KTH ROYAL INSTITUTE
OF TECHNOLOGY

Degree Project in Computer Science and Engineering

Second cycle, 30 credits

Misbehaviour mitigation in vehicular platoons by formation restructuring

A novel approach to attack mitigation

MICHAEL C. HARTMANN

Stockholm, Sweden, 2024

Misbehaviour mitigation in vehicular platoons by formation restructuring

A novel approach to attack mitigation

MICHAEL C. HARTMANN

Master’s Programme, Computer Science, 120 credits
Date: May 27, 2024

Supervisor: Konstantinos Kalogiannis
Examiner: Panagiotis Papadimitratos
School of Electrical Engineering and Computer Science
Swedish title: Begränsning av felaktigt beteende i fordonsplutoner genom omstrukturering av formationen
Swedish subtitle: En ny metod för att mildra angrepp

Abstract

In the context of autonomous vehicular transportation, platooning has emerged as a promising approach to optimize traffic flow, reduce fuel consumption, and enhance road safety by enabling vehicles to travel in close proximity to one another. This is done by leveraging the networking capabilities of modern vehicles. However, the susceptibility of platooning systems to malicious attacks necessitates robust misbehaviour mitigation protocols. This study introduces PRIME (Platoon Restructuring for Incident Mitigation and Exclusion), a novel misbehaviour mitigation protocol aimed at preventing disturbances within vehicular platooning systems. By utilizing the platoon's structure, PRIME effectively isolates and excludes attackers, positioning the accused vehicle at the rear of the convoy. The protocol aims furthermore to be robust against colluding attackers and against abuse of the protocol itself. The protocol's performance is assessed both independently with time based mitigation and in conjunction with an additional mitigation system based on the evaluation of received beacon data. To ensure the validity of the obtained results, the study also analyzes and explains several shortcomings of the PLEXE simulation environment used throughout this study, revealing instances where unexpected outcomes were produced. These discrepancies are attributed to the implementation nuances of certain platooning controllers within PLEXE, shedding light on potential areas for refinement within vehicle simulation environments for accurate evaluation of attacks on platooning protocols. Through extensive testing across various attack scenarios, PRIME demonstrates its capacity to restore platoon stability after the attack has been detected, showcasing promising prospects for enhancing platooning system security. The results indicate that, when combined with a quick and reliable detection mechanism all simulated attacks can be effectively mitigated and the attacker can be isolated to prevent future disturbances.

Keywords

V2V security, CACC, Platooning, Vehicular Protocols, Misbehaviour Mitigation

Sammanfattning

Inom området för autonom fordonstransport har platoonkörning under de senaste åren framstått som en lovande metod för att optimera det allmänna trafikflödet, minska bränsleförbrukningen hos de involverade fordonen och även öka trafiksäkerheten genom att låta fordon köra nära varandra. Detta uppnås genom att utnyttja kommunikationsförmågan hos moderna fordon. Sårbarheten hos nätverkssystem för cyberattacker kräver dock robusta och motståndskraftiga protokoll för att minska missbruk. Denna studie presenterar PRIME (Platoon Restructuring for Incident Mitigation and Exclusion), ett nytt protokoll för missbruksreduktion som syftar till att förhindra störningar i platooner. Genom att utnyttja platoonens struktur isolerar och utesluter PRIME effektivt angripare genom att placera det anklagade fordonet längst bak i konvojen. Protokollet syftar också till att vara robust mot samverkande angripare och mot missbruk av protokollet självt. Protokollets prestanda utvärderas både självständigt med tidsbaserad minskning och i kombination med ett ytterligare minskningssystem baserat på utvärdering av mottagna beacon-data. För att säkerställa giltigheten av de erhållna resultaten analyserar och förklarar denna studie även flera problem med platoon-simuleringsmiljön PLEXE, som användes under studien, och visar fall där oväntade resultat uppnåddes. Dessa diskrepanser kan till stor del tillskrivas implementeringen av vissa kontroller inom PLEXE och belyser därmed potentiella områden för förbättring av simuleringsmiljöer för en mer noggrann utvärdering av attacker mot platoon-protokoll. Genom omfattande tester i olika attackscenarier visar PRIME sin förmåga att återställa platoonens stabilitet efter att en attack har upptäckts och ger därmed lovande utsikter för att förbättra platoonsäkerheten. Resultaten av denna studie tyder på att med PRIME, i kombination med en snabb och pålitlig attackdetektion, kan alla simulerade attacker effektivt avvisas och att angriparen kan tillräckligt isoleras för att förhindra framtida störningar

Nyckelord

V2V Säkerhet, CACC, Platooning, Fordonsprotokoll, Angreppsbegränsning

Zusammenfassung

Im Bereich des autonomen Fahrzeugtransports hat sich Platooning in den letzen Jahren als vielversprechender Ansatz zur Optimierung des allgemeinen Verkehrsflusses, zur Reduzierung des Kraftstoffverbrauchs der involvierten Fahrzeuge und auch zur Erhöhung der Verkehrssicherheit herausgestellt, indem Fahrzeuge in enger Nähe zueinander fahren können. Dies wird durch die Nutzung der Kommunikationsfähigkeiten moderner Fahrzeuge erreicht. Die Anfälligkeit von vernetzen Systemen für Cyber-Angriffe erfordert jedoch robuste und resiliente Protokolle zur Missbrauchsminderung. Diese Studie stellt PRIME (Platoon Restructuring for Incident Mitigation and Exclusion) vor, ein neuartiges Protokoll zur Missbrauchsminderung, das darauf abzielt, Störungen in Platoons zu verhindern. Durch Nutzung der Platoonstruktur isoliert und schließt PRIME Angreifer effektiv aus, indem das beschuldigte Fahrzeug am Ende des Konvois platziert wird. Das Protokoll zielt zudem darauf ab, robust gegen kooperierende Angreifer und gegen Missbrauch des Protokolls selbst zu sein. Die Leistung des Protokolls wird sowohl unabhängig mit zeitbasierter Minderung als auch in Verbindung mit einem zusätzlichen Minderungssystem auf der Grundlage der Auswertung empfangener Beacon-Daten bewertet. Um die Gültigkeit der erhaltenen Ergebnisse sicherzustellen, analysiert und erklärt diese Arbeit auch mehrere Probleme der Platoon-Simulationsumgebung PLEXE, die während der Studie verwendet wurde, und zeigt Fälle auf, in denen unerwartete Ergebnisse erzielt wurden. Diese Diskrepanzen können in großen Teilen auf die Implementierung bestimmter Controller innerhalb von PLEXE zurückgeführt werden und beleuchten damit potenzielle Bereiche zur Verbesserung von Simulationsumgebungen für eine genauere Bewertung von Angriffen auf Platoonprotokolle. In umfangreiche Tests in verschiedenen Angriffsszenarien zeigt PRIME seine Fähigkeit, die Stabilität des Platoons nach der Erkennung eines Angriffs wiederherzustellen und gibt damit vielversprechende Aussichten zur Verbesserung der Sicherheit von Platoons. Die Ergebnisse dieser Studie deuten darauf hin, dass mit PRIME, in Kombination mit einer schnellen und zuverlässigen Angriffserkennung, alle simulierten Angriffe effektiv abgewehrt werden können und der Angreifer hinreichend isoliert werden kann, um zukünftige Störungen zu verhindern.

Schlüsselwörter

V2V Sicherheit, CACC, Platooning, Fahrzeug-Netzwerkprotokolle, Angriffsmitigierung

Acknowledgments

This thesis would not have been possible without the continuous support and guidance of the people around me. First and foremost I would like to thank my supervisor Konstantinos Kalogiannis, who spent countless hours reviewing, supporting and questioning my research. His patience and mentorship have been instrumental in helping me navigate through the challenges encountered along the way. I also want to thank Prof. Panos Papadimitratos of the Networked Systems Security (NSS) group at KTH, who not only provided valuable insights, but also guided the process by posing questions, which played a crucial role in shaping the design of the system.
Finally, I want to express my deepest gratitude to my friends and family who helped me though this complicated time and encouraged me along the way.

Stockholm, May 2024
Michael C. Hartmann

Contents

List of Figures

List of Tables

Listings

List of acronyms and abbreviations

ACC	Adaptive Cruise Control
ART	Acceptance Range Threshold
BDL	Bidirectional Leader Topology
CACC	Cooperative Adaptive Cruise Control
CAM	Cooperative Awareness Message
CNN	Convolutional Neural Network
CTH	Constant Time Headway
CVS	Constant Vehicle Spacing
DENM	Decentralized Environmental Notification Message
DOS	Denial Of Service
ETSI	European Telecommunications Standards Institute
ITS	Intelligent Transportation Systems
LTCA	Long Term Certificate Authority
MDS	Misbehaviour Detection System
PCA	Pseudonym Certificate Authority
PKI	Public Key Infrastructure
PRA	Pseudonym Resolution Authority
SSC	Simple Speed Check
V2V	Vehicle To Vehicle
V2X	Vehicle To Everything
VANET	Vehicular Ad-hoc Network
VPKI	Vehicular Public Key Infrastructure

Chapter 1

Introduction

1.1 Background

With a rising interest in autonomous vehicles, as well as smart cities and interconnected infrastructure, new opportunities for safer, quicker and more reliable transportation are increasingly explored [1]. Several concepts from the area of Intelligent Transportation Systems (ITS) have shown promising capabilities to improve upon conventional traffic management and create a more efficient and safer environment [2]. While ITS are not necessarily limited to a specific mode of transportation, in the context of this work it will mainly refer to systems operating on the road network. ITS have been present on roads for a long time in the form of navigation systems or traffic light control systems. However, in recent years the aspect of wireless communication has come to the forefront in this area. Utilizing the communication capabilities of modern vehicles and roadside infrastructure has opened up new possibilities to enhance transportation systems. The possibilities range from notifying drivers or autonomous vehicles about approaching emergency vehicles [3], over warning about unsafe conditions like stationary vehicles [4] to cooperative maneuvering of autonomous and semi-autonomous vehicles [5].

Specifically the concept of platooning holds the promise of leveraging the autonomous driving and communication capabilities of modern vehicles to improve upon several critical aspects of vehicular transportation. Platooning is built on the premise that cooperative vehicles can reduce the safety gap between themselves and their immediate predecessor by coordinating their driving behaviour with each other. Sharing information about a vehicle's status, allows following vehicles to respond quicker to changes in speed or acceleration than if they would rely solely on their own sensor readings to estimate these parameters, therefore significantly reducing the brake lag [5]. In its essence, platooning improves upon the autonomous driving controller known as Adaptive Cruise Control (ACC). Since it adds cooperation via message passing to the otherwise independent driving, the platooning controller is called Cooperative Adaptive Cruise Control (CACC) [6]. CACC controllers have the capability to send and receive messages from other CACC controllers, leading to the formation of tightly packed vehicle convoys. Eliminating human error as well as long reaction times not only enhances the safety of the involved vehicles and passengers, it also makes traveling more comfortable and reduces inefficient braking and maneuvering. From a sustainability perspective a noteworthy benefit is the reduction of fuel consumption for platooning vehicles lowering the cost of operation while also reducing harmful emissions [7]. A vehicle platoons structure and formation can vary. While there are many possible forms of platoons for different purposes, for this work the formation under investigation will be a straight convoy where all vehicles, except for the leading vehicle, follow their predecessor on the same lane.

