's origin identification is evaluated, these reconstructed trajectories will result in 1 TP from considered pid_{C1} as trip's origin, 2 FP from considered pid_{A1} and pid_{C3} as trip's origin, and 1 FN from missing pid_{B1} as trip's origin. Considering pid_{A1} as a false positive represents our interpretation to avoid misinterpretation by the adversary. An example of the misinterpretation is that the adversary correlates the incorrect pseudonym with the user's identity that belongs to the correctly identified trip's origin. From Figure 6.3, the adversary may correlate pid_{B1} and pid_{B2} with the user's identity that is associated with the first location in pid_{A1}. When these incorrect pseudonyms, pid_{B1} and pid_{B2}, are reused by their true owner—vehicle B—in a later stage, the adversary will incorrectly interpret that the user with vehicle A is the person who traversed in this trip. Additionally, a correct-but-incomplete reconstructed trajectory with a correct origin/destination will be considered a true positive, e.g., pid_{C1} for the trip's origin identification. Therefore, a pseudonym that has not been linked with any preceding pseudonym will be considered a true positive for origin identification if it is the first pseudonym in the trajectory, even if it is not linked with the rest of its trajectory. Similarly, a pseudonym that does not have a successful link will be considered a true positive for destination identification if it is the last pseudonym in the trajectory, even if it is not linked with the rest of its trajectory.

Tables 6.6 to 6.9 show evaluations of precision, recall, and F1-score of the trip's origin and destination identified by the adversary. The tables are sep-

arated by the message type corresponding to the pseudonym-changing mechanisms and the evaluation on either origin (Tables 6.6 to 6.7) or destination (Tables 6.8 to 6.9) identification. Each table presents the evaluation according to the trajectory reconstruction approach, data association method, and the kinematics for probability estimation.

Table 6.6: Precision, Recall, and F1-score of retrieved trip's origin on Basic Safety Messages (SAE standard).

Models	Data Assoc.	Kinemetics	Precision	Recall	F1-score
Path History			0.9189	0.9999	0.9577
Bayesian	Local	Location	0.6667	0.3049	0.4184
		All-kinematics	**0.6668**	**0.3116**	**0.4247**
	Global	Location	0.9144	0.8380	0.8745
		All-kinematics	**0.9265**	**0.8606**	**0.8924**
Kalman filter	Local	Location	**0.9827**	**0.9784**	**0.9805**
		All-kinematics	0.9620	0.9655	0.9637
	Global	Location	**0.9888**	**0.9867**	**0.9878**
		All-kinematics	0.9706	0.9798	0.9752

In general, most trajectory reconstruction approaches perform well with precision and recall over 80% of the trip's origin, except for the Bayesian's local data association. A low recall rate in the Bayesian approach with local data association indicates that only a few locations are identified as the trip's origin. The underlying issue arises when the model incorrectly links pseudonyms across different trajectories. For instance, the last pseudonym in a trajectory is linked with another pseudonym, which is the first pseudonym from a different trajectory, as depicted with reconstructed trajectory 1 in Figure 6.3. While those trajectories are correctly reconstructed trajectories, the additional link renders the combined trajectory into a false positive, as the reconstructed trajectory is made up of pseudonyms from different vehicles, and simultaneously

Table 6.7: Precision, Recall, and F1-score of retrieved trip's origin on Cooperative Awareness Messages (ETSI standard).

Models	Data Assoc.	Kinemetics	Precision	Recall	F1-score
Path History			0.8159	0.9999	0.8986
Bayesian	Local	Location	0.7372	0.4598	0.5664
		All-kinematics	**0.7416**	**0.4976**	**0.5956**
	Global	Location	**0.9822**	0.9705	**0.9763**
		All-kinematics	0.9472	**0.9779**	0.9623
Kalman filter	Local	Location	**0.8993**	**0.9092**	**0.9042**
		All-kinematics	0.7897	0.8992	0.8409
	Global	Location	**0.9397**	**0.9789**	**0.9589**
		All-kinematics	0.8307	0.9749	0.8971

creates two false negatives, as two origins will be missing from the classification results. This underperformance when compared with other approaches highlights a weakness in trajectory reconstructions, where the Bayesian approach with the local data association struggles with under-identification due to mislinked pseudonyms across trips.

When comparing the evaluations between location-based and all-kinematics probability estimation, there are only small differences between these two methods. The small differences in precision, recall, and F1-score suggest that incorporating additional information into probability estimation may not always provide a significant advantage in estimating the relationship between pseudonyms. To the contrary, employing the global data association to readjust the probability provides a higher precision, recall, and F1-score across all of the trajectory reconstruction approaches. A large improvement is observed in the Bayesian trajectory reconstruction model, where the global data association increases the recall rates on both location-based and all-kinematics approaches.

For the path history approach, the model performs as well as other approaches to trajectory reconstruction. From the evaluations of ambiguous trajectories in Tables 6.1 and 6.2, the results of reconstructed trajectories from the path history approach do not contain any of the ambiguous trajectories. On the other hand, both Bayesian and Kalman filter approaches with local data association produce up to 6.8% of ambiguous trajectories compared to the total reconstructed trajectories. As a result, these ambiguous trajectories contributed to the false positive identification of the trip's origin; therefore, making the precision, recall, and F1-score of the local data association-based trajectory reconstructions lower than the path history approach. It is also important to note that the Kalman filter approach with local data association through location-based probability estimation achieves the lowest rate of ambiguous trajectories, making origin identification as good as the path history trajectory reconstruction model. Additionally, since the global data association method solves the ambiguous trajectory problem, the precision, recall, and F1-score with the global data association are as high as those of the path history approach.

The evaluations of precision, recall, and F1-score on destination identification show similar trends to the evaluations in origin identification. It is expected that the evaluations of the destination and the origin identifications will have similar results when compared with each other. To explain the similarity between origin and destination identification, we should consider that a reconstructed trajectory can be (1) correctly and fully reconstructed, (2) correctly but fragmented reconstructed, or (3) incorrectly reconstructed (having a mix of pseudonyms from different vehicles). Recalling Figure 6.3, the second type of reconstructed trajectory is depicted with the reconstructed trajectories 2 and 3, and the third type of reconstructed trajectory is depicted with the reconstructed trajectory 1. The correctly and fully reconstructed trajectory provides a true positive for both origin and destination identification. The second type of reconstructed trajectory is a correctly but fragmented recon-

Table 6.8: Precision, Recall, and F1-score of retrieved trip's destination on Basic Safety Messages (SAE standard).

Models	Data Assoc.	Kinemetics	Precision	Recall	F1-score
Path History			0.9190	0.9999	0.9578
Bayesian	Local	Location	0.6668	0.3050	0.4185
		All-kinematics	**0.6678**	**0.3120**	**0.4253**
	Global	Location	0.9113	0.8351	0.8716
		All-kinematics	**0.9260**	**0.8601**	**0.8918**
Kalman filter	Local	Location	**0.9843**	**0.9800**	**0.9821**
		All-kinematics	0.9665	0.9699	0.9682
	Global	Location	**0.9897**	**0.9876**	**0.9886**
		All-kinematics	0.9731	0.9823	0.9777

structed trajectory, where one or more links between pseudonyms are missing to create the fully and correctly reconstructed trajectory. This type of reconstructed trajectory simultaneously creates a true positive and a false positive on one of the origin or destination identification, or a false positive on both. Consider a reconstructed trajectory that is fragmented into three parts due to two missing links between pseudonyms. While the first fragmented trajectory correctly identifies the trip's origin, it leads to a false identification of the trip's destination. Similarly, the last fragment correctly identified the trip's destination while simultaneously producing a false origin identification. Lastly, the intermediate fragment creates false positives for both origin and destination identifications. As a result, both origin and destination identification will receive +1 true positive and +2 false positives at the same rate from these fragmented trajectories.

The small differences observed between origin and destination identification largely come from the third type of outcome from the trajectory reconstruction, where the reconstructed trajectory is mixed with pseudonyms from

Table 6.9: Precision, Recall, and F1-score of retrieved trip's destination on Cooperative Awareness Messages (ETSI standard).

Models	Data Assoc.	Kinemetics	Precision	Recall	F1-score
Path History			0.8159	0.9999	0.8986
Bayesian	Local	Location	0.7409	0.4622	0.5693
		All-kinematics	**0.7504**	**0.5036**	**0.6027**
	Global	Location	**0.9805**	0.9688	**0.9746**
		All-kinematics	0.9468	**0.9775**	0.9619
Kalman filter	Local	Location	**0.9091**	**0.9192**	**0.9141**
		All-kinematics	0.8034	0.9148	0.8555
	Global	Location	**0.9395**	**0.9788**	**0.9587**
		All-kinematics	0.8313	0.9757	0.8977

different vehicles. Consider an example with two reconstructed trajectories. The first reconstructed trajectory is a fragment that contains a trip's origin, contributing +1 true positive for origin identification, but with an invalid destination; the fragment adds +1 false positive to the destination identification. The second fragment contains the trip's destination, but it is incorrectly linked with an additional pseudonym from another vehicle. As a result, the second fragment contributes +1 false positive for both origin and destination identification. Furthermore, as the trip's origin and destination will be missing from the second fragment, it also contributes +1 false negative on both origin and destination identification. By combining the identification results, the origin identification receives +1 true positive, false positive, and false negative. On the other hand, the destination identification receives +2 false positives and +1 false negative, and suffers more in precision than the origin identification. It is worth noting that the incorrectly reconstructed trajectory can also occur at the first fragment, resulting in a reverse imbalance from the example.