ITS and its sub-systems, including the concept of platooning, can be suspect to mechanical or software failures which introduce some uncertainties. In addition to that, there is always the risk of intentional attacks with the aim to disrupt or destroy traffic flow, or even to target individual entities. The specific system under investigation for this work will be the networking between vehicles, since it is the core of the platooning functionality. The vehicular communication in platoons is based on the IEEE 802.11p standard, which allows message passing between high speed vehicles on a dedicated 5.9 GHz channel [8] and can largely be narrowed down to two kinds of messages. The first kind is the Cooperative Awareness Message (CAM) [9] which is the backbone of the platoon functionality. This message, usually transmitted as a beacon with a 10Hz frequency, can contain information about the current speed, acceleration and position of the vehicle. This beacon makes it possible for other connected vehicles to adjust their own behaviour in accordance to the information they receive to keep a stable distance. The second kind of message is the Decentralized Environmental Notification Message (DENM) [9]. This message is only sent when triggered by an event, to share information about environmental influences, e.g., dangers or traffic congestion, with other vehicles. Considering the importance of correct messages in this context, the security requirements for V2X communications are very strict. All messages need to be unaltered and properly authenticated in order to prevent impersonation and falsification of messages in transit [10].

Protecting messages in transit, however, is not a guarantee for their content to be non-malicious. Malicious or compromised authenticated entities can still send false data with the explicit goal to introduce instability or even provoke collisions [11, 12, 13].
These internal attackers pose a significant threat, since other traffic members are reliant on the truthfulness of their communication. The amount of instability a internal attacker can introduce into a platoon can be quantified by how many vehicles are negatively affected by the misbehaviour and how strongly they can be made to deviate from their intended behaviour. This means that the attack capability of an attacker is directly correlated with their position inside the platoon. Under the assumption of a non-malicious leader in a platoon where each vehicle listens to messages by their direct predecessor, the strongest attacking position would therefore be the second place, since everyone, except for the platoon leader, will need to react to the misbehaviour to a greater or lesser extent. Since instabilities are attenuated downstream in a platoon, the strongest and most direct effects will affect the direct follower of the attacking entity. The task of handling the threat of internal entities disrupting the platooning functionality is complex, since often the lines between intentional misbehaviour and noise- or error induced instabilities can be blurred. This is why resilient detection and mitigation systems are required to prevent attacks on vehicle platoons.

1.2 Problem

While enhanced safety is an important design goal and often proclaimed benefit of platooning, the usage of networked systems introduces several security issues which, if abused, might lead to instabilities or even collisions between vehicles [11]. Since the comparatively smaller inter vehicle distances vastly reduce the response time, platoons must be capable of reliably and quickly responding to and mitigating misbehaviour. Misbehaviour mitigation strategies often face a trade-off between safety and stability. Decreasing the platoons sensitivity to misbehaviour can make it more stable since minor sensor errors and negligible measuring differences will not have an impact on the driving behaviour. However, decreased sensitivity can potentially enable an attacker to make gradual changes to the platoons behaviour that go undetected until it is too late to effectively mitigate the damage. On the other hand a too strong focus on safety over stability, usually meaning the immediate fallback to ACC in case of detected misbehaviour, can quickly lead to the loss of many of the platooning benefits, especially considering the prevalence of false positives in detection systems. The question that arises here is, whether a mitigation strategy can lead to a safe exclusion of an offending vehicle without running into the risk of sacrificing the platoon's own stability in the process.

1.3 Purpose and goals

The purpose of this work is to investigate the feasibility and efficiency of a new type of misbehaviour mitigation strategy for platooning vehicles. The strategy in question has the potential to weaken and exclude offending vehicles from platooning in a way that eliminates them as a threat to the platoon. This would enable vehicles to resume platooning and restore the stability of their convoy even in the continued presence of adversarial entities. Considering the environmental and economical benefits of platooning, this strategy could help to retain them by securing this mode of operation. Additionally, by mitigating threats to moving vehicles, this approach could contribute to safer road conditions and prevent potentially lethal collisions. The main objective of this thesis is to describe, test and evaluate this novel mitigation system, which allows a physical restructuring of the platoon formation to expel and isolate attackers from dangerous positions. The system in question is designed to handle attackers who tamper with the communication systems as well as attackers who aim to abuse the restructuring mechanism of the mitigation protocol itself to further their position in the platoon.

The goal of this project is to improve upon several shortcoming in currently proposed misbehaviour mitigation systems. This goal can be subdivided into the following more specific sub-goals:

1. Eliminate the threat of an attacker remaining inside a platoon formation after being detected as misbehaving
2. Reduce uncertainty about maliciousness by providing attackers with a challenge to either admit to misbehaviour or relinquish their position
3. Create a platoon restructuring system which cannot be abused by an internal attacker
4. Regaining full platoon stability after a successful execution of the protocol

1.4 Research Methodology

This work uses a largely quantitative approach to the topic. Similar to other research conducted in this area, the project is mostly focused on obtaining measurable performance data from a simulated platoon to quantify the effects of various attacks with and without the misbehaviour mitigation system in place. Additionally, this work aims to qualitatively compare and discuss the benefits of the proposed system in relation to other work in this area.

1.5 Ethical considerations

Vehicular transportation, be it of humans or of goods, is a vital aspect of modern society. Disruptions on the road network can not only lead to delays in supply chains and public transportation, but can also lead to the congestion of essential transportation routes. In the worst cases attacks on ITS could directly lead to accidents and potentially even casualties, which is why securing these systems is imperative. The system proposed in this work aims to neutralize attackers inside a platoon, which could, if left unchecked, provoke mass collisions with multiple involved vehicles. To safeguard not only human life but also the vital importance of our road network to societal stability this work aims to do its part by proposing a more secure and sustainable approach to misbehaviour mitigation.

1.6 Delimitations

In a typical traffic scenario, the human factor can significantly influence outcomes. However, in the context of this work, the human factor has been omitted. The assumption is that all vehicles operate under the guidance of autonomous driving controllers, and there is no external traffic present outside the platoon. This deliberate choice focuses the investigation solely on the technical aspects of the system. Consequently, the findings may not guarantee replicability under realistic traffic conditions. The simulated vehicles used to derive the results are considered uniform in size and engine capabilities. This assumption simplifies the scenario, as variations in factors such as acceleration capabilities can influence platoon performance in a realistic setting.

1.7 Structure of the thesis

The thesis project is structured as follows: Chapter 2 introduces several core systems which are present in the platooning environment. The chapter details general ITS technologies as well as several platooning specific systems. Chapter 3 gives a clear definition of the applied methodology and describes the various assumptions made in the context of this work. It also describes the specifics of the used simulation environment and the implementation of the simulated scenarios. In Chapter 4 the results of the conducted experiments are shown and thoroughly examined. The results are discussed in depth and the findings are compared to the goals which have been set for the new mitigation system. Finally Chapter 5 will summarize the findings and draw final conclusions based on the obtained results. Here the identified shortcomings and observed limitations are discussed and formulated into an outlook for future research into this area.

Chapter 2

Background

For the successful implementation of platooning among connected vehicles, the establishment of a robust ecosystem involving communication standards and connected infrastructure is imperative. This chapter aims to provide the reader with a comprehensive understanding of the systems integral to this process. The initial focus in Section 2.1 is on describing the information exchange between vehicles and infrastructure in Intelligent Transportation Systems (ITS). It will delve into the transmission and protection of their communications, emphasizing the techniques employed to ensure the security and privacy of these interactions. In Section 2.2, the scope narrows down to the specific ITS application of platooning. This section will introduce the structure of a platoon and outline commonly used communication topologies. Furthermore, it will provide a concise overview of various maneuvers that a platoon can execute, relevant to the context of this study. Given the diversity in implementing platooning functionality, Section 2.3 will review the platooning controllers utilized in this thesis. Finally, Section 2.4 will explore previous academic work in this domain.

2.1 Secure and Private Vehicular Communication Systems

Vehicular Ad-hoc Networks VANET are among the most promising mobile ad-hoc network applications [14, 15]. Vehicles in a VANET are equipped with sensor systems and radio interfaces which allow them to detect dangerous situations and notify other vehicles in their local area [16], or to broadcast their status in order to allow other vehicles to react accordingly, allowing for better safety and more informed decision making [17]. There are mainly two kinds of messages that are exchanged between the connected vehicles in the ITS. First, the Cooperative Awareness Message (CAM). These beacons can contain information about the current speed, acceleration or direction of the sending vehicle. This data is frequently changing, therefore the beacons are disseminated with a 10 Hz frequency, which is every 100 ms. CAM can also be used to transmit data like the vehicle role which is not as urgent and therefore only sent every 500 ms [8]. The other frequently used beacon is the Decentralized Environmental Notification Message (DENM). This kind of message contains information that is relevant to the local environment which is propagated throughout the network of connected vehicles and infrastructure in a multi-hop way. This message can warn of environmental hazards like accidents, hazards on the road or traffic jams [8].

2.1.1 Identity and Credential Management

There are many risks involved in ITS which could potentially allow attackers to disrupt the system if left unchecked. To ensure the security of the communication and to preserve the privacy of its participants, communication standards by the the European Telecommunications Standards Institute (ETSI) [18], the IEEE 1609.2 [19] as well as the newer 3GPP TS 33.185 [20] describe a Public Key Infrastructure (PKI) which allows vehicles and infrastructure to communicate. The PKI for this purpose is called a Vehicular Public Key Infrastructure (VPKI) [21, 22], as there are a few differences and context specific challenges which are not present for a conventional PKI. In a VPKI the vehicles are issued two kinds of certificates for identification - long-term certificates which provide an identity within the system and short-term certificates which are used as pseudonyms to sign messages. The first step to acquire an identity within the VPKI is to register with a Long Term Certificate Authority (LTCA) [22]. The certificate issued by the VPKI can then subsequently be used to authenticate with a local Pseudonym Certificate Authority (PCA), which then issues a pool of short-term certificates, or pseudonyms, which the vehicle can use to sign their messages without exposing their true long-term identity. The short-term identity is regularly changed to prevent tracing of the vehicles journeys in order to preserve its privacy [23, 24, 25, 26]. The use of these cryptographic measures allow for privacy towards other vehicles and potential eavesdroppers, while not violating non-repudiation in case of misbehaviour. In the case of misbehaviour the violating entities pseudonym can be resolved by the Pseudonym Resolution

Authority (PRA), an entity separate from the LTCA and the PCA which queries their information to link the long-term and short-term identity of an offending vehicle [27].

2.1.2 Security

The VPKI itself can greatly increase the security of the network by fulfilling three main security requirements:

1. Authentication and communication integrity
2. Authorization and access control
3. Non-repudiation, accountability and eviction

The structure of the VPKI allows only correctly registered vehicles with valid long-term identities to participate in the communication therefore preventing impersonation attacks on the network [28]. Ensuring communication integrity in vehicular communication is a challenging task due to fluctuating neighbour density and the delay it adds to the processing of each beacon. To solve this issue, the task to verify incoming beacons can be shared among neighbouring vehicles. This is especially helpful in preventing denial of service due to clogging of the communication [29]. The non-repudiation as well as the accountability requirement can be fulfilled due to the PRA. In case sufficient evidence, in the form of incriminating messages, is collected, this entity can resolve an offending vehicles pseudonym and reveal its long-term identity for revocation purposes, thus excluding misbehaving vehicles from the network. Another important requirement of the VPKI is that it needs to ensure that at any given time no vehicle is in possession of multiple valid pseudonyms [30]. This would enable malicious actors to mount sybil attacks [31] in which they could simulate several vehicles at the same time, allowing them to disrupt the system e.g. by creating phantom traffic jams [32]. Since the ticket or token used to obtain the pseudonyms from the PCA is signed with the private key corresponding to the vehicles long-term certificate it can be verified that no two pseudonyms acquired with the same ticket are used simultaneously [33].

2.1.3 Privacy

A vehicular communication system like this cannot be deployed unless the security and privacy of its users are ensured [34, 35, 36]. Even benign network nodes must be assumed, at best, honest-but-curious. This means that, while adhering correctly to all protocols, the entities will try to acquire as much personal data as possible in the process, possibly violating the privacy of other entities. This has to be assumed as true, since all entities of the network, including infrastructure, potentially gather data for their own purposes [37]. While complete anonymity cannot be provided in the VPKI, due to the accountability requirement, each vehicle should only reveal the minimum necessary information in V2X communication to ensure unlinkability. First, the LTCA is not supposed to know which PCA the vehicle seeks to obtain pseudonyms from when issuing a ticket. Therefore, the vehicle conceals the targeted PCA during the ticket request only sharing the timeframe for which it needs the pseudonyms with the LTCA. Secondly, the PCA and other vehicles or infrastructure should not be able to infer a vehicles long-term identity from their communication with it. For this purpose the ticket instead of the long-term certificate is used to authenticate to PCA while all other network members are exclusively contacted with the pseudonyms obtained from the PCA. Lastly, it should not be possible to link separate pseudonyms as belonging to the same entity. As a first step to fulfill this requirement, pseudonyms come with a limited time-to-live. This prevents predictable pseudonym changing patterns which would allow an observer to create pseudonym profiles that could, e.g. due to their starting point, reveal a likely home address [38]. However, even frequent pseudonym changes cannot prevent an observer with the capability of tracking all messages from linking pseudonyms together. The predictability of movement on a road network allows an observer to estimate a probable next position for a vehicle [38]. This makes it possible to track a vehicles pseudonym changes with a high likelihood. To avoid linkability in this context there have been several different proposals. The physical aspect of a VANET implies that vehicles are limited to movement which is both physically possible and coherent with traffic rules. As shown above this makes tracking of vehicles, even with pseudonyms, considerably easier. One proposal to enhance unlinkability is therefore to change pseudonyms in areas where this predictability of movement is not given, e.g. in intersections. These areas are then referred to as mix zones [39, 40, 41]. Within these mix zones it is considerably harder for an observer to link pseudonyms. A problem with the mix zone approach is that they only serve their purpose if a sufficient amount of vehicles is present during the pseudonym change time. To mitigate this limitation it has been proposed to create simulated traffic, so called ”chaff” vehicles, in these zones. To do this, RSUs in the area sign and transmit messages in the mix zone with pseudonyms to limit the observability of real vehicles [40, 41]. In [42], the authors propose a system of swappable and non-swappable pseudonyms for mix zones to further confuse possible observers. Their approach makes it possible for vehicles to swap their pseudonyms with each other while travelling in a mix context. Consequently one pseudonym can represent a different vehicle before and after the swap, making pseudonyms reusable and increasing the entropy of the mix zone.

2.2 Platooning

Platooning is a Vehicle Ad-Hoc Network (VANET) application, intending to capitalize on the communication capabilities among vehicles to enhance their driving behavior. When vehicles share a common route or a portion of it, they may choose to travel together and collaborate in their driving actions. This collaborative approach has the potential to significantly enhance various aspects of road mobility. Through cooperation, vehicles can optimize road usage by minimizing the gaps between them. The communication of acceleration and braking actions allows for a substantial reduction in reaction time. As highlighted by [5], this reduction in gaps not only improves road efficiency but also has the added advantage of decreasing air resistance for following vehicles. Consequently, this reduction in air resistance contributes to lower fuel consumption and, by extension, results in reduced harmful emissions [7].

2.2.1 Platoon structure

A platoon consists, in general, of a leader vehicle, which makes the driving decisions, and several follower vehicles, which adapt their driving behaviour according to the messages transmitted to them. The exact communication structure, however, can vary. The aim of the platoon is to achieve string stability [43]. String stability implies that distance errors between the vehicles decrease as they propagate through the platoon, meaning that a distance error, introduced at index i in the platoon, affects each vehicle at $j > i$ to a lesser degree the further j is from i in the platoon formation. To achieve this, every vehicle in the platoon needs to be capable of keeping the desired inter vehicle distance based on its own status and the messages it receives from its platoon members. The information flow topology of a platoon dictates with which platoon-members each vehicle communicates to coordinate their behaviour. Limiting the communication to only the nearest neighbour, i.e. the predecessor, has been proven to be unable to fulfill the string stability requirement when operating with the goal of keeping a constant spacing within the platoon [44]. To mitigate this, Darbha et al. proposed that one vehicle, usually the leader, needs to be able to communicate with a large number of platoon members to make it string stable and scalable [44]. The simplest topology fulfilling this is therefore a Predecessor-Leader following topology. In this topology each platoon member listens to the messages of their direct predecessor and the broadcasts of a globally reachable platoon leader to coordinate their behaviour. There are several variations of this topology, including topologies with several predecessors and bidirectional variations which allow communication with the follower. The performance of these topologies is more closely described and analyzed in [45]. Here the authors propose that a Bidirectional Leader Topology (BDL), i.e. a topology with bidirectional communication between predecessor and follower plus a globally reachable leader, provides the best internal stability and scalability.

2.2.2 Maneuvers

Platoons are not static entities. Over the course of a platoons lifetime, from its formation to its dissolving, members might exit the platoon at various positions or new members might join it. The focal point of platooning maneuvers is the platoon leader. Upon the formation of a platoon, one vehicle is determined to be the platoon leader. The leader is not only in charge of dictating the platoon driving parameters, but also acts as the coordinator of platoon maneuvers. Vehicles that desire to join or leave the platoon need to send the respective requests to the platoon leader in order to have them approved or denied and to have the desired maneuvers initiated. A platoon leader needs to coordinate the platoon response to these requests in a way that ensures the platoons prolonged stability. This means that the platoon leader will, for example, not accept maneuver requests while the platoon is currently performing another maneuver. A leader will also not allow the platoon to grow past a predetermined size, denying all join requests that would lead to an exceeding of this value [46]. If a platoon is not currently conducting a maneuver and of a size smaller than its maximum it will accept new vehicles to join. A potential Joiner will send a request to the platoon leader awaiting a response whether or not a join at the indicated position is possible. There are two possibilities for the join maneuver, depending on the position the Joiner requests. If the indicated position is the rear of the platoon, the Joiner needs to align itself behind the tail vehicle of the platoon upon receiving the confirmation that the join is accepted. The Joiner needs to accelerate to catch up with the platoon and upon reaching the desired position notify the platoon leader that it is in position ready to receive the platoon properties. The leader will then provide the required information, allowing the Joiner to change his operational mode to CACC. As a last step the leader will send a broadcast to the entire platoon, notifying them about the updated platoon formation. The second variant of the join maneuver is more complicated. Should a middle join be accepted, the Joiner will need to align itself with the platoon at the desired index while travelling on the neighbouring lane. Upon notifying the leader that it is in position and ready to join, the leader will send a message to the vehicle currently occupying the desired position within the platoon. The vehicle is requested to increase the inter-vehicle distance to its predecessor in order to create a gap in the platoon large enough to fit the Joiner. Upon receiving a confirmation about the completion of this task, the leader will notify the Joiner that it is now safe to change to the platoon lane. When the Joiner reached the desired position it will notify the leader who then broadcasts the new formation to the entire platoon to enable the Joiner to change to CACC and allows the vehicle now following the newly joined vehicle to revert to the normal inter-vehicle distance. While it seems counter-intuitive and complicated to allow middle-joins, research has shown that middle-joins can actually improve platoon stability [47]. An exit maneuver functions similar to the join maneuvers. The vehicle intending to leave the platoon will request the maneuver from the leader. Upon receiving confirmation it will change lane to leave the platoon and change its operational mode. The leader will update the formation and broadcast this information to the remaining platoon members.

2.3 Controllers

The functionality of CACC can be implemented in several different ways. For this work the four controllers, implemented in the most commonly used simulation tools, will be used. In general vehicles in CACC rely on four different kinematic parameters to adjust their driving behaviour: Distance, Acceleration, Speed and Position. However there are several CACC controllers which differ in the way they obtain and use this data. The default implementation of CACC in the SUMO environment is the PATH controller [48]. This controller adheres to a Predecessor-Leader topology and implements a Constant Vehicle Spacing (CVS) policy, meaning that it tries to keep a predetermined fixed distance to its predecessor. It obtains speed and acceleration information from both the predecessor and the leader and relies very little on its own sensors. Another CVS controller is Flatbed [49]. The main difference between Flatbed and PATH is that Flatbed obtains only the speed information from the leader and the predecessor, giving the latter a higher priority when computing their own acceleration. It reduces the required external information and gives its own sensors a higher weight when compared to PATH. Opposed to the CVS policy, it is also possible to implement the intra-platoon distance via a Constant Time Headway (CTH) policy. With this policy the distance between the vehicles is dependent on the speed with which the platoon is travelling. The goal is to keep the distance based on the time it takes a vehicle to traverse the distance between its front bumper and the predecessors back bumper. This implies that the gap gets larger with higher speeds and shrinks at lower speeds. One controller that employs this policy is the Ploeg controller [50]. This controller uses a predecessor following topology, which, as mentioned earlier, leads to string instability with larger platoon sizes. The Ploeg controller only obtains acceleration data from its predecessor and, in case of disturbances, predicts the value should it not receive a beacon in time. The last controller evaluated in this work, is the Consensus controller [51]. This controller is very versatile in the sense that it supports multiple topologies and can work with both CTH and CVS policy. Per default it will also use a Predecessor-Leader topology.

2.4 Related Work

In platooning systems, the reliability and availability of transmitted beacons as well as strict adherence to established protocols by all members are crucial for seamless operation. This research primarily focuses on understanding how a platoon can operate when faced with an attacker or misbehaving entity, with a specific emphasis on mitigating the negative impacts of diverse attacks. This section reviews existing studies that explore potential security challenges and attack vectors within the platooning environment. It also discusses a range of proposed techniques aimed at detecting misbehaving entities within platoons and distinguishing them from normal benign members. Additionally, various misbehaviour mitigation systems are examined, each offering solutions to protect both platoons and individual vehicles from the adverse effects of attacks.

2.4.1 Attacks on Vehicle Platoons

Platooning systems, particularly the networking dynamics between vehicles, are susceptible to an array of attacks. Among these, common network-based attacks, including Denial Of Service (DOS), can be orchestrated by either a platoon internal attacker or an external adversary. Such attacks may involve tactics like clogging the networking channel with bogus messages [52] or outright jamming the communication links [11] to prevent the vehicles from receiving beacons. Jamming attacks, in particular, aim to disrupt the communication between platooning vehicles, thereby compromising the overall functionality and effectiveness of the platoon. Additionally, these attacks could be precisely timed to prevent specific messages, such as those related to maneuvers, creating potentially hazardous situations [13, 53]. DOS attacks, if left unmitigated, can induce severe instabilities within the platoon, especially if sustained over an extended duration and in scenarios where a quick and accurate response to messages is vital. While cryptographic measures are typically employed for authentication purposes to prevent external interference, they may fall short in safeguarding against internal adversaries who falsify their messages [54, 11]. A malicious platoon member could manipulate parameters such as their speed, acceleration, or distance, injecting instability into the platoon [52, 11, 54]. In the absence of adequate mitigation techniques, this instability could quickly escalate to collisions with potentially fatal consequences [12]. Beyond the intrinsic safety measures, the effectiveness of an attack is dependent on various factors. The distance between vehicles and their traveling speed directly influence the time available for reaction before potential collisions with neighboring cars during an attack. Equally significant is the attacker's position within the platoon. In a predecessor-leader topology, falsified messages predominantly impact following vehicles, yet they having no direct consequences for those positioned in front of the attacker. Consequently, the attacker's position is a pivotal factor determining the extent of the platoon affected by the attack. Of particular concern is the potent attack position assumed by the platoon leader. Architectures reliant on a globally reachable leader, such as the predecessor-leader topology explained in Section 1.1, render a malicious leader capable of influencing the entire platoon. This not only induces potentially dangerous instability but also poses a greater challenge in terms of early detection and mitigation [55]. Another dangerous attack scenario demonstrated in [13, 56], shows that attackers can exploit specific situations, such as maneuvers, to amplify the impact and efficacy of their attacks.