To further understand the extent of privacy risks in VANETs, we now evalu-

ate the adversary's ability to fully reconstruct vehicle trajectories. A complete trajectory reconstruction represents one of the most severe privacy threats, as it implies that the adversary has successfully linked all relevant pseudonym pairs associated with a trip. This capability not only allows the adversary to determine the vehicle's origin and destination but also to infer the precise path taken throughout the journey. Such detailed reconstruction can lead to highly sensitive inferences about user behaviour and personal locations. Accordingly, this evaluation focuses on measuring how accurately each proposed model can reconstruct entire vehicle trajectories, which directly reflects the level of privacy leakage in VANET communications.

To evaluate the adversary's ability in trajectory reconstruction, a definition of a complete reconstructed trajectory must be provided. We consider a reconstructed trajectory to be complete if all of the pseudonyms used in the trajectory are presented in the same set of pseudonyms. Moreover, a fully reconstructed trajectory with an additional pseudonym at the beginning or the end of the trip will not be considered a complete reconstructed trajectory. The additional pseudonym at the beginning of the fully reconstructed trajectory can occur when the first pseudonym of the trajectory is the best candidate for another pseudonym that does not belong to this trip. Similarly, the additional pseudonym at the end may occur if the last pseudonym of the trajectory is able to find a suitable candidate pseudonym that does not belong to this trajectory. These additional pseudonyms can cause misinterpretations of the trip's origin and destination; hence, the presence of these pseudonyms nullifies the completeness of the trajectory.

Following the previous metrics, we continue to use precision, recall, and F1-score to evaluate the reconstructed trajectories. A reconstructed trajectory will be considered a true positive if and only if it is a complete reconstructed trajectory. Trajectories that are not considered as a completely reconstructed trajectory, including a single pseudonym that is not linked with its trajectory, will be considered as false positive trajectories. Lastly, the complete

trajectories that the trajectory reconstruction models have not identified will be considered as false negative trajectories. Tables 6.10 and 6.11 show the precision, recall, and F1-score evaluations of the complete trajectory reconstruction by three proposed models: path history-based, Bayesian approach, and Kalman filter approach. The tables are separated by the message types, BSM or CAM, which are the results of employing different VANET standards.

Table 6.10: Precision, Recall, and F1-score of complete reconstructed trajectories on Basic Safety Messages (SAE standard).

Models	Data Assoc.	Kinemetics	Precision	Recall	F1-score
Path History			0.8417	0.9159	0.8772
Bayesian	Local	Location	0.6667	0.3049	0.4184
		All-kinematics	**0.6668**	**0.3116**	**0.4247**
	Global	Location	0.9111	0.8350	0.8714
		All-kinematics	**0.9246**	**0.8588**	**0.8905**
Kalman filter	Local	Location	**0.9776**	**0.9733**	**0.9755**
		All-kinematics	0.9452	0.9486	0.9469
	Global	Location	**0.9835**	**0.9814**	**0.9825**
		All-kinematics	0.9528	0.9618	0.9573

Over half of the evaluations achieve scores above 85% on at least one of precision, recall, or F1, with most results below 85% belonging to the path history-based and local data association methods. For the path history approach, while its precision may not be as good as that of the global data association in the other method, it consistently achieves high recall, particularly in the BSM case. The drop in precision can be described by fragmented trajectories, where missing links create incomplete reconstructions that increase false positives. Nevertheless, the strong results in origin and destination identifications indicate that the path history method effectively captures correct pseudonym linkages, even if it struggles to reconstruct them into a complete

Table 6.11: Precision, Recall, and F1-score of complete reconstructed trajectories on Cooperative Awareness Messages (ETSI standard).

Models	Data Assoc.	Kinemetics	Precision	Recall	F1-score
Path History			0.6581	0.8066	0.7249
Bayesian	Local	Location	**0.7372**	0.4598	0.5664
		All-kinematics	0.7337	**0.4923**	**0.5892**
	Global	Location	**0.9802**	**0.9684**	**0.9743**
		All-kinematics	0.9092	0.9387	0.9237
Kalman filter	Local	Location	**0.8537**	**0.8631**	**0.8584**
		All-kinematics	0.6525	0.7429	0.6948
	Global	Location	**0.8918**	**0.9290**	**0.9100**
		All-kinematics	0.6875	0.8068	0.7424

trajectory.

In the local data association approach, as previously discussed in origin identification, the problem of ambiguous trajectories, where pseudonyms are assigned to more than one trajectory, causes the models' performance to suffer from a high false negative rate. This issue becomes more evident when evaluating complete trip reconstruction, since stricter conditions are used to classify a trajectory as completely reconstructed. Consequently, local data association shows recall rates below 50%, meaning fewer than half of the trips are fully recovered. Also, recalling from Table 6.2, the Bayesian approach under local data association produces a larger number of ambiguous trajectories, resulting in a lower recall rate of complete trajectories when compared to the Kalman filter approach. This drawback presents a potential area for improvement in the global data association method within the Bayesian model compared to the Kalman filter model. As a result, the different rates of improvement with the global data association method can be observed between the Bayesian and Kalman filter approaches, where the Bayesian models receive a signifi-

cant improvement, more than 40% in recall rate, over those in the Kalman filter approach, less than 7% in recall rate. Nevertheless, the global association consistently improves precision and recall across all pseudonym-changing standards, highlighting its effectiveness in mitigating ambiguities inherent in local association.

It is also worth noting the differences in precision, recall, and F1-score between using location-based and all kinematics probability estimation, where most of the evaluations show that the location-based approach slightly outperforms the all kinematics approach. The better results with location-based probability estimation suggest that including kinematics such as speed and heading may not be helpful in the trajectory reconstruction process. While we hypothesise that having more information in the probability estimation will produce a higher distinguishable pseudonyms' relations, it might not be the case in the road network where vehicles travel at the same speed and in the same direction. A significant performance gap is also observed under the Kalman filter trajectory reconstruction in the CAM scenario, where the all-kinematics approach has a 10% lower recall rate and a 20% lower precision rate compared to those with location-based probability estimation. Despite the improvement in global data association, the initially low precision and recall in this scenario render the lowest performance among all trajectory tracking approaches using the Kalman filter. Upon closer examination by grouping the reconstructed trajectories based on the true identity of the vehicle, we found that the low precision and recall in such a scenario are the result of fragmented trajectories, where one or more links are missing from the completely reconstructed trajectory. This finding aligns with the evaluations of origin and destination identifications, where the Kalman filter approach in the CAM scenario performs well in recalling the trip's origin and destination. In other words, the model is able to link pseudonyms together correctly but cannot always form a continuous, fully reconstructed path.

To this end, to highlight the severity of the privacy threats in VANET com-

munication, we classify the reconstructed trajectories into five categories: (1) completely reconstructed, (2) correctly reconstructed with the trip's origin, (3) correctly reconstructed with the trip's destination, (4) correctly reconstructed without the trip's origin and destination, and (5) mixed pseudonyms. The first group of reconstructed trajectories corresponds to the previously evaluated on completely reconstructed trajectories, and represents the strongest privacy threat among the types of reconstructed trajectories. The second and third groups reflect the evaluation of origin and destination evaluations, where the model correctly identifies the trip's origin or destination. Unlike the earlier origin/destination evaluations, here we only count trajectories that correctly identify one location but fail on another. These types of reconstructed trajectories resemble the fragment reconstructed trajectory, whether it is the first or last fragment of the full trajectory. On the other hand, the fourth group in our classification resembles the middle fragment of the trajectory, where neither the origin nor the destination of the fragment belongs to the correct group of the trip's origin and destination. Lastly, the fifth group of reconstructed trajectories highlights the incorrectly reconstructed trajectories; hence, a higher percentage of trajectories in this group indicates the adversary's inability to reconstruct a vehicle trajectory.

The implications for these groups of reconstructed trajectories are that a high percentage of the (1) completely reconstructed trajectories highlights the severe privacy threat in VANET communication. Although the types (2) and (3) of reconstructed trajectory, where the adversary is able to identify the trip's origin or destination correctly, do not pose a privacy threat as the (1) type, they indicate some level of privacy threat, as these locations may tie to the user's home or workplace addresses. Then, a higher percentage of the intermediate fragment reconstructed trajectories (4) and the mixed pseudonym type (5) imply higher privacy preservation, as the adversary may incorrectly interpret the origin and destination on these reconstructed trajectories. Tables 6.12 and 6.13 show the percentage of reconstructed trajectories classified into one

of the five groups we have discussed.

Table 6.12: Percentage of reconstructed trajectories on Basic Safety Messages (SAE standard).

Models	Data Assoc.	Kinemetics	Percentage of Trajectory in the Group				
			(1)	(2)	(3)	(4)	(5)
Path History			84.18	7.72	7.72	0.38	0.00
Bayesian	Local	Location	66.67	0.00	0.01	0.00	33.32
		All-kinematics	66.68	0.00	0.10	0.00	33.22
	Global	Location	91.11	0.33	0.02	0.00	8.54
		All-kinematics	92.46	0.19	0.14	0.00	7.21
Kalman filter	Local	Location	97.76	0.72	0.67	0.00	0.85
		All-kinematics	94.52	2.00	2.13	0.00	1.35
	Global	Location	98.35	0.53	0.61	0.00	0.50
		All-kinematics	95.28	1.78	2.03	0.00	0.91

Overall, the combination of completely reconstructed trajectories (1) and correctly reconstructed trajectories with identified origins (3) or destinations (3) accounts for over 90% of the reconstructed trajectories in most trajectory reconstruction models, except for those employing local data association in the Bayesian trajectory reconstruction approach. The high proportion of reconstructed trajectories that can facilitate the deanonymisation process highlights the severity of privacy risks in VANET communication. Notably, the majority of these trajectories arise from trajectory reconstruction models that achieve at least 80% of the completely reconstructed trajectory, which provides the most significant advantage to an adversary. In addition, three approaches with a lower completely reconstructed trajectory rate (less than 80%) still achieve a high proportion of useful outcomes for the adversary by correctly recovering either the origins or destinations. These include the path history-based method in CAM and two Kalman filter approaches with all-kinematics probability estimation in CAM.