2.4.2 Misbehaviour Detection Mechanisms

Detecting misbehaviour among the members of a platoon is a challenging task. The most straightforward way to detect misbehaviour is to perform a plausibility check. This mechanism will identify messages that contain nonsensical, impossible or highly implausible contents. This means that messages containing values which lie outside of a certain deviation range can be immediately discarded. Mechanisms like the Simple Speed Check (SSC) and Acceptance Range Threshold (ART), as described in [57, 58], are examples for these plausibility checks. While plausibility checks can vastly limit the attackers capabilities by constraining the falsification possibilities to realistic movement, they are not sufficient to prevent more carefully crafted attacks which aim to mimic realistic behaviour. By gradually changing values in their falsified messages, an attacker can circumvent simple detection mechanisms while still inflicting instability upon the platoon [59]. To detect more sophisticated attacks and the vehicle conducting them, Khanapuri et al. [60] propose to use a deep learning approach. The Convolutional Neural Network (CNN) proposed in their work, coupled with several pre-processing techniques, achieves even in noisy environments accuracy levels of up to 96.3%. This model, like many detection techniques in the field of platooning, assumes the attacker is not the platoon leader. Another technique that makes use of a benign leader as a reference point is by Bernard et al. [61]. The detection of misbehaviour works by computing a reputation score for each vehicle in the platoon which can be shared among the platoon members. The larger and more frequent a vehicle deviates in its behaviour from the messages sent by the platoon leader, the lower their reputation becomes. The requirement of having a trusted party i.e the platoon leader, limits the capabilities of these detection mechanisms. Sun et al. [62] therefore propose a misbehaviour detection system which detects false data based on secure sensing mechanisms. The system makes use of the angle of arrival and the Doppler speed to physically verify the contents of a message and an extended Kalman filter to track the vehicles' true but unknown state. While this mechanism does not rely on a trustworthy leader or a honest majority inside the platoon it makes the assumption of having at least one honest neighbour, i.e. a directly neighbouring platoon member. This honest neighbour is needed only to aid in reliably tracking the misbehaving vehicle by sending an indirect measurement of the offending vehicle to the vehicle tracking the supposed attacker to allow an estimation of the true acceleration and position. This system, however, only works in mostly straight highway scenarios without sharp turns. The detectability of misbehaviour can vary widely depending on the state the platoon is in. As shown by Kalogiannis et al. [13], many current detection mechanisms cannot reliably detect attacks or misbehaviour conducted during maneuvers e.g. join- or leave-maneuvers. For this purpose the authors propose the usage of a Hidden Markov Model, which allows for the combination of maneuver- and misbehaviour detection to obtain reliable results during maneuvering. Kamel et al. [63] have proposed several machine learning based classifiers which, trained on features extracted from several plausibility checks, can distinguish misbehaving vehicles from benign ones. However, the authors system is not necessarily exclusive to platooning but rather considers a general V2X scenario and has not been evaluated for a platooning scenario.

2.4.3 Mitigation Techniques

Upon the detection of anomalous behaviour, the next step is to adequately mitigate the negative effects the misbehaviour would bring to the platoon. The goal of misbehaviour mitigation is first and foremost the prevention of damage i.e. collisions. However, an important secondary goal is to preserve the platoon stability and therefore the benefits platooning brings. This means in turn that the reaction to perceived misbehaviour should not be stronger than necessary. The naive solution to preventing damage upon detection of misbehaviour would be to revert from CACC back to ACC. This means effectively stopping the platooning behaviour completely, which while usually effective, is an inefficient approach to mitigation [64]. This solution has several drawbacks which show that a more adaptive solution is required. Firstly, misbehaviour detection systems are not infallible, this means that there are still false positives which could trigger the mitigation behaviour in harmless situations. By directly reverting to ACC the platoon cohesion is immediately broken, an outcome which in turn means that all platooning benefits are lost for the sake of safety. Secondly, breaking the platoon immediately when misbehaviour is detected allows for cheap DOS attacks, which aim to abuse this mitigation mechanism to disrupt the platoon [64]. On the other side of the spectrum, a mitigation technique which puts more emphasis on the platoon cohesion would be what can be broadly classified as *drop false messages* as described by Wolf et al. [64]. This, as the name implies, means to simply disregard and drop messages which are considered false. While this would allow for strong platoon cohesion, it also implies that the vehicle is operating blindly, or more accurately only limited to its own sensors until correct messages arrive again. A behaviour which is inherently unsafe and quickly becomes unstable if no correct messages arrive [64]. To allow for a more adaptive behaviour which aims to provide a more reasonable trade-off between safety and platoon cohesion Wolf et al. [64] propose a suspiciousness based mitigation. This means that instead of directly reverting back to ACC the controller of the vehicle behind the attacker computes a suspiciousness parameter based on the magnitude of the deviation from the leader behaviour and the previous suspiciousness of the offending vehicle. This parameter is then used to decide how large the safety distance between follower and predecessor should be, the largest possible value being the ACC distance. By adjusting the strength of the response to the strength of the attack, minor sensor errors and minuscule misbehaviour will not directly trigger a dissolution of the platoon, preserving the platoons cohesion. Only prolonged or strong misbehaviour will lead to the, in this case reasonable, degradation to ACC. Sajjad et al. [65] propose a mitigation scheme for a bidirectional platooning topology. The mechanism works by switching from the bidirectional topology to a unidirectional topology upon detection of misbehaviour. This entails that all cars behind the attacker will unidirectionally focus on frontal collision avoidance, while all cars in front of the attacker will focus on rear collision avoidance. As pointed out by the authors this approach ensures safety by preventing both frontal and rear collisions, yet, it also gives the attacker significant control over the platoons movement and should therefore not be used as a stand-alone solution. Another approach to both detect and mitigate attacks using a set-membership filter has been proposed by Mousavinejad et al. [66]. By using previously obtained measurements and state information, each vehicle can predict a set of next states of their predecessor vehicle. Should the predecessor display behaviour that is not corresponding with the state it disseminates via its messages, the state will not intersect with previously computed set of predicted states meaning that the received signal must be incorrect. In this case the mitigation mechanism will use the state estimation rather then the communicated state to adjust the vehicles behaviour. Additionally, if there is no intersection between the estimated state based on current measurements and the predicted states, the mitigation mechanism assumes that the sensor data is compromised and will rely on its state prediction rather than its estimation to adjust the vehicles behaviour. This approach, however, will not be efficient against an attacker who hides their falsification attempts by gradually falsifying within the assumed noise boundaries. Kamel et al. [63] state that the reporting of misbehaviour plays a crucial role during the mitigation process, since individual vehicles do not have the capability to revoke certificates. Yet, reports in this context need to have their validity ensured, since trust in the integrity of the vehicle reporting the misbehaviour cannot be freely given. One way in which this problem has been approached, is by introducing a certain cost for the reporting vehicle to discourage abuse of the reporting function. Using a so called ”suicide protocol”, both the accuser and the accused would be removed from the network when misbehaviour is reported [67]. The gathering of sufficient incriminating information for the revocation authorities without continued risk of exposure to falsified messages is the main way to prevent misbehaviour in the long-term. However, immediate actions need to be taken to ensure the safety in the presence of an attacker. This work aims to provide a solution which can bridge this gap and secure a platoon after it has been faced with adverse behaviour.

2.5 Summary

As this chapter has shown, the concept of platooning is still very much a work in progress. To adequately mitigate attacks on the platooning system, strategies have to hold the balance between efficiency in retaining the benefits of platooning and safety. The above described detection and mitigation strategies have the major drawback that there oftentimes is a grey area where it becomes complicated to determine whether a deviating vehicle is truly misbehaving. In current literature it is assumed that misbehaving vehicles will be reported in order to have their certificate revoked. Especially in the case of false positives this is a costly and inefficient approach to handle deviating platoon members. The persistent risk in many here described mitigation strategies is that the attacker is either isolated inside the platoon, creating a possibly dangerous sensor blind spot for other followers, or the vehicles will to some extent seize to utilize the CACC functionality, sacrificing the platooning benefits. A mitigation strategy, bridging the time between detection and possible revocation, which aims to decrease the threat a possible attacker still poses will be discussed in the following chapters.

Chapter 3

Methodology

To simulate and assess the impact of attacks on a platoon incorporating the proposed mitigation system, certain assumptions must first be established. This chapter aims to outline the overarching assumptions regarding the system and the entities within it, with a particular focus on the adversary. The subsequent section will delve into the specifics of the implementation of the attacks, detection mechanisms, and the mitigation techniques employed in the conducted simulations. It will provide a detailed account of the strategies utilized to simulate and evaluate the system, shedding light on the setup for the presented scenarios. Furthermore, the chapter will elaborate on the tools used for simulation and evaluation, offering clear insights into the configuration of simulations. Concluding this section, there will be a discussion on how the data obtained from the simulations has been analyzed.

3.1 System Model

3.1.1 Assumptions

This work establishes several fundamental assumptions for the simulation environment. It assumes the presence of a multi-vehicle platoon situated in a straight highway scenario featuring multiple lanes and devoid of any external traffic, excluding the platooning vehicles themselves. Additionally, each vehicle within the platoon is considered identical in terms of kinematic properties. The platoon is introduced into the scenario in a steady state, meaning that the completion of the platoon formation process has been reached and each vehicle is at the desired distance to their predecessor. Each member is assigned the correct velocity, acceleration, and position to maintain an appropriate distance from its predecessor. Communication between vehicles is established based on the assumption that the VPKI and its cryptographic properties, as detailed in Chapter 2, are in place and appropriately configured. Every vehicle is assumed to possess valid cryptographic material enabling secure communication with other platoon members. Furthermore, it is assumed that each vehicle is equipped with sensor capabilities enabling the measurement of frontal as well as rear distances, along with the estimation of their predecessor's speed. To simulate a more realistic platoon movement, the leading vehicle will oscillate its speed by 2km/h at a frequency of 0.2Hz. These assumptions collectively form the foundational framework for the subsequent simulation and evaluation of the proposed mitigation system.

3.1.2 Adversary Model

As outlined in Section 2.4, there exists a wide variety of potential attack vectors. However, this work concentrates primarily on internal attackers executing falsification attacks, as well as DOS attacks. The adversary can be positioned at any location within the platoon, excluding the leader position. This exclusion arises from the proposed algorithm, which involves internal position changes within the platoon. A misbehaving leader cannot be mitigated in the same way since it would in turn mean that another platoon member would be elevated to leader status. A process which in itself would open up dangerous attack vectors. The attacker possesses the capability to freely manipulate the kinematic parameters transmitted via its beacons, including speed, acceleration, and position falsification. Notably, physical misbehaviour in the form of malicious driving actions is explicitly disallowed. The attacker is assumed to have no influence over the hardware within the system and cannot alter any aspect of communication content aside from the aforementioned parameters. For this work, three types of attackers are defined. The first type is a malicious misbehaving attacker, who, when detected, refuses to conform to the mitigation protocol and remains in their position despite requests to vacate. The second type is a misbehaving vehicle, whether malicious or simply suffering from technical issues, who complies when instructed to relinquish their position. The third type is an attacker seeking to exploit the mitigation protocol itself by falsely accusing their predecessor to secure a more advantageous position, specifically closer to the leader. It is crucial to note that, within the scope of this work, only the second type of attacker will be subject to quantitative simulation. The other two will be discussed qualitatively in Sec.4.3.

3.2 Implementation Details

3.2.1 Attack Implementation

The attacks, implemented for this work, come in three different variations. The first is a naive falsification attack where a constant is added to one of the kinematic parameters which is communicated to the platoon members. The falsification of the beacon is designed to manipulate the follower of the attacking vehicle to either accelerate or brake depending on the chosen value. The magnitude of the falsification can be individually chosen for all three types of attack. This kind of attack can be used to either simulate naive falsification or events like emergency braking. It is important to notice here that this attack always falsifies exactly one parameter which means that controllers, who do not use this specific parameter in their computations, will not react to this kind of attack. The second attack extends this approach by making the falsification of parameters gradual. Instead of directly adding a constant value to the communicated kinematic parameters, the attacker will slowly increase or decrease the value over the course of several messages to mimic normal acceleration or deceleration behaviour. Additionally, instead of falsifying just one parameter, the attack will change the other kinematic parameters to fit the falsified value to prevent immediate detection by systems which check for consistency between values. These more complex attacks can therefore evade detection by simple plausibility checks, which would detect abrupt, impossible or unlikely changes. Falsifying these values within the normal noise boundaries would allow an attacker to make small changes to the platoon behaviour which would go unnoticed by most misbehaviour detection systems. However, since this study is not focused on the analysis of detection systems, the chosen values have been selected to be large enough to eventually cause disturbances and in some cases even collisions at the expense of stealthiness. The exact implementation used for the simulation can be seen in Listing 3.1

```
double BaseProtocol::falsifiyData(double data)
{
    // Gradually falsify kinematic values
    data += falsificationValue;
    if(attack.compare(”ACC”) == 0)
    {
        falsificationValue -= 0.015;
    }
    else if(attack.compare(”POS”) == 0)
    {
        falsificationValue -= 2.5;
    }
    else if(attack.compare(”SPEED”) == 0)
    {
        falsificationValue -= 0.5;
    }
    return data;
}
```

```cpp
std::vector<double> BaseProtocol::combinedAttack(VEHICLE_DATA &data)
{
    std::vector<double> kinematics{ 0, 0, 0 }; // initialize empty kinematics vector
    double interval = 0.1; //interval in which beacons are sent

    if(attack.compare(”ACC”) == 0) //Gradual Acceleration attack
    {
        kinematics[2] = falsifiyData(data.acceleration); //Falsifiy Acceleration
        kinematics[1] = data.speed + kinematics[2] * interval; // Falsify Speed to match false Acceleration
        kinematics[0] = previousPosition + kinematics[1] * interval + 0.5*kinematics[2] * pow(interval,2); //Falsify position to match false acceleration
    }
    else if(attack.compare(”POS”) == 0) //Gradual Position attack
    {
        kinematics[0] = falsifiyData(data.positionX); //Falsify Position
        kinematics[2] = (kinematics[1]-previousSpeed)/interval; //Falsifiy Acceleration to match false Position
        kinematics[1] = (kinematics[0]-previousPosition)/interval; // Falsify Speed to match false Position
    }
    else if(attack.compare(”SPEED”) == 0) //Gradual Speed attack
    {
        kinematics[1] = falsifiyData(data.speed); //Falsify Speed
        kinematics[2] = (kinematics[1]-0.5 - kinematics[1])/interval; //Falsifiy Acceleration to match false Speed
        kinematics[2] = (kinematics[1]-0.5 - kinematics[1])/interval;
        kinematics[0] = falsePosition + (kinematics[1]) * interval + 0.5*kinematics[2] * pow(interval,2); //Falsify position to match false Speed
    }

    return kinematics;
}
```

Listing 3.1: Gradual attack implementation

The third attack type covered within this work is a jamming attack. In the scope of this work this attack has been implemented not by having the attacker interfere with the network communication, but rather as the victim dropping received packets after the attack starts to simulate a clogging DOS attack. This attack type has been chosen since jamming attacks on networks are, compared to the other mentioned attack types, easy to conduct, and could potentially be executed even by attackers who have no control over the vehicles on-board unit.

3.2.2 Detection Implementation

Since the focus of this work is on the mitigation of misbehaviour rather than its detection, the implementation assumes a reasonable detection system to be in place. To simulate the functionality of the detection mechanism, the main mitigation behaviour of the follower will be initiated at a predefined point in time after the first falsified message has been sent by the attacker. To cover different attack strengths as well as different detection capabilities the system has been tested with a detection time of 0.2, 0.5, 1 and 2 seconds to fall within the detection time of most misbehaviour detection systems. The detection will only be activated if the mitigation parameter has been set to *True*. Another detection mechanism used in this work is the simple detection system used by Wolf et al. [64]. This detection system compares the predecessors sent acceleration to the leaders sent acceleration and based on their difference determines whether or not the deviation constitutes misbehaviour or noise, i.e., falling within acceptable boundaries.

3.2.3 Mitigation Implementation

The misbehaviour mitigation system in this work consists of two components which can work independently or together. The first line of defence which is active already before the attack has started is the suspiciousness based mitigation strategy proposed by Wolf et al. [64]. The second component is the PRIME mitigation protocol introduced by this thesis which will only become active when activated by a MDS.

3.2.3.1 Suspiciousness based mitigation

The suspiciousness based mitigation by Wolf et al. [64] aims to provide a proportional response to misbehaviour by, instead of immediately breaking the platoon and reverting to ACC, gradually increasing the safe distance to the offending vehicle depending on how ”suspicious” its behaviour is. This system has been implemented by saving and updating the latest leader acceleration value with every received beacon. Upon receiving a beacon from the predecessor the probability p of it being malicious is computed as $p = |\frac{a_L - a_P}{a_L}|$ where a_L is the normalized leader acceleration and a_P the normalized predecessor acceleration. Based on this, the suspiciousness of the predecessor at timestep t can be computed, using the current attack probability p, a dampening factor α and the previous suspiciousness s_{t-1}, as $s_t = (1 - \alpha) \cdot s_{t-1} + \alpha \cdot p$. This suspiciousness parameter s_t can now be used to determine a new safe headway factor h by normalizing it between two predefined thresholds, $noise$ and $misb$, as $\frac{s_t - noise}{misb - noise}$. The $noise$ threshold indicates the value until which the suspiciousness is considered benign, since some fluctuations in the suspiciousness parameter are expected in a noisy environment. The misbehaviour threshold ($misb$) marks the value after which the behaviour is considered malicious, justifying a full response, which in this case means reverting to ACC. This way, upon the suspiciousness value exceeding the noise threshold, the headway factor h will gradually rise from 0 to 1 the closer the suspiciousness gets to the misbehaviour threshold. Finally, this factor is used to compute the new headway by applying the factor to a standard 2 seconds ACC headway. For CTH controllers this is straightforward, since the new headway is simply $h \cdot 2$. For CVS controllers the headway can be computed by first converting the time headway to a distance value, as $(speed \cdot \frac{1000}{3600} \cdot 2) * h$. In its original description this mitigation technique is exclusively designed and tested for the PATH controller. For this work, however, it has been adapted for other controllers as well. When run independently this mitigation system will, upon exceeding the misbehaviour threshold, simply continue at the ACC distance of 2 seconds. Should it, however, be run together with the main mitigation system, the exceeding of the misbehaviour threshold marks the point at which the main mitigation is activated, instead of a simple fallback to ACC. In the original description of this system, the vehicle gains a temporary platoon leader status for all following vehicles, while increasing its distance to its predecessor. This is to prevent following vehicles to assume that the distancing process itself constitutes misbehaviour. However, due to the glaring risk of immediate privilege escalation by any vehicle using this protocol, this facet of the system has not been implemented. The gradual distancing will therefore generally lead to a slight increase in suspiciousness of the vehicle, yet not to an extent that the following vehicles revert to ACC. Wolf et al. [64] do not explicitly state that suspicious messages are being disregarded. Yet, to achieve the desired behaviour this is necessary, since the falsified values would otherwise continue to interfere with the victims driving behaviour. In the case of an attacker falsifying its values to simulate braking or deceleration, it would even lead to an amplification of the attacks strength, since the victim would not only try to match the perceived deceleration of its predecessor, but at the same time increase its distance to the predecessor. Therefore, for this work the assumption is made, that, if the suspiciousness of the predecessor surpasses the $noise$ threshold, its messages are being disregarded until the suspiciousness reaches acceptable levels again, i.e., falls back below the $noise$ threshold. A detailed code exempt showing the implementation of this system can be found in the Appendix at A.1.

3.2.3.2 PRIME

The implementation of the proposed PRIME misbehaviour mitigation system is structured as follows: When an attack is detected by the vehicle immediately behind the attacker, it initiates a misbehaviour report to the platoon leader. This report, formatted as an exclusion request, includes the ID of the offending vehicle. The platoon leader, upon receiving the report, contacts the accused vehicle, requesting it to leave the platoon and change lanes, allowing the rest of the platoon to close the gap. To validate the lane change, the platoon leader now contacts the vehicle that was the predecessor to the attacker. Using its rear-facing sensors, this vehicle confirms whether the attacker has genuinely left the platoon by comparing the measured distance to its direct platoon follower after the exclusion. If the measured distance does not align with the values received via beacons from the follower, indicating a non-platoon member in between, the attacker has not physically left. Conversely, if the measurements match, the vehicle reports to the platoon leader, confirming the attacker's compliance. Upon receiving confirmation of the attacker's compliance and lane change, the exclusion process is repeated for the vehicle that initiated the exclusion request. The exclusion of the accuser adds a layer of protection against abuse of the protocol itself. By forcing the accuser to relinquish their position as well, it can be ensured that no attacker can advance within a platoon by reporting their predecessor. With both vehicles out of the platoon, the gap can be closed. The vehicle that filed the report attempts to rejoin the platoon a few seconds after the lane change is completed. It's crucial to note that the rejoin request is restricted to rejoining at the end of the platoon. Middle join maneuvers are prohibited in this context due to the excluded vehicles being considered suspicious, making them unfit to rejoin at a position where they could potentially launch another attack. After the successful rejoin of the reporting vehicle, the attacker sends a request to join the platoon. For the context of this work this request will be accepted. Letting the attacker assume the tail end position of the platoon to showcase the now nullified attack potential. Fig. 3.1 shows a general sequence diagram, detailing the order of operations required during the exclusion of a vehicle, while Fig. 3.2 does the same for the rejoining. Together they detail a full run of the PRIME protocol after an attack has been detected. It has to be noted that the mentioned functions, while not explicitly shown in the diagram, are all performed with network security in mind. The communication between vehicles is secured as described in Sec. 2.1. The parameters, necessary for the security of the messages, have been omitted in the figure for readability. A description of the actions performed by the functions can be found in Table 3.1 and Table 3.2 respectively.