It is worth noting that the reconstructed trajectories with types (2), (3), and (4) are incomplete due to one or more missing links, which prevent the

Table 6.13: Percentage of reconstructed trajectories on Cooperative Awareness Messages (ETSI standard).

Models	Data Assoc.	Kinemetics	Percentage of Trajectory in the Group				
			(1)	(2)	(3)	(4)	(5)
Path History			65.82	15.77	15.77	2.63	0.00
Bayesian	Local	Location	73.72	0.00	0.37	0.00	25.90
		All-kinematics	73.37	0.79	1.67	0.01	24.16
	Global	Location	98.02	0.20	0.04	0.00	1.74
		All-kinematics	90.92	3.79	3.75	0.14	1.39
Kalman filter	Local	Location	85.37	7.80	5.55	0.29	1.00
		All-kinematics	65.25	16.66	15.09	1.78	1.22
	Global	Location	89.18	4.79	4.77	0.09	1.17
		All-kinematics	68.75	14.32	14.39	1.25	1.29

reconstruction of the complete trajectory. High percentages in these categories show that while the model identifies correct pseudonym pairs, it fails to recover all links. Another important observation is on the intermediate reconstructed trajectory, type (4), where its percentage is much lower than the reconstructed trajectory that includes either the trip's origin (2) or destination (3). The presence of an intermediate reconstructed trajectory indicates that two or more links between pseudonyms are missing, preventing a full reconstruction of the trajectory. In other words, at least one link is not identified to connect the intermediate reconstructed trajectory to the trip's origin, and at least one link is missing to connect it to the trip's destination. Since type (4) is rare (occurring in fewer than 3% of reconstructed trajectories), most incomplete reconstructions are missing only a single link, rather than multiple links. This observation strengthened our earlier finding in the complete reconstructed trajectories evaluations, where the Kalman filter performs poorly under all kinematics probability estimation in CAM due to the unidentified links to complete the trajectory reconstruction process.

A comparison of Tables 6.12 and 6.13, which report results from the same trajectory reconstruction model with different pseudonym-changing mecha-

nisms, shows that the amount of fragmented trajectories is generally lower under SAE standards (BSM scenario) than under ETSI standards (CAM scenario). As the vehicle uses fewer pseudonyms in a trip under the SAE standards compared to the ETSI standards, there are fewer links to be identified by the trajectory reconstruction model; therefore, it becomes easier to reconstruct the full trajectory under SAE standards. The evidence for this observation can be seen in the larger percentage of complete trips following SAE standards compared to ETSI standards, as observed in the path history and Kalman filter approach. Recalling the evaluations of linkage between pseudonyms in Tables 6.3 and 6.4, the difference in the recall rate between SAE and ETSI standards is less than 1%. However, since the vehicles with ETSI standards used more pseudonyms during the trip, the 1% difference becomes a large number of links between pseudonyms, which results in a large number of fragmented trajectories. In contrast, the Bayesian model does not shift towards more complete reconstructions but instead produces more mixed-pseudonym trajectories, suggesting an overlinkage problem under this approach.

6.4 Summary

In this chapter, we quantify the potential privacy loss in VANET communication by evaluating the adversary's ability to reconstruct vehicle trajectories. The chapter begins with assessments of ambiguous trajectories, which occur when two or more trajectories merge, creating a false impression that a vehicle appears in multiple locations simultaneously. In the path history approach, our synthetic dataset produces no ambiguous trajectories. In contrast, the Bayesian and Kalman filter models under local data association show a small number of ambiguous trajectories, with the Bayesian approach showing the highest percentage of ambiguous trajectories.

We then evaluate the trajectory reconstruction models through a series of evaluations, including pseudonym linkage, normalised entropy, trip ori-

gin and destination identification, and complete trajectory reconstruction. Across these evaluations, using all kinematics for probability estimation does not consistently improve performance compared to location-based estimation. For the model-wise evaluations, in most scenarios, the Kalman filter outperforms the Bayesian approach. While the Kalman filter may reconstruct fewer complete trajectories than the Bayesian model in CAM scenarios, it produces fewer mixed-pseudonym trajectories, indicating a better linkage between pseudonyms. On the other hand, the path history approach performs comparably to both Bayesian and Kalman filter models, suggesting it can serve as a simpler yet effective reconstruction method.

Finally, comparing across standards, we observe that vehicles under ETSI standards (CAM scenarios) are generally less prone to complete trajectory reconstruction than vehicles under SAE standards (BSM scenarios), largely due to the higher number of pseudonyms used per trip. Our evaluations highlight the privacy risks associated with VANET communication, which aligns with the privacy threats identified in Section 3.1, as well as the varying effectiveness of trajectory reconstruction methods under different conditions.

Chapter 7

Conclusion

7.1 Closing Remarks

In this thesis, we focused on the analysis of privacy leakage in Vehicular Ad hoc Networks (VANETs) communication, specifically in a setting where Basic Safety Messages (BSMs) or Cooperative Awareness Messages (CAMs) are used, and the pseudonym-changing mechanism is employed according to the SAE or ETSI standards.

The BSMs and CAMs enable a vehicle to exchange information with other vehicles, with the simplest form being the periodic broadcast of its location and kinematics to prevent road accidents. Such applications trade off the road user's privacy by periodically exposing their whereabouts on the road, and potentially exposing sensitive locations tied to the user's identity. To mitigate privacy risks, a pseudonym is used in place of the user's real identity, and a pseudonym-changing mechanism is employed to update this pseudonym periodically.

In such a scenario, we explored privacy threats in VANET communication by asking "What are the potential privacy threats in VANET standards?". By keeping VANET standards as a core focus, we leveraged the privacy threat elicitation framework, the LINDDUN framework, and identified 13 potential privacy threats in VANET communication. These privacy threats range from a

minor threat, "lack of awareness", to a severe privacy threat of "exposure of the user's identity once a trajectory is reconstructed". Noticing that the user's privacy depends on the adversary's ability to reconstruct the vehicle's trajectory, we quantified the severity of the threat by answering the questions: "How does the information in the messages affect the privacy of the users?" and "How effective or ineffective are the pseudonym-changing mechanisms in preventing the users from being tracked?". Three trajectory reconstruction models were proposed to evaluate the effect of having all vehicle kinematics available in the messages and the effectiveness of the pseudonym-changing mechanisms. These models include (1) the path history-based approach, which presents a unique method for VANET communication, (2) the Bayesian probability estimation, and (3) the Kalman filter approach.

Our series of evaluations at each step of the trajectory reconstruction process showed that all trajectory reconstruction models can identify the linkage between pseudonyms, with recall rates above 90%. Due to unidentified links and overlinkage between pseudonyms, the completely reconstructed trajectory rate ranged from 65% to 98%, with the percentage of the reconstructed trajectory varying from model to model. A comparison between standards from two major standard bodies revealed only a small difference between the ETSI and SAE standards in identifying the links between pseudonyms. However, due to the larger number of pseudonyms used in a trip, the percentage of completely reconstructed trajectories tends to be lower under the ETSI standard (CAM) than the SAE standard (BSM). When comparing the trajectory reconstruction models, we observed that the Bayesian model tends to overly link pseudonyms, resulting in a higher number of false positives than those in path history-based and Kalman filter models. On the other hand, path history-based and Kalman filter trajectory reconstruction tend to underlink pseudonyms, increasing the number of false negatives and leading to fragmented trajectories. Lastly, we observed that leveraging other kinematics, such as speed and heading, in the trajectory reconstruction process did

not yield a significant improvement. Moreover, in most cases, trajectory reconstruction with all-kinematics resulted in lower performance than with the location-based approach.

These findings demonstrated that pseudonym-changing mechanisms, as currently defined in both ETSI and SAE standards, do not provide strong privacy protection: adversaries can reconstruct most trajectories despite frequent pseudonym updates. Such reconstruction enables attackers to infer sensitive information, such as home or workplace locations, thereby undermining user trust in VANET systems. The high percentage of completely reconstructed trajectories poses a privacy threat that VANET users must accept in order to gain the benefit from the system. This trade-off between road safety and user privacy highlights a critical challenge in VANET design, ensuring safety without exposing users to severe tracking risks.

7.2 Future Work

Building on the scope, limitations, and findings of this thesis, several directions for future research could be taken in the study of privacy-focused VANET communications. As this thesis focused solely on the basic message types—BSM and CAM—further studies on other message types and privacy leakage on other entities could be conducted to understand the complete list of potential privacy threats in VANETs. A potential avenue for this path of future work is exploring the existing public datasets with Signal Phasing and Timing (SPaT) and Traveler Information Message (TIM) from the USDOT CV Pilot projects [114–116]. This path of future work can follow the processes we used in this research, including understanding the nature of messages and their applications, and reevaluating the privacy threats with the LINDDUN framework with a new set of assumptions. Another branch of this path of future work can be eliciting the privacy leakage with the LINDDUN framework on other entities involved in VANET communication that are available

in the Architecture Reference for Cooperative and Intelligent Transportation (ARC-IT). Recalling Figure 3.3 data flow of vehicle based on ARC-IT, each of the entities that is not the vehicle can be investigated for potential privacy threats with the LINDDUN framework.

Each of the identified privacy threats presented in Section 3.1 represents a potential path for investigation. Future work stemming from our findings could include a survey on awareness of VANETs' privacy risks and the acceptance rate of confidently participating in VANETs, after the potential privacy threats from this research have been presented to users to gain understanding of how much risk users can accept. Another branch of research can be a study on the success rate of user deanonymisation after the trajectory is reconstructed, where the reconstructed rate is not 100% success.