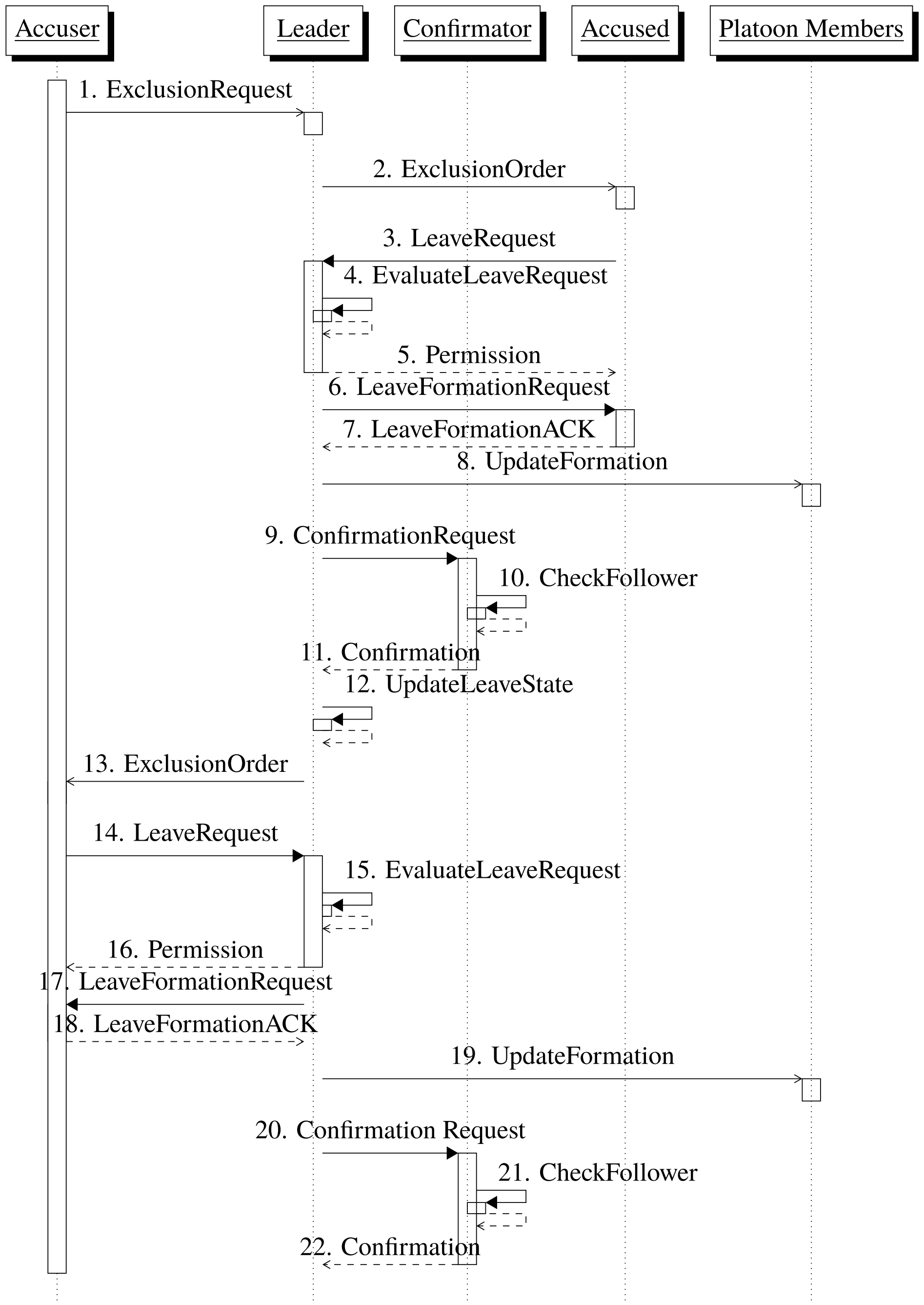


Figure 3.1: Sequence diagram of the PRIME exclusion protocol

Function	**Definition**
ExclusionRequest	Vehicle requests exclusion of its predecessor due to misbehaviour
ExclusionOrder	Leader orders vehicle to leave the platoon
LeaveRequest	Vehicle requests permission to initiate the leaving process
EvaluateLeaveRequest	Leader evaluates whether a leave maneuver is possible and permitted at the moment
Permission	Leader either permits or denies the initiation of the leave maneuver. Denial will lead to a repetition of the LeaveRequest
LeaveFormationRequest	Leader requests the vehicle to leave the formation and switch to ACC
LeaveFormationACK	Vehicle confirms to the leader that it is no longer acting as part of the platoon
UpdateFormation	Leader notifies all platoon members of the new platoon formation
ConfirmationRequest	Leader requests confirmation that the excluded vehicle has physically left the platoon
CheckFollower	Confirming vehicle uses its sensors to evaluate whether its follower has left
Confirmation	Confirming vehicle notifies the Leader that the vehicle has left the platoon
UpdateLeaveState	The accused vehicle has been fully excluded. Leader updates its internal state to reflect that, before restarting the exclusion process for the accuser

Table 3.1: Annotation table for Fig. 3.1

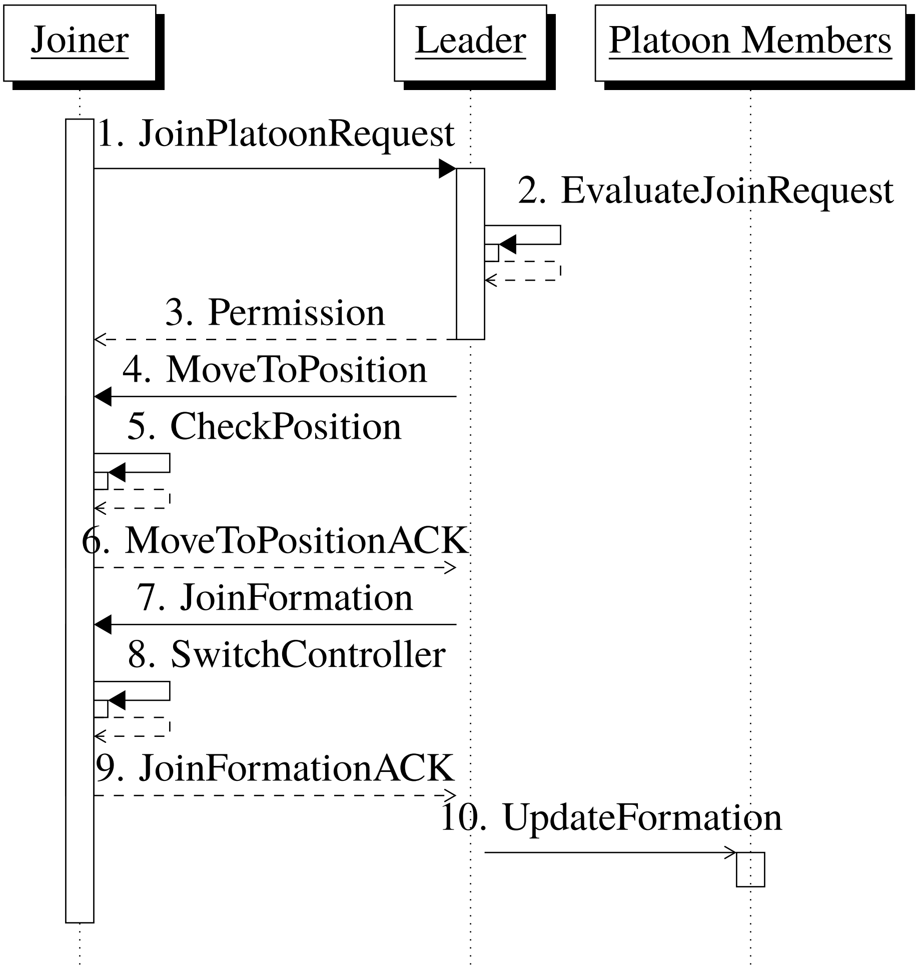


Figure 3.2: Sequence diagram of the PRIME rejoin protocol

Function	**Definition**
JoinPlatoonRequest	Vehicle requests to join the platoon
EvaluateJoinRequest	Leader evaluates whether maneuver is possible and permitted
Permission	Leader permits or denies the request. Denial can lead to a repetition of the request
MoveToPosition	Leader requests the joiner to move to the back of the platoon behind the last platoon vehicle
CheckPosition	Vehicle evaluates whether it has reached the desired position
MoveToPositionACK	Vehicle confirms that it has reached the desired position
JoinFormation	Leader sends all relevant platooning information to the newly joined vehicle
SwitchController	Vehicle switches to platooning controller
JoinFormationACK	Vehicle confirms that it has fully joined the platoon
UpdateFormation	Leader notifies all platoon members about the new platoon formation

Table 3.2: Annotation table for Fig. 3.2

3.2.4 Replicability

The simulation has been set up in a way that allows easy changes to several parameters, defining the behaviour of benign and malicious nodes in the platoon. The configuration file can be used to set various properties of the simulated scenario. These configuration parameters include:

1. Activation or deactivation of defensive behaviour

2. Gradual or constant attack type

3. Which kinematic parameter to falsify and to what extent

4. Attack and detection times

All test cases and simulation parameters can be replicated and tested either in bulk or individually.

3.3 Simulation and Evaluation Tools

3.3.1 Simulation Environment

The simulation environment chosen for this work requires a combination of three different tools. To simulate the vehicle dynamics, the platooning functionality and the different controllers as described in Sec 1.1, a tool named Plexe [68], which builds on the vehicular network simulation framework Veins [69], is utilized. Plexe offers several predefined scenarios to evaluate platooning performance in different traffic situations, e.g. pedestrians, heavy traffic, platooning maneuvers etc. In the context of this thesis only the predefined join maneuver for platoons has been utilized. Plexe utilizes several functionalities of a tool called Simulation of Urban MObility (SUMO) [70]. SUMO implements the functionalities to simulate road networks, vehicles and several of the the entities present in the ITS. Plexe and SUMO both require another tool called Objective Modular Network Testbed (OMNeT++) [71], a network simulation tool for distributed systems. OMNet++ is the perquisite for Plexe to allow the network communication between the different entities. All here described tools use the C++ programming language. The main functionality of the platoon, the attacks and the misbehaviour mitigation system have been implemented in Plexe. However, to properly work with all tested controllers SUMO had to be extended, since not all controllers supported a restructuring of the platoon formation by default.

3.3.2 Simulation Setup

To evaluate the proposed mitigation system multiple test scenarios were considered. The basis of all these test scenarios is a basic platooning scenario, with 7 cars travelling down a straight road for a simulation time of 120 seconds. The parameters describing this scenario can be found in Table 3.3. These values apply unaltered to all considered test scenarios. In this base scenario the only alteration to the unobstructed straight driving behaviour is introduced by having the leader slightly oscillate it speed, to simulate a more realistic and imperfect driving behaviour. Each experiment runs for 120 seconds with a warm-up period of 5 seconds in the beginning, to allow the platoon to stabilize fully. Should the experiment involve any alterations to the base scenario, they will only be applied after the warm-up period has passed. Each experiment ends either after the simulation time has run out, or if a crash occurs. In case of collisions the final difference between the speeds of the two colliding vehicles

Property	**Value**
Controller	PATH, Flatbed, Consensus Ploeg
Spacing	$5m, 5m, 0.8s, 0.5s$
Platoon size	7
Leader speed	$80km/h, 100km/h, 120km/h$
Oscillation amplitude	$2km/h$
Oscillation frequency	$0.2Hz$
Simulation time	$120s$
Warm-up period	$5s$
Attacker Index	3

Table 3.3: General simulation parameters

Attack configuration	**Falsification value**
Constant speed attack	$[-3, 3]km/h$
Constant acceleration attack	$[-1.5, 1.5]m/s^2$
Constant position attack	$[-10, 10]m$
Gradual speed attack	$[-0.5, 0.5]km/h$ per beacon
Gradual acceleration attack	$[-0.015, 0.015]m/s^2$ per beacon
Gradual position attack	$[-2.5, 2.5]m$ per beacon

Table 3.4: Attack parameters

is used to quantify the severity of the crash i.e. the impact velocity. The platoon size of 7 vehicles has been chosen, to have several followers behind the attacker, positioned at index 3, to properly analyze the attack impact on the platoon stability. However, since the protocol under investigation involves a restructuring of the platoon, where a vehicle has to decelerate to the end of the platoon, longer or shorter columns would require considerably longer or shorter simulation times, respectively, to complete the process.