Data simulation is another important branch that can be separated into multiple works. From Section 4.2, we encountered technical difficulty in our first attempt at synthetic data simulation. This technical difficulty can open up into another future work, where the simulation process can be improved, either code-based by optimising the simulation code, or computation-based by parallelising the simulation. Additionally, as this research only used TAPAS Cologne for data simulation, a methodology for incorporating other real-world datasets for data simulation can be developed into a small project that could have a high impact on contributing datasets for VANETs privacy quantification. The concept from Buchholz et al. [7] is another work that can be aligned with this research, where privacy is considered as part of the simulation process and its result.

Future work should also aim to improve trajectory reconstruction models, highlighting a stronger adversary than those presented in this thesis. While the results from our evaluation in Chapter 6 already showed a high recall rate, the completely reconstructed trajectory rates varied among different trajectory reconstruction models. Avenues for future work in this branch include improving these models, for instance by fine-tuning the model parameters or

investigating methodologies to work around the models' weaknesses that were pointed out at the beginning of Chapter 5.

Lastly, methods such as "mix zone" for pseudonym-changing spots, which claim to perform better in preserving user privacy in VANETs, could be evaluated for their feasibility of implementation alongside current standards and tested in realistic scenarios.

References

[1] S. Warren and L. Brandeis, "The right to privacy," in *Killing the Messenger: 100 Years of Media Criticism*, pp. 1–21, Columbia University Press, 1989.

[2] "Regulation (EU) 2016/679 of the european parliament and of the council of 27 april 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing directive 95/46/EC (general data protection regulation) (text with EEA relevance)." Legislative Body: EP, CONSIL.

[3] H. Nissenbaum, "Privacy as contextual integrity," vol. 79.

[4] R. L. Finn, D. Wright, and M. Friedewald, "Seven types of privacy," *European data protection: coming of age*, pp. 3–32, 2013.

[5] R. Clarke, "Introduction to dataveillance and information privacy and definitions of terms," *www.anu.edu.au/people/Roger.Clarke/DV/Intro.html*, 1997.

[6] M. Howard and S. Lipner, *The security development lifecycle*, vol. 8. Microsoft Press Redmond, 2006.

[7] E. Buchholz, A. Abuadbba, S. Wang, S. Nepal, and S. S. Kanhere, "SoK: Can trajectory generation combine privacy and utility?," vol. 2024, no. 3, pp. 75–93.

[8] K. Emara, W. Woerndl, and J. Schlichter, "On evaluation of location privacy preserving schemes for VANET safety applications," vol. 63, pp. 11–23.

[9] P. Cichy, T. O. Salge, and R. Kohli, "Privacy concerns and data sharing in the internet of things: Mixed methods evidence from connected cars.," *Mis Quarterly*, vol. 45, no. 4, 2021.

[10] G. Bella, P. Biondi, M. D. Vincenzi, and G. Tudisco, "Privacy and modern cars through a dual lens," in *2021 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*, pp. 136–143, IEEE.

[11] Z. Cai and A. Xiong, "Understand users' privacy perception and decision of {V2X} communication in connected autonomous vehicles," in *32nd USENIX Security Symposium (USENIX Security 23)*, pp. 2975–2992, 2023.

[12] T. M. CORPORATION, "Apology and notice concerning newly discovered potential data leakage of customer information due to cloud settings | corporate | global newsroom."

[13] "Toyota data breach confirmed after 240gb of stolen data leaked on hacking forum - security spotlight." Section: Security Spotlight.

[14] G. Baran, "Volkswagen data breach: 800,000 electric car owners' data leaked."

[15] "Subaru security flaw allows hackers to remotely track and hijack your vehicle." Section: Security Spotlight.

[16] B. Condon, "Is your car spying on you? what it means that tesla shared data in the las vegas explosion." Section: Business.

[17] "Reflecting on the 2023 toyota data breach | CSA."

[18] Tampa Hillsborough Expressway Authority, "THEA Connected Vehicle Pilot." `https://theacvpilot.com/`, 2025. Accessed: 2023-09-26.

[19] Wyoming DOT, "Wyoming DOT Connected Vehicle Pilot." `https://wydotcvp.wyoroad.info/` and `https://web.archive.org/web/20230607200238/https://wydotcvp.wyoroad.info/`, 2019. Accessed: 2023-09-26.

[20] "Intelligent Transportation Systems - Joint Program Office Home Page." `https://www.its.dot.gov/index.htm` and `https://web.archive.org/web/20250103165615/https://www.its.dot.gov/index.htm`. Accessed: 2025-01-08.

[21] SAE, "SAE J2945/1 On-Board System Requirements for V2V Safety Communications," *Standard*, 2020.

[22] ETSI, "ETSI TS 102 941 V2.2.1 (2022-11): "Intelligent Transport Systems (ITS); Security; Trust and Privacy Management; Release 2"," *Standard*, 2022.

[23] ITS DOT, "SCMS CV pilots documentation : Use case 3: OBE pseudonym certificates provisioning." `https://www.its.dot.gov/research_areas/cybersecurity/scms/27426990.html` and

https://web.archive.org/web/20241204134924/https://its.dot.gov/research_areas/cybersecurity/scms/27426990.html. Accessed: 2025-01-12.

[24] Camp LLC—Vehicle Safety Communications 5 Consortium and others, "Security Credential Management System Proof-of-Concept Implementation: EE Requirements and Specifications Supporting SCMS Software Release 1.2.2," 2016.

[25] ETSI, "ETSI TR 103 415 V1.1.1 (2018-04): "Intelligent Transport Systems (ITS); Security; Pre-standardization study on pseudonym change management"," *Standard*, 2018.

[26] A. R. Beresford and F. Stajano, "Location privacy in pervasive computing," *IEEE Pervasive computing*, vol. 2, no. 1, pp. 46–55, 2003.

[27] H. Dok, R. Echevarria, and H. Fu, "Privacy issues for vehicular ad-hoc network," in *International Conference on Future Generation Communication and Networking*, pp. 370–383, Springer, 2009.

[28] D. Hahn, A. Munir, and V. Behzadan, "Security and privacy issues in intelligent transportation systems: Classification and challenges," *IEEE Intelligent Transportation Systems Magazine*, vol. 13, no. 1, pp. 181–196, 2019.

[29] A. Khoshgozaran and C. Shahabi, "Blind evaluation of nearest neighbor queries using space transformation to preserve location privacy," in *International symposium on spatial and temporal databases*, pp. 239–257, Springer, 2007.

[30] G. Ghinita, P. Kalnis, A. Khoshgozaran, C. Shahabi, and K.-L. Tan, "Private queries in location based services: anonymizers are not necessary," in *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pp. 121–132, 2008.

[31] B. Hoh, M. Gruteser, R. Herring, J. Ban, D. Work, J.-C. Herrera, A. M. Bayen, M. Annavaram, and Q. Jacobson, "Virtual trip lines for distributed privacy-preserving traffic monitoring," in *Proceedings of the 6th international conference on Mobile systems, applications, and services*, pp. 15–28, 2008.

[32] J. Meyerowitz and R. Roy Choudhury, "Hiding stars with fireworks: location privacy through camouflage," in *Proceedings of the 15th annual international conference on Mobile computing and networking*, pp. 345–356, 2009.

[33] R. Lu, X. Lin, T. H. Luan, X. Liang, and X. Shen, "Anonymity analysis on social spot based pseudonym changing for location privacy in vanets," in *2011 IEEE International Conference on Communications (ICC)*, pp. 1–5, IEEE, 2011.

[34] J. Lim, H. Yu, K. Kim, M. Kim, and S.-B. Lee, "Preserving location privacy of connected vehicles with highly accurate location updates," *IEEE Communications Letters*, vol. 21, no. 3, pp. 540–543, 2016.

[35] N. Guo, L. Ma, and T. Gao, "Independent mix zone for location privacy in vehicular networks," *IEEE Access*, vol. 6, pp. 16842–16850, 2018.

[36] K. Sampigethaya, L. Huang, M. Li, R. Poovendran, K. Matsuura, and K. Sezaki, "Caravan: Providing location privacy for vanet," *Proceedings of Embedded Security in Cars (ESCAR)*, vol. 8, 2005.

[37] M. Li, K. Sampigethaya, L. Huang, and R. Poovendran, "Swing & swap: user-centric approaches towards maximizing location privacy," in *Proceedings of the 5th ACM workshop on Privacy in electronic society*, pp. 19–28, 2006.

[38] K. Sampigethaya, M. Li, L. Huang, and R. Poovendran, "Amoeba: Robust location privacy scheme for vanet," *IEEE Journal on Selected Areas in communications*, vol. 25, no. 8, pp. 1569–1589, 2007.

[39] S. Jinyuan, Z. Chi, and F. Yuguang, "An id-based framework achieving privacy and non-repudiation," in *Proceedings of IEEE vehicular ad hoc networks, military communications conference (MILCOM 2007)*, pp. 1–7, 2007.

[40] L. Buttyán, T. Holczer, A. Weimerskirch, and W. Whyte, "Slow: A practical pseudonym changing scheme for location privacy in vanets," in *2009 IEEE vehicular networking conference (VNC)*, pp. 1–8, IEEE, 2009.

[41] B. Wiedersheim, Z. Ma, F. Kargl, and P. Papadimitratos, "Privacy in inter-vehicular networks: Why simple pseudonym change is not enough," in *2010 Seventh international conference on wireless on-demand network systems and services (WONS)*, pp. 176–183, IEEE, 2010.

[42] D. Eckhoff, C. Sommer, T. Gansen, R. German, and F. Dressler, "Strong and affordable location privacy in vanets: Identity diffusion using timeslots and swapping," in *2010 IEEE Vehicular Networking Conference*, pp. 174–181, IEEE, 2010.

[43] R. Lu, X. Lin, T. H. Luan, X. Liang, and X. Shen, "Pseudonym changing at social spots: An effective strategy for location privacy in vanets," *IEEE transactions on vehicular technology*, vol. 61, no. 1, pp. 86–96, 2011.

[44] D. Eckhoff and C. Sommer, "Marrying safety with privacy: A holistic solution for location privacy in vanets," in *2016 IEEE Vehicular Networking Conference (VNC)*, pp. 1–8, IEEE, 2016.