Table 3.4 shows the parameters used in the various falsification attacks. Two different variations of falsification attacks, constant and gradual attacks, have been tested in this work. Constant falsification attacks falsify a single parameter by adding or subtracting a constant from the true value. Gradual attacks on the other hands falsify multiple parameters at once to make sure that position, acceleration and velocity follow a believable pattern. These attacks gradually increase or decrease a kinematic parameter with each sent beacon to mimic e.g. accelerating or decelerating behaviour. The parameters of the attacks have been chosen by testing out how strongly they impact the platoon. Since the controllers react very differently to changes in specific parameters, they are not finetuned to provoke a specific outcome in all test-cases. The final parameters have been chosen to provide disturbances which are noticeable enough to not be confused with noise, yet vary in their impact on platoon stability. Finally, the last kind of attack which has been tested is the jamming attack. This attack will start after the 5 seconds warm-up phase cutting vehicle 4 off from receiving any further beacons. This kind of attack has been tested in two different configurations. The two CVS controllers, PATH and Flatbed, use in SUMO by default an option which utilizes the last received acceleration of the predecessor to predict the future speed. This would, under normal circumstances, allow for a faster stabilization of the platoon. However, in the jamming scenario this option leads to strong reactions by the jammed vehicle.

Chapter 4

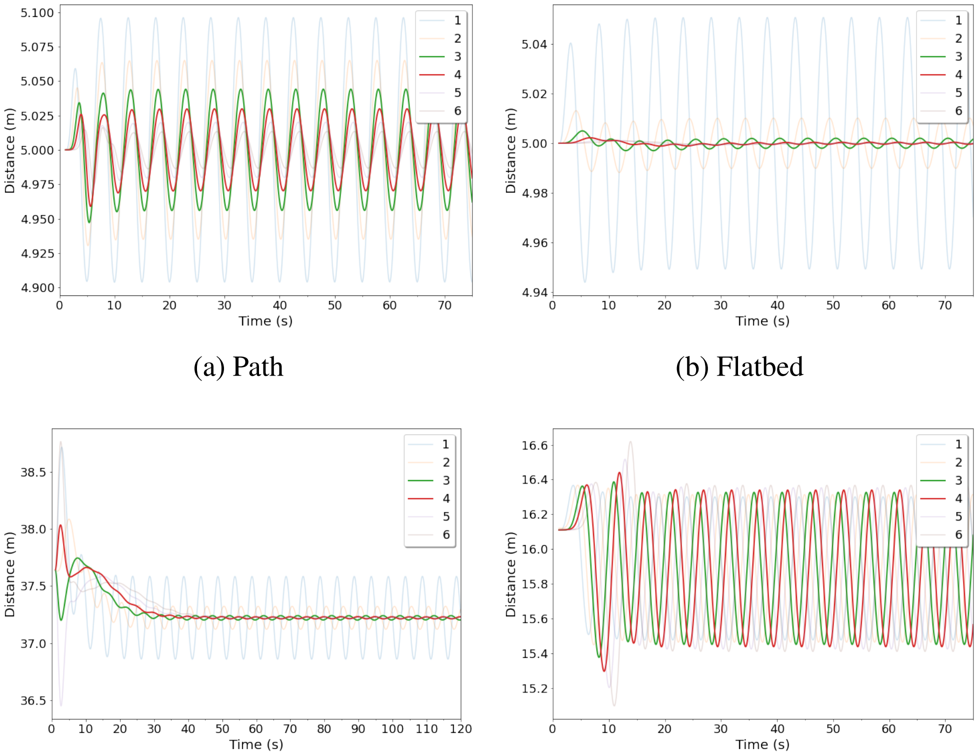
System Analysis

Ensuring the full recovery of a platoon's stability after the detection of misbehaviour is a challenge that existing mitigation strategies have not yet adequately addressed. Many strategies primarily focus on mitigating short-term issues and consider falling back to ACC as the final solution if the attack persists. The proposed strategy presented here aims to establish a system that effectively isolates the suspected attacker at the tail end of the platoon, facilitating the recovery of a stable platoon formation. To evaluate the performance and potential of this mitigation strategy, several metrics are considered. The analysis revolves around two main aspects. The primary objective is to prevent collisions. Therefore, the initial analysis explores how effectively the system mitigates situations that would typically lead to collisions. The secondary goal is to retain and regain stability. Hence, the system is also assessed based on how severely attacks disrupt the stable state of the platoon and the duration it takes to return to a stable state. To quantify the destructive potential of attacks leading to collisions, the impact severity of colliding vehicles has been analyzed to provide a more complete picture of the attack situation. Additionally, it has been measured how long a full execution of the here proposed protocol takes.

4.1 Metrics

Vehicle platoons, under benign conditions, are capable of upholding the inter vehicle spacing in a stable manner due to V2V communication. Therefore, in instability inside a platoon can be quantified by fluctuations in the inter vehicle distance. To analyze the impact an attack has on a platoon member and its followers, this work uses the difference between the desired distance and the actual distance a vehicle keeps to its predecessor. Should the distance between vehicles reach a value of 0m, it constitutes a collision. The severity of this collision can be quantified by measuring the difference in speed both vehicles have at the moment of impact, to estimate the potential damage. The last metric we apply is the execution time of the protocol. To make this protocol feasible for road usage it has to complete within a reasonable timeframe as to not obstruct the second lane for a considerable amount of time.

4.2 Results



(a) Path

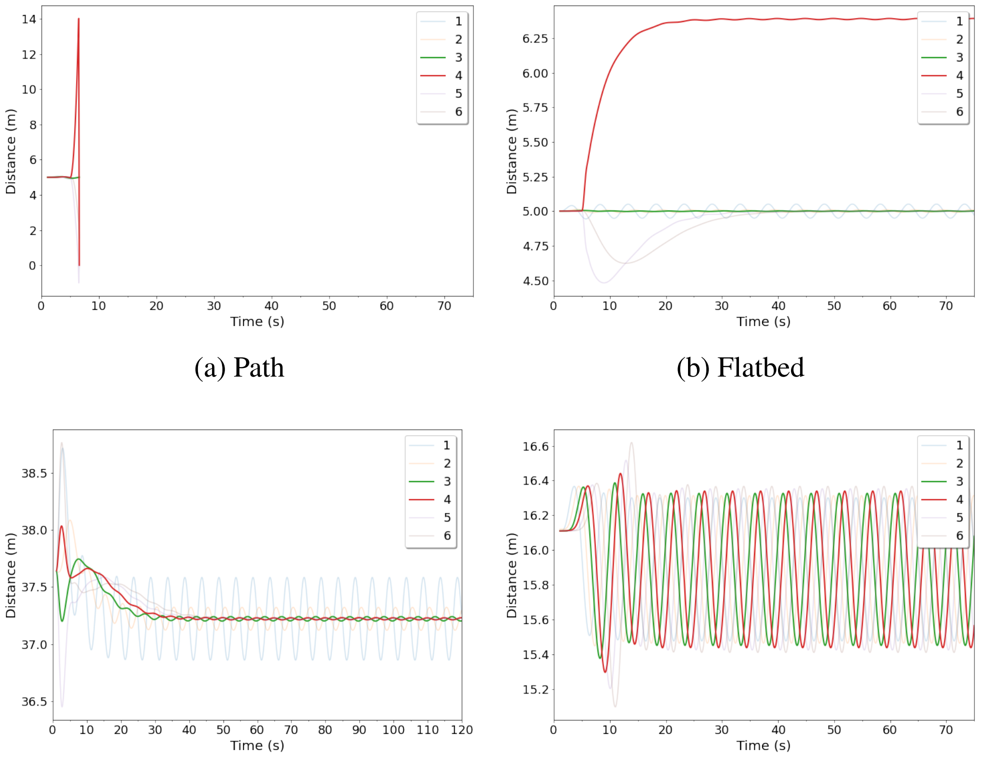
(b) Flatbed

(c) Consensus

(d) Ploeg

Figure 4.1: Distances to predecessor in absence of attacks at 100km/h; Leader (index 0) is omitted, since there is no predecessor

In Figure 4.1 we can observe how the four controllers behave if no attacker is present. As expected, the vehicles will match the oscillation of the leader with the amplitude of their own oscillation becoming smaller the further back in the platoon the vehicle is positioned. This scenario serves as a baseline example to compare the following attack scenarios against. The figure exemplifies that the controllers show noteworthy differences in their behaviour even in the absence of attacks.



(a) Path

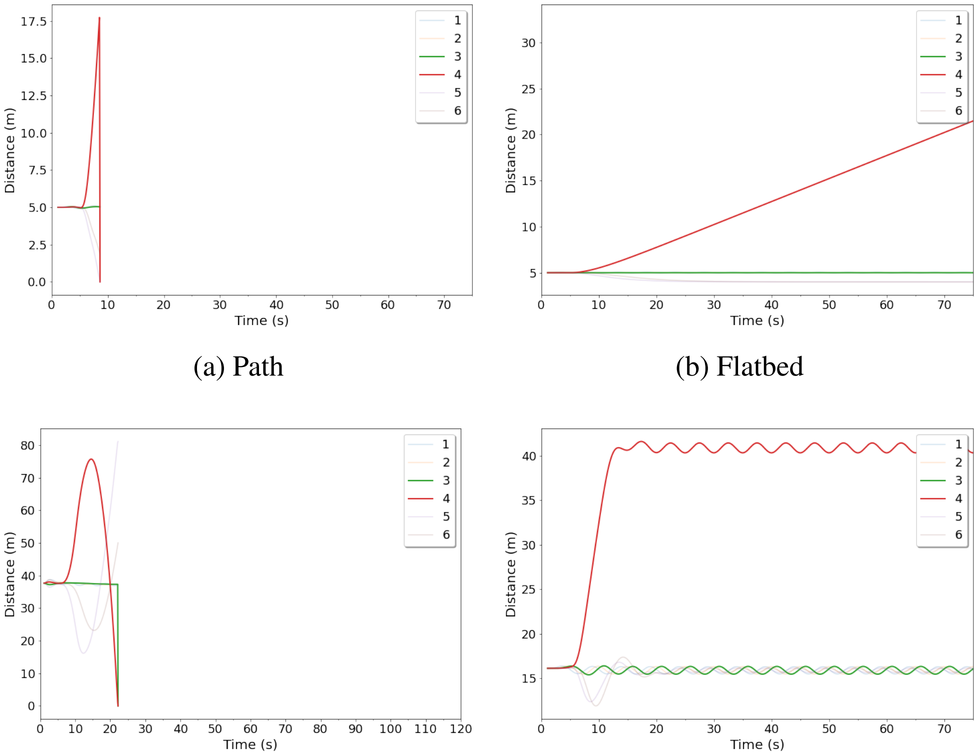
(b) Flatbed

(c) Consensus

(d) Ploeg

Figure 4.2: Distances to predecessor during constant negative speed attack (-3 km/h) with attacking vehicle at position 3 and no mitigation at 100km/h

Figure 4.2 showcases the behaviour of the vehicles when vehicle 3 starts offsetting its own speed by a constant -3km/h, starting at 5 seconds. In the absence of any kind of mitigation system these falsified messages are fully trusted and their contents will be used by the follower to compute their new acceleration. For the Path controller this attack leads to instant, strong deceleration by vehicle 4, which subsequently crashes into its follower. A similar reaction can be seen with the Flatbed controller. Here, however, the controller has a more moderate reaction to the speed change by its predecessor, leading to only a slight increase in the distance between vehicle 4 and 3. Both, Consensus and Ploeg, do not utilize the speed parameter at all to compute their behaviour and show therefore no reaction to this falsification attack since all other parameters remain unchanged.