[45] A. Boualouache and S. Moussaoui, "Tapcs: Traffic-aware pseudonym changing strategy for vanets," *Peer-to-Peer networking and Applications*, vol. 10, pp. 1008–1020, 2017.

[46] A. Wahid, H. Yasmeen, M. A. Shah, M. Alam, and S. C. Shah, "Holistic approach for coupling privacy with safety in vanets," *Computer networks*, vol. 148, pp. 214–230, 2019.

[47] R. Shokri, G. Theodorakopoulos, J.-Y. Le Boudec, and J.-P. Hubaux, "Quantifying location privacy," in *2011 IEEE Symposium on Security and Privacy*, pp. 247–262, IEEE.

[48] I. Wagner and D. Eckhoff, "Technical privacy metrics: A systematic survey," vol. 51, no. 3, pp. 1–38.

[49] I. Wagner, "Measuring privacy in vehicular networks," in *2017 IEEE 42nd Conference on Local Computer Networks (LCN)*, pp. 183–186, IEEE.

[50] G. P. Corser, H. Fu, and A. Banihani, "Evaluating location privacy in vehicular communications and applications," vol. 17, no. 9, pp. 2658–2667.

[51] J. Cui, J. Wen, S. Han, and H. Zhong, "Efficient privacy-preserving scheme for real-time location data in vehicular ad-hoc network," *IEEE Internet of Things Journal*, vol. 5, no. 5, pp. 3491–3498, 2018.

[52] N. D'Silva, T. Shahi, Ø. T. D. Husveg, A. Sanjeeve, E. Buchholz, and S. S. Kanhere, "Demystifying trajectory recovery from ash: An open-source evaluation and enhancement," *arXiv preprint arXiv:2409.14645*, 2024.

[53] M. Gruteser and D. Grunwald, "Anonymous usage of location-based services through spatial and temporal cloaking," in *Proceedings of the 1st international conference on Mobile systems, applications and services*, pp. 31–42, 2003.

[54] G. Ghinita, M. L. Damiani, C. Silvestri, and E. Bertino, "Preventing velocity-based linkage attacks in location-aware applications," in *Proceedings of the 17th ACM SIGSPATIAL international conference on advances in geographic information systems*, pp. 246–255, 2009.

[55] Z. Ma, F. Kargl, and M. Weber, "A location privacy metric for v2x communication systems," in *2009 IEEE Sarnoff Symposium*, pp. 1–6, IEEE.

[56] N. Bißmeyer, H. Stübing, E. Schoch, S. Götz, J. P. Stotz, and B. Lonc, "A generic public key infrastructure for securing car-to-x communication," in *18th ITS World Congress, Orlando, USA*, vol. 14, 2011.

[57] L. Frank, D. Garcia, E. Hurley, A. Kiernan, N. Nahas, R. Walsh, and B. A. Hamilton, "Security credentials management system (scms) design and analysis for the connected vehicle system: draft.," 2013.

[58] W. Whyte, A. Weimerskirch, V. Kumar, and T. Hehn, "A security credential management system for v2v communications," in *2013 IEEE Vehicular Networking Conference*, pp. 1–8, IEEE, 2013.

[59] J. Schneeberger and United States. Department of Transportation. Intelligent Transportation Systems Joint Program Office, "Security credential management system (SCMS) technical primer."

[60] "Architecture Reference for Cooperative and Intelligent Transportation." `https://www.arc-it.net/`. Accessed: 2025-03-20.

[61] "ITS architecture and standards | ITS joint program office." `https://www.its.dot.gov/resources/Architecture-and-Standards/`. Accessed: 2025-03-20.

[62] "Physical objects, vehicle." `https://www.arc-it.net/html/physobjects/physobj4.html`. Accessed: 2025-03-20.

[63] "Physical objects, vehicle, triples." `https://www.arc-it.net/html/physobjects/physobj4.html#tab-2`. Accessed: 2025-03-20.

[64] "Vehicle - other vehicles: vehicle location and motion." `https://www.arc-it.net/html/triples/s4d6f8.html#tab-3`. Accessed: 2025-03-20.

[65] "tov-vehicle_location." `https://www.arc-it.net/html/dataflows2/df2440.html`. Accessed: 2025-03-20.

[66] "Vehicle basic safety communication." https://www.arc-it.net/html/functionalobjects/funobj39.html#tab-1. Accessed: 2025-03-20.

[67] "Physical objects, vehicle, security." https://www.arc-it.net/html/physobjects/physobj4.html#tab-3. Accessed: 2025-03-20.

[68] ETSI, TR, "ETSI TR 102 638 (V1.1.1) (2009-06): "Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Definitions"," *Standard*, 2009.

[69] SAE, "SAE J2735 (R) V2X Communications Message Set Dictionary," *Standard*, 2020.

[70] ETSI, "ETSI TS 102 894-2 V1.3.1 (2018-08): "Intelligent Transport Systems (ITS); Users and applications requirements; Part 2: Applications and facilities layer common data dictionary"," *Standard*, 2018.

[71] ETSI, "ETSI EN 302 637-2 (V1.4.1) (2019-04): "Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 2: Specification of Cooperative Awareness Basic Service"," *Standard*, 2019.

[72] ETSI, "ETSI TS 102 940 V2.1.1 (2021-07): "Intelligent Transport Systems (ITS); Security; ITS communications security architecture and security management; Release 2"," *Standard*, 2021.

[73] ETSI, "ETSI TS 103 097 V2.1.1 (2021-10s): "Intelligent Transport Systems (ITS); Security; Security header and certificate formats; Release 2"," *Standard*, 2021.

[74] S. Johnson, L. Rolfes, V. J. Blue, S. L. Reich, HNTB Corporation, and Tampa Hillsborough Expressway Authority, "Connected vehicle pilot deployment program phase II data privacy plan - tampa (THEA).."

[75] SAE, "SAE J2945/2 Dedicated Short Range Communications (DSRC) Performance Requirements for V2V Safety Awareness," *Standard*, 2018.

[76] M. Deng, K. Wuyts, R. Scandariato, B. Preneel, and W. Joosen, "A privacy threat analysis framework: supporting the elicitation and fulfillment of privacy requirements," vol. 16, no. 1, pp. 3–32.

[77] K. Wuyts and W. Joosen, "Linddun privacy threat modeling: a tutorial," *CW Reports*, 2015.

[78] "Privacy threat trees | linddun.org." `https://linddun.org/threat-trees/`. Accessed: 2025-03-30.

[79] B. Chah, A. Lombard, A. Bkakria, R. Yaich, A. Abbas-Turki, and S. Galland, "Privacy threat analysis for connected and autonomous vehicles," *Procedia Computer Science*, vol. 210, pp. 36–44, 2022.

[80] M. Raciti and G. Bella, "How to model privacy threats in the automotive domain," *arXiv preprint arXiv:2303.10370*, 2023.

[81] M. Raciti and G. Bella, "A threat model for soft privacy on smart cars," in *2023 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*, pp. 1–10, IEEE.

[82] R. Shokri, G. Theodorakopoulos, G. Danezis, J.-P. Hubaux, and J.-Y. Le Boudec, "Quantifying location privacy: the case of sporadic location exposure," in *International Symposium on Privacy Enhancing Technologies Symposium*, pp. 57–76, Springer, 2011.

[83] R. Shokri, J. Freudiger, M. Jadliwala, and J.-P. Hubaux, "A distortion-based metric for location privacy," in *Proceedings of the 8th ACM workshop on Privacy in the electronic society*, pp. 21–30, ACM, 2009.

[84] Z. Ma, F. Kargl, and M. Weber, "Measuring location privacy in v2x communication systems with accumulated information," in *2009 IEEE 6th International Conference on Mobile Adhoc and Sensor Systems*, pp. 322–331, IEEE.

[85] R. Shokri, J. Freudiger, and J.-P. Hubaux, "A unified framework for location privacy," 2010.

[86] U.S. Department of Transportation, "U.S. DOT ITS Connected Vehicle Pilot Sandbox." `https://usdot-its-cvpilot-publicdata.s3.amazonaws.com/index.html`, 2019.

[87] "Connected vehicle pilot (CVP) open data."

[88] L. Bracciale, M. Bonola, P. Loreti, G. Bianchi, R. Amici, and A. Rabuffi, "Crawdad roma/taxi," 2022.

[89] M. Gramaglia, O. Trullols-Cruces, D. Naboulsi, M. Fiore, and M. Calderon, "Vehicular networks on two madrid highways," in *2014 Eleventh Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*, pp. 423–431, 2014.

[90] M. Piorkowski, N. Sarafijanovic-Djukic, and M. Grossglauser, "Crawdad epfl/mobility," 2022.

[91] J. Yuan, Y. Zheng, C. Zhang, W. Xie, X. Xie, G. Sun, and Y. Huang, "T-drive: Driving directions based on taxi trajectories," in *Proceedings of 18th ACM SIGSPATIAL Conference on Advances in Geographical Information Systems*, ACM SIGSPATIAL GIS 2010, November 2010. Best Paper Award.

[92] E. Buchholz, A. Abuadbba, S. Wang, S. Nepal, and S. S. Kanhere, "Reconstruction attack on differential private trajectory protection mechanisms," in *Proceedings of the 38th Annual Computer Security Applications Conference*, pp. 279–292, ACM.

[93] Y. Zheng, H. Fu, X. Xie, W.-Y. Ma, and Q. Li, *Geolife GPS trajectory dataset - User Guide*, geolife gps trajectories 1.1 ed., July 2011.

[94] M. D. Pesé, X. Pu, and K. G. Shin, "SPy: Car steering reveals your trip route!," vol. 2020, no. 2, pp. 155–174.

[95] "TAPASCologne - SUMO documentation." `https://sumo.dlr.de/docs/Data/Scenarios/TAPASCologne.html`.

[96] MITRE, "Technical memorandum: Modeling and simulation of areas of potential v2v privacy risk."

[97] K. Wang, L. Wang, and M. Cui, "Trajectory tracking and recovery attacks in VANET systems," vol. 31, no. 17, p. e3797.

[98] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Transactions of the ASME–Journal of Basic Engineering*, vol. 82, no. Series D, pp. 35–45, 1960.

[99] H. Ghorbani, "Mahalanobis distance and its application for detecting multivariate outliers," *Facta Universitatis, Series: Mathematics and Informatics*, pp. 583–595, 2019.

[100] "Physical objects, vehicle, functionality." `https://www.arc-it.net/html/physobjects/physobj4.html#tab-1`. Accessed: 2025-03-20.

[101] "Physical objects, vehicle, interfaces diagram." `https://www.arc-it.net/html/physobjects/physobj4.html#tab-4`. Accessed: 2025-03-20.

[102] S. H. Jodka, "Dashboard confessions: unveiling the privacy issues in connected cars,"

[103] P. Golle and K. Partridge, "On the anonymity of home/work location pairs," in *International Conference on Pervasive Computing*, pp. 390–397, Springer, 2009.

[104] GeeksforGeeks, "Haversine formula to find distance between two points on a sphere." `https://www.geeksforgeeks.org/haversine-formula-to-find-distance-between-two-points-on-a-sphere/`, 2022.

[105] Tampa Hillsborough Expressway Authority, "Tampa CV Pilot Basic Safety Message (BSM) Sample." `https://data.transportation.gov/Automobiles/Tampa-CV-Pilot-Basic-Safety-Message-BSM-Sample/nm7w-nvbm`, 2019.

[106] Wyoming Department of Transportation, "Wyoming cv pilot basic safety message one day sample." `https://data.transportation.gov/Automobiles/Wyoming-CV-Pilot-Basic-Safety-Message-One-Day-Samp/7qu7-j8rx`, 2019.

[107] United States Department of Transportation. Federal Highway Administration. Intelligent Transportation Systems Joint Programs Office (ITS JPO), "New york city CV pilot event message one month sample." `https://data.transportation.gov/d/vxmm-9bi5`.

[108] A. Varga and R. Hornig, "An overview of the omnet++ simulation environment," in *Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops*, pp. 1–10, 2008.

[109] C. Sommer, R. German, and F. Dressler, "Bidirectionally coupled network and road traffic simulation for improved ivc analysis," *IEEE Transactions on Mobile Computing*, vol. 10, no. 1, pp. 3–15, 2011.

[110] R. Riebl, H.-J. Gunther, C. Facchi, and L. Wolf, "Artery: Extending veins for VANET applications," in *2015 International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS)*, pp. 450–456, IEEE.

[111] S. Uppoor and M. Fiore, "Large-scale urban vehicular mobility for networking research," in *2011 IEEE Vehicular Networking Conference (VNC)*, pp. 62–69, IEEE.

[112] S. S. Blackman and R. Popoli, "Design and analysis of modern tracking systems," *(No Title)*, 1999.

[113] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with applications to tracking and navigation: theory algorithms and software*. John Wiley & Sons, 2001.

[114] Tampa Hillsborough Expressway Authority, "Tampa CV pilot signal phasing and timing (SPaT) sample." `https://data.transportation.gov/d/xn7c-yu2n`.

[115] Tampa Hillsborough Expressway Authority, "Tampa CV pilot traveler information message (TIM) sample." `https://data.transportation.gov/d/in46-gmir`.

[116] Wyoming Department Of Transportation, "Wyoming CV pilot traveler information message sample (schema version 6)." `https://data.transportation.gov/d/6nxx-nmxk`.

Appendices

Appendix A

Information in BSM and CAM

Table A.1 presents information fields in the Basic Safety Message (BSM) and Cooperative Awareness Message (CAM) according to the standards [21,69,70]. For a comparison between BSM and CAM, the fields representing similar information are grouped into a row. The table's columns show the information that is presented in the message, the requisite information to broadcast the message, the information's unit, and its value ranges. According to the standards [21,70], the vehicle must be able to provide all of the requisite information to broadcast the message; otherwise, the message will not be broadcast. It is worth noting that to avoid an invalid message due to not being able to supply the requisite information, these information fields usually allow an 'unavailable' value to be used in the field. For example, the enumerated value 15 in the altitude confidence in CAM represents the inability to provide the confidence value of the altitude.

Due to a space limitation, we shortened the word 'enumerate' in the unit column to 'enum.' The enumerated unit usually represents a range of values; therefore, it must be converted to the actual range before being used. For instance, the value 0 in altitude confidence in CAM represents that the accuracy of the altitude's value is less than or equal to 0.01 metres.

Table A.1: Data elements in Basic Safety Message and Cooperative Awareness Message.

Information	**Required**		**Unit**		**Value Range**	
	BSM	**CAM**	**BSM**	**CAM**	**BSM**	**CAM**
Pseudo-identity	✓	✓	octet string	-	size(4)	0–4294967295
SecondMark	✓	✓	ms	ms	0–65535	0–65535
Latitude	✓	✓	0.1 $\mu°$	0.1 $\mu°$	-900M–900M	-900M–900M
Longitude	✓	✓	0.1 $\mu°$	0.1 $\mu°$	-18B–18B	-18B–18B
Altitude	✓	✓	10 cm	1 cm	-4096–61439	-100K–800K
Semi-major confidence	✓	✓	5 cm	1 cm	0–255	0–4095
Semi-minor confidence	✓	✓	5 cm	1 cm	0–255	0–4095
Semi-major orientation	✓	✓	0.0055°	0.1°	0–65535	0–3601
Altitude confidence		✓	-	enum.	-	0–15
Heading	✓	✓	0.0125°	0.1°	0–28800	0–3601
Heading confidence		✓	-	0.1°	-	0–127
Speed	✓	✓	2 cm/s	1 cm/s	0–8191	0–16383
Speed confidence		✓	-	1 cm/s	-	1–127
Drive direction	✓	✓	enum.	enum.	0–3	0–2
Longitudinal acceleration	✓	✓	1 cm/s^2	0.1 m/s^2	-2000–2001	-160–161

Continued on next page

Table A.1: Data elements in Basic Safety Message and Cooperative Awareness Message. (Continued)

Information	Required		Unit		Value Range	
	BSM	CAM	BSM	CAM	BSM	CAM
Longitudinal acceleration confidence		✓	-	0.1 m/s^2	-	0–102
Lateral acceleration	✓	✓	0.01 m/s^2	0.01 m/s^2	-2000–2001	-160–161
Vertical acceleration	✓		0.1962 m/s^2	-	-127–127	-
Vehicle length	✓	✓	1 cm	0.1 m	0–4095	1–1023
Vehicle length confidence		✓	-	enum.	-	0–4
Vehicle width	✓	✓	1 cm	0.1 m	0–1023	1–62
Curvation		✓	-	1/10000 m	-	-1023–1023
Curvation confidence		✓	-	enum.	-	0–7
Curvation calculation mode		✓	-	enum.	-	0–2
Yaw rate	✓	✓	0.01°/s	0.01°/s	-32767–32767	-32766–32767
Yaw rate confidence		✓	-	enum.	-	0–8
Message count	✓		-	-	0–127	-
Transmission State	✓		enum.	-	0–7	-

Continued on next page

Table A.1: Data elements in Basic Safety Message and Cooperative Awareness Message. (Continued)

Information	**Required**		**Unit**		**Value Range**	
	BSM	**CAM**	**BSM**	**CAM**	**BSM**	**CAM**
Steering wheel angle	✓		1.5°	-	-126–127	-
Brake system status	✓		string sequence	-		-
Path History Point						
Longitude offset	✓	✓	0.1 μ°	0.1 μ°	-131072–131071	-131071–131072
Latitude offset	✓	✓	0.1 μ°	0.1 μ°	-131072–131071	-131071–131072
Altitude offset	✓	✓	10 cm	1 cm	-2048–2047	-12700–12800
Time offset	✓		0.01 s	0.01 s	1–65535	1–65535

It is also worth noting that the path history in both standards [69, 70] will be presented as an array of path history points, where each offset value in the path history point is calculated from the previously broadcast message and its associated value presented in the current message. The array of path history points is also sorted, such that the least index represents the most recent message history broadcast by the vehicle [21].

Appendix B

ARC-IT's Security Class

The Architecture Reference for Cooperative and Intelligent Transportation (ARC-IT) is an architecture used in the planning, designing, and implementing an intelligent transportation system (ITS) [60]. The ARC-IT provides various information that can be considered during the development of ITS. This appendix provides the information gathered from the ARC-IT's website that will be used during the privacy threats analysis in the thesis. This information includes the device security classes, the vehicle's functionality and its security classes, the information's source and destination, known as *triple's source and destination*, and its related security classes.

A device security class represents security requirements that a device must hold in order to participate in ITS's applications. Assigning the security class to a device or physical object in ITS takes the requirement for the security triad into account. Table B.1 shows the list of device security classes and their associated security requirements. Based on the table, the class 1 device security class represents the lowest security requirements for a device, and the class 5 is the highest security class.

The next information that we gathered from the ARC-IT is the vehicle's functionality and its associated device security class. Since this thesis focuses on the privacy of the VANET's users, the vehicle physical object in ARC-IT becomes our focus point to explore its related service packages, functionalities,

Table B.1: Device security classes and associated security requirements.

Security Class	Confidentiality	Integrity	Availability
Class 1	Low	Moderate	Moderate
Class 2	Moderate	Moderate	Moderate
Class 3	Moderate	High	Moderate
Class 4	High	High	Moderate
Class 5	High	High	High

and security classes. We list the vehicle's service packages, functionalities, and the triple's source and destination in Figures 3.1 and 3.2. A service package is equivalent to an application which the vehicle can use during its participation in the intelligent transport system. These service packages, when considered with the vehicle object, have a device security class that must be achieved to maintain the confidentiality, integrity, and availability of the information that is used in the service package. The service package's security class is depicted in Figures 3.1 and 3.2 as the typeset of the service package.