(a) Path

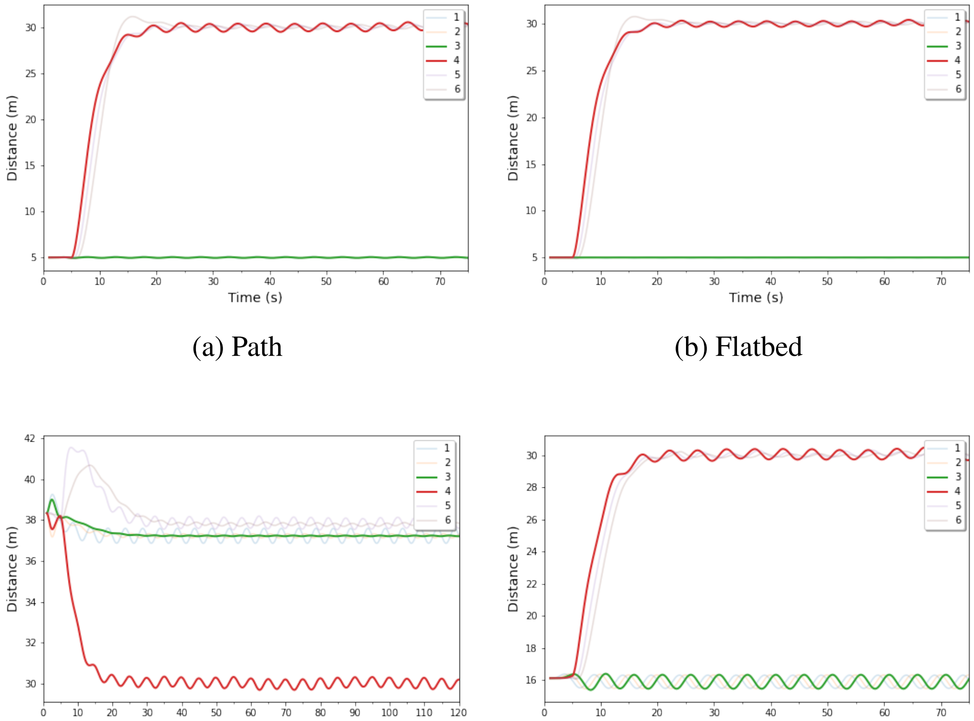
(b) Flatbed

(c) Consensus

(d) Ploeg

Figure 4.3: Distances to predecessor during gradual speed attack without mitigation at 100km/h

In Figure 4.3 the attack has been changed to a gradual attack, where all parameters are falsified based on a gradually decreasing speed. With each sent beacon the attacker deducts 0.5km/h from its speed value and recomputes its acceleration and position accordingly. For the Path controller this attack leads, again, to a collision after rapid deceleration by vehicle 4. Vehicle 4, when using the Flatbed controller, will keep increasing its distance to the attacker. Yet, other than PATH, Flatbed computes its acceleration with a stronger weight on the leader behaviour rather than the predecessor, which in turn leads to a less intense reaction to the here applied falsifications. The deceleration it exhibits is slow enough for the following vehicles to match, which in turn means that no collision will take place. Consensus and Ploeg, show a very different reaction. They show a strong initial reaction to the falsification attacks, but as the position parameter reaches a position which is no longer in front of the follower vehicle this changes. Consensus, due to its implementation, will accelerate as a result leading to a crash with the attacker vehicle, while Ploeg will simply stop decelerating. These results can be attributed to the complete absence of any mitigation system which would prevent physically impossible parameters from influencing the controllers.



(a) Path

(b) Flatbed

(c) Consensus

(d) Ploeg

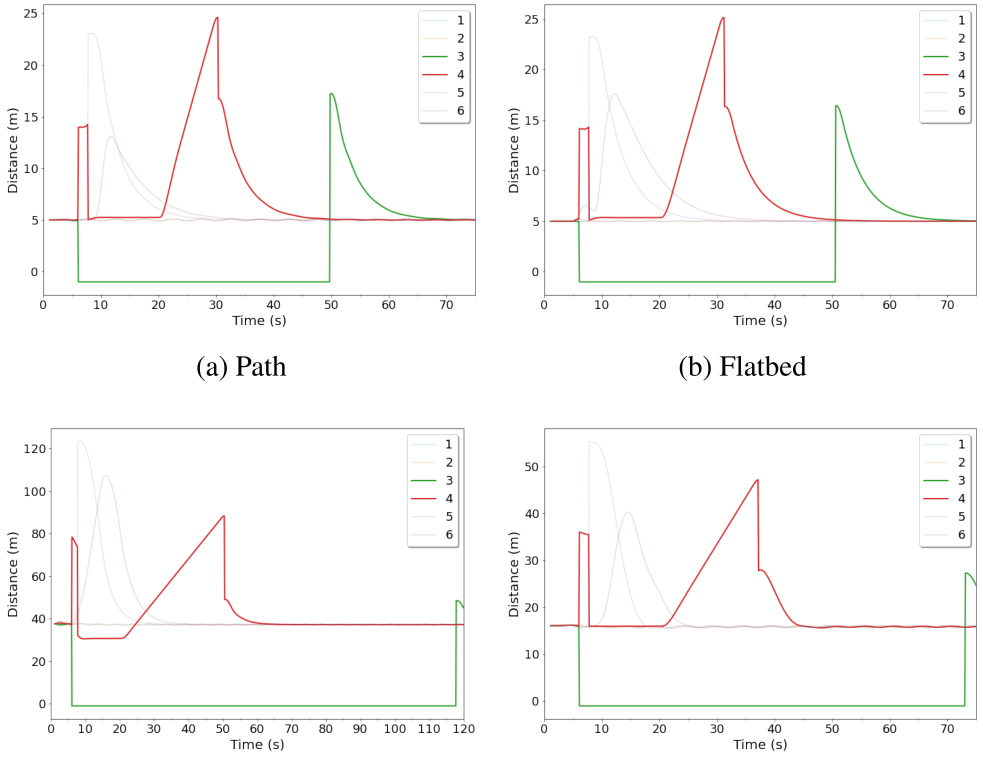
Figure 4.4: Distances to predecessor during gradual speed attack with suspiciousness based mitigation at 100km/h

Figure 4.4 shows the same attack with only the suspiciousness based mitigation enabled. The mitigation system leads, for all vehicles behind the attacker, to an increase in their distance to their predecessor. The maximum distance is the ACC headway of 2 sec. As long as the suspiciousness threshold is surpassed the messages by the predecessor are being discarded. This mitigation system effectively prevents crashes, but, in the case of an ongoing attack, prevents the platoon from re-stabilizing fully. In this scenario the suspiciousness will remain at its maximum, effectively breaking the platoon apart behind the attacker. Again, Consensus exhibits behaviour which falls outside of the expected results. Due to its implementation, the distance it keeps does not actually match a 0.8 second headway. This in turn means that during the execution of this mitigation system the distance to the predecessor will actually decrease rather than increase to the ACC distance of 2 seconds.

Attack Configuration	**Detected**
Gradual Speed Attack	True
Gradual Acceleration Attack	True
Gradual Position Attack	True
Constant Speed Attack	False
Constant Acceleration Attack	True
Constant Position Attack	False

Table 4.1: Attack detection with suspiciousness based mitigation

It is important to note that the suspiciousness based mitigation system is implemented to detect misbehaviour based on the difference in acceleration to the leader vehicle. Therefore, constant attacks which falsify any other parameters but leave the acceleration at its true value will not be detected even though they can influence the controller behaviour. Which attacks are detected and which are not can be seen in Table 4.1.



(a) Path

(b) Flatbed

(c) Consensus

(d) Ploeg

Figure 4.5: Distances to predecessor during gradual speed attack with both PRIME mitigation and suspiciousness based mitigation at 100km/h

In Figure 4.5 the same attack is being run. This time, however, surpassing the misbehaviour threshold will trigger the PRIME system. The sudden drop in distance to 0 meters for vehicle 3 happens when vehicle 3 changes lane to leave the platoon after being evicted by the leader due to vehicle 4 reporting the attack. In this scenario there is no outside traffic, therefore the attacker can leave immediately when the PRIME system starts. Shortly after the attacker (vehicle 3), the follower (vehicle 4) leaves the platoon. Vehicle 4 now decelerates to the back of the platoon and rejoins. After the successful rejoin of vehicle 4, vehicle 3 repeats the same procedure and joins at the tail end of the platoon. The difference in the timings between the controllers can largely be attributed to the differing inter vehicle distances. Especially Consensus and Ploeg have significantly larger distances than Path and Flatbed at this speed. Reaching the tail end of the platoon takes accordingly longer in these cases.

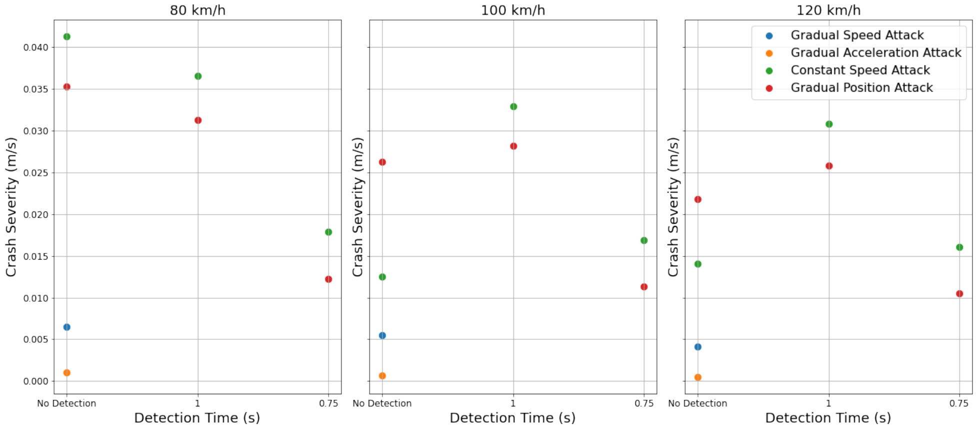


Figure 4.6: Collision intensities for PATH during negative attacks when PRIME is triggered with different detection times.

As an alternative to the suspiciousness based mitigation, PRIME can also be triggered by another detection system. Whether or not PRIME can prevent collisions is in this case dependent on the detection time. Fig.4.6 shows the collision intensities when using the PATH controller at different speeds depending on the chosen detection time. The figure displays only collision intensities for the PATH controller, since no other controller collides during negative attacks. This is either due to larger intra-platoon distances and therefore longer reaction time for the follower, or, in the case of the Flatbed controller, due to the intensity with which the controller reacts to received beacons by the predecessor. The detection time indicates how many seconds after the attack the detection system classifies the predecessor as misbehaving and initiates the PRIME mitigation system. Without a detection system we can observe that four of the attacks lead to a collision between the victim and their follower. If the attack is being detected 1 second after the first falsified message, the constant speed attack as well as the gradual position attack still lead to a collision, since the deceleration of the victim is at this point already too high to still prevent the crash. PRIME being triggered at 0.75 seconds after the attack shows a decrease in collision severity since the false messages are being discarded earlier. However