Additionally, each service package will also have functionalities that enable it to perform its tasks. Since the service package performs these functionalities, we assume that the functionality will carry the security class that is associated with the service package. Table B.2 lists the functionalities that the vehicle must be able to perform to be able to use all service packages that are listed in Figures 3.1 and 3.2. It is worth noting that the same functionality can be used in multiple service packages, and the service packages may have different security classes that must be achieved. Therefore, a functionality may have multiple security classes attached to it. In Figure 3.1 and 3.2, we choose to represent the highest security class, having the largest number of security classes, that is attached to the functionality.

Table B.2: Vehicle's functionalties and the security classes.

Functionality	**Class 5**	**Class 4**	**Class 3**	**Class 2**	**Class 1**
Vehicle Basic Safety Communication			✓	✓	✓
Vehicle Control Automation			✓		✓
Vehicle Control Warning			✓		✓
Vehicle Cooperative Cruise Control			✓		
Vehicle Device Coordination			✓		
Vehicle Eco-Driving Assist			✓	✓	
Vehicle Emergency Notification			✓		
Vehicle Emissions Monitoring				✓	
Vehicle Environmental Monitoring			✓	✓	✓
Vehicle Gap Assist			✓		
Vehicle Intersection Warning			✓	✓	
Vehicle Loading Zone Access				✓	
Vehicle Location Determination	✓				
Vehicle Map Management			✓		
Vehicle METR Discrepancy Reporting					
Vehicle METR Interpretation					
Vehicle METR Management				✓	
Vehicle METR Reception					
Vehicle Platoon Operations			✓		

Continued on next page

Table B.2: Vehicle's functionalties and the security classes. (Continued)

Functionality	**Class 5**	**Class 4**	**Class 3**	**Class 2**	**Class 1**
Vehicle Queue Warning			✓		
Vehicle Rail Crossing Warning			✓		
Vehicle Restricted Lanes Application			✓	✓	✓
Vehicle Safety Monitoring					✓
Vehicle Secure Area Access System				✓	
Vehicle Situation Data Monitoring			✓	✓	
Vehicle Speed Management Assist					✓
Vehicle Support Services					✓
Vehicle System Executive			✓		
Vehicle System Monitoring and Diagnostics			✓		
Vehicle Traveler Information Reception			✓	✓	✓

Lastly, the functionalities listed in the previous table will have the information source and destination for exchanging information to perform their task. Similar to the previous table, we assume that the information source and destination, known as the triple source and destination in ARC-IT, carry the security class from the functionality they are attached to. Table B.3 and B.4 list the triple's source and destination that are related to the service packages used by a vehicle. Following the same manner as the vehicle's functionalities, we select the highest security class that is attached to the triple's source and destination to be shown in Figures 3.1 and 3.2.

Table B.3: Vehicle's triple source and the security classes.

Triple Source	**Class 5**	**Class 4**	**Class 3**	**Class 2**	**Class 1**
Basic Vehicle	✓		✓	✓	✓
Commercial Vehicle OBE			✓	✓	✓
Connected Vehicle Roadside Equipment	✓		✓	✓	✓
Driver			✓	✓	✓
Emergency Management Center			✓		
Emergency Vehicle OBE			✓	✓	✓
Enforcement Center			✓		
ITS Object					✓
Location and Time Data Source	✓				
Maint and Constr Vehicle OBE			✓	✓	✓
Map Update System			✓	✓	
METR Distribution Center				✓	
Micromobility Vehicle OBE			✓		
Other Vehicles			✓	✓	✓
Parking Area Equipment				✓	
Personal			✓		
Personal Information Device			✓		
Potential Obstacles					✓
Roadway Environment			✓	✓	✓
Transit Vehicle OBE			✓	✓	✓

Continued on next page

Table B.3: Vehicle's triple source and the security classes. (Continued)

Triple Source	Class 5	Class 4	Class 3	Class 2	Class 1
Transportation Information Center			✓	✓	✓
Vehicle			✓	✓	✓
Vehicle Characteristics			✓		✓
Vehicle Service Center			✓		
Vulnerable Road Users			✓		✓

Table B.4: Vehicle's triple destination and the security classes.

Triple Destination	Class 5	Class 4	Class 3	Class 2	Class 1
Basic Vehicle			✓	✓	✓
Connected Vehicle Roadside Equipment			✓	✓	✓
Driver			✓	✓	✓
Emergency Management Center			✓		
Emergency Vehicle OBE			✓		
Enforcement Center			✓		
Map Update System			✓		
METR Discrepancy Handling System				✓	
METR System				✓	
Micromobility Vehicle OBE			✓	✓	✓
Other Vehicles			✓	✓	✓
Parking Area Equipment				✓	

Continued on next page

Table B.4: Vehicle's triple destination and the security classes. (Continued)

Triple Destination	**Class 5**	**Class 4**	**Class 3**	**Class 2**	**Class 1**
Personal			✓		
Personal Information Device			✓	✓	✓
Personnel Device			✓	✓	✓
Service Monitor System			✓		
Transportation Information Center			✓	✓	✓
Vehicle	✓		✓	✓	✓
Vehicle Service Center			✓		

Appendix C

LINDDUN Threat Trees

This appendix presents the LINDDUN privacy threat trees that were used during the privacy threat elicitation in Section 3.1. To identify the privacy threats, we first identify potential categories of privacy threats with Table 2.2. Then, for each privacy threat category, an associated privacy threat tree is used to identify the privacy threat that exists in the system. There are seven trees in total, each representing one LINDDUN category: linking, identifying, non-repudiation, detecting, data disclosure, unawareness, and non-compliance. The privacy threat tree's root represents the category of the privacy threat. Each node between the root and leaf node represents an assumption about the system, i.e., a condition that must be met for the privacy threat to occur. Each leaf node in the privacy threat tree represents a privacy threat that presents in a system if all conditions between the root and leaf nodes are met.

To use the privacy threat tree, the privacy threat elicitator performs a depth-first search on each of the privacy threat trees. Upon reaching an intermediate node between the root and leaf nodes, the privacy threat elicitator analyses whether the node's condition is present in the system. Assumptions can be made about the system on these nodes to continue traversing to the tree's leaf node. Once the leaf node is reached, the privacy threat presented on the leaf node is considered present in the system. Then, the privacy threat on the leaf node is documented, along with assumptions on intermediate nodes.

Lastly, the privacy threat elicitor continues the depth-first search until all of the privacy threat trees are explored. The following figures illustrate seven privacy threat trees, taken from [78], used during our VANETs privacy threat elicitation. Due to the space limit, we highlight the leaf nodes with a bold border instead of presenting all of them in the lowest level of the figure.

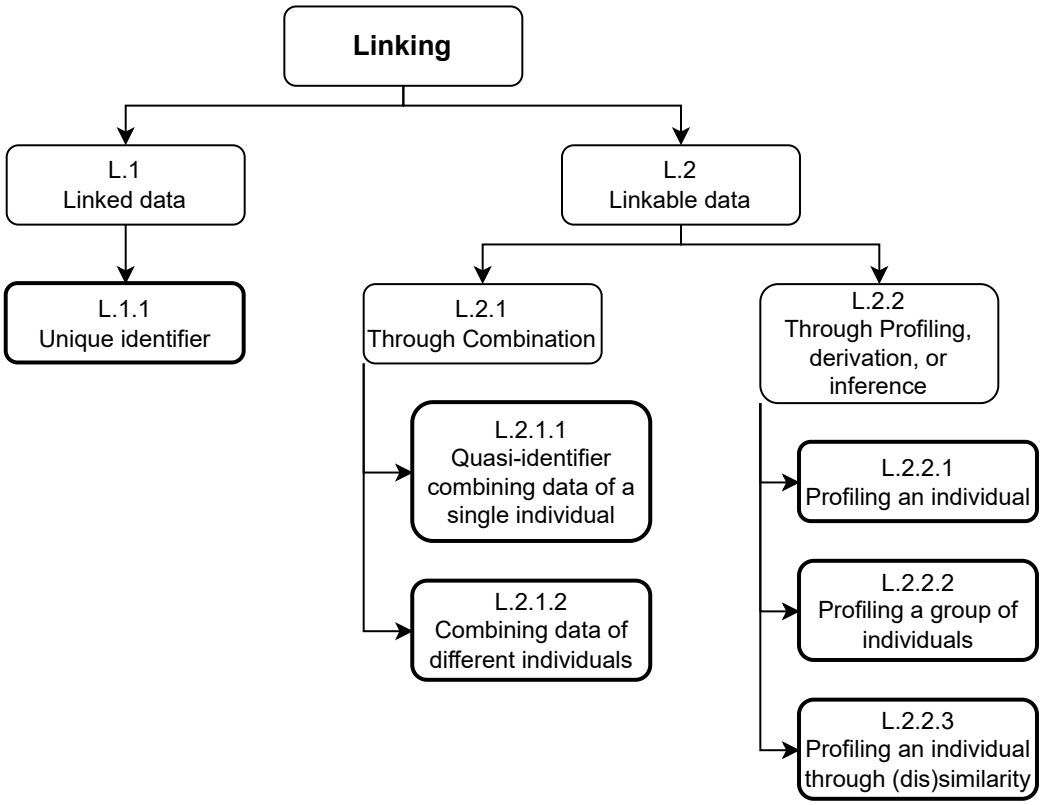


Figure C.1: Threat tree - Linking.

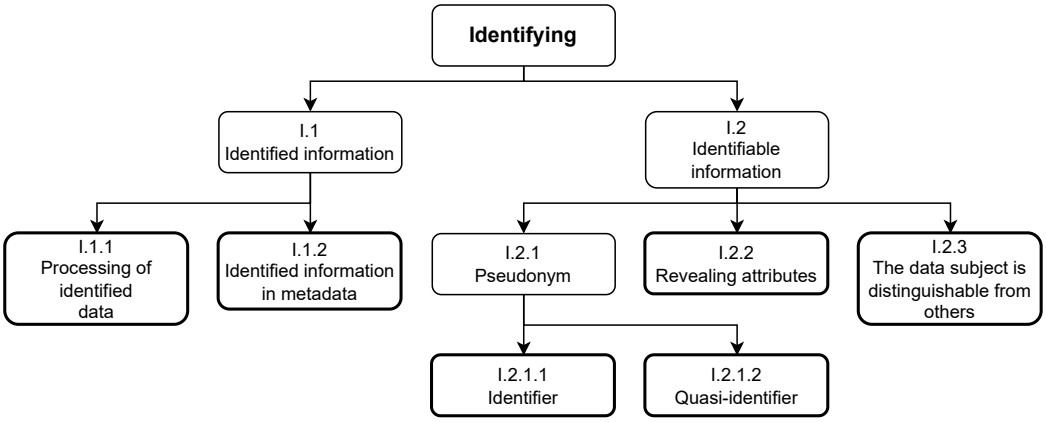


Figure C.2: Threat tree - Identifying.

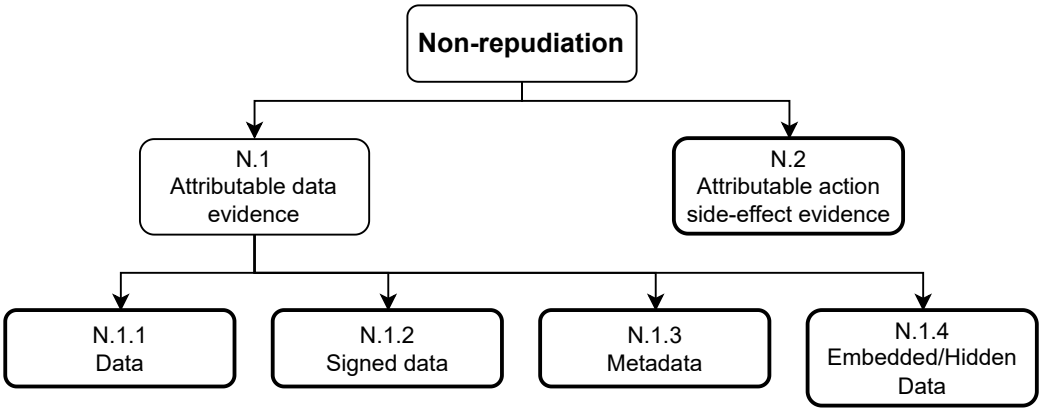


Figure C.3: Threat tree - Non-repudiation.

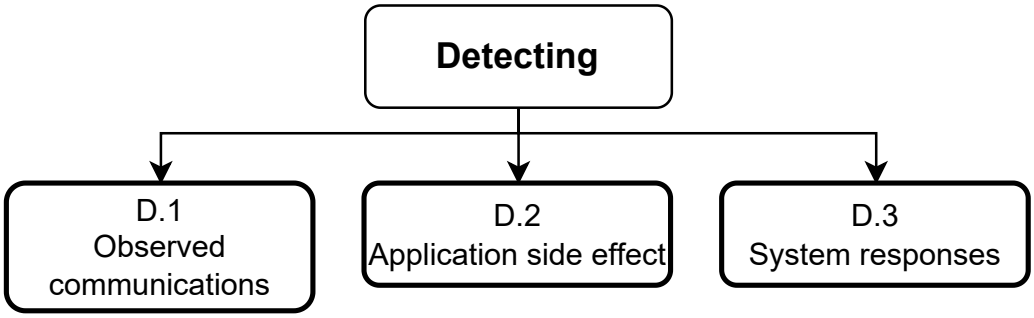


Figure C.4: Threat tree - Detecting.

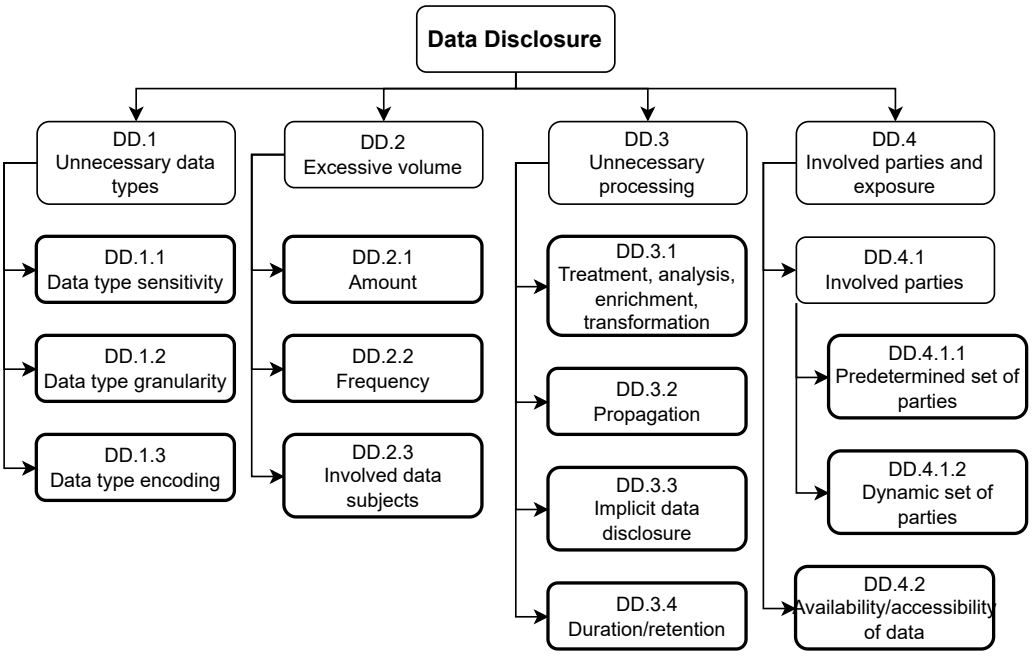


Figure C.5: Threat tree - Data Disclosure.

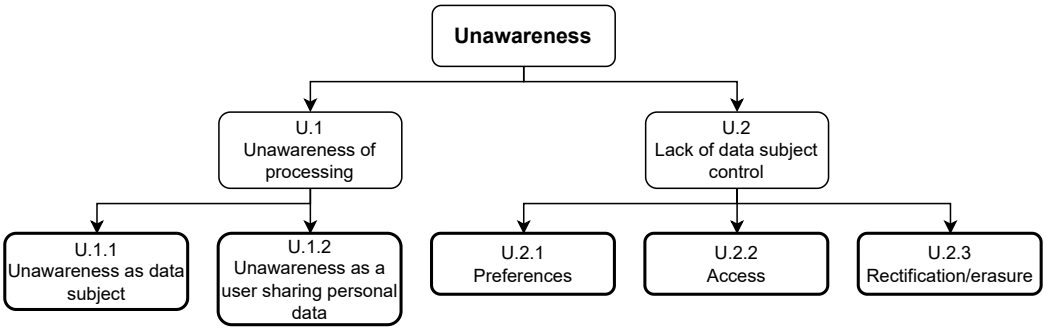


Figure C.6: Threat tree - Unawareness.

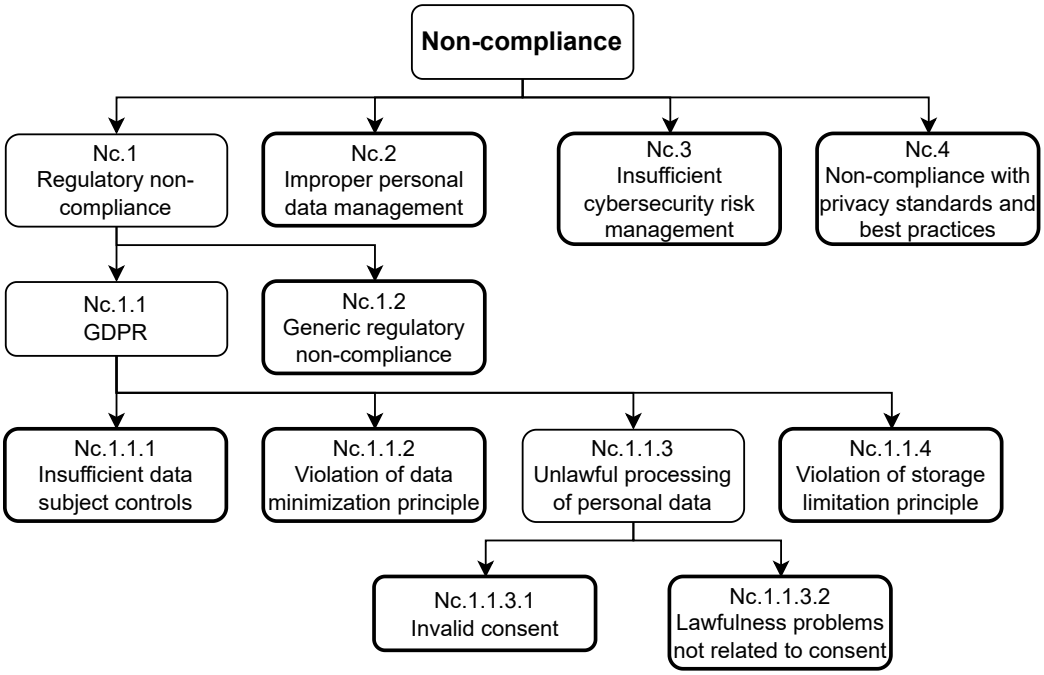


Figure C.7: Threat tree - Non-compliance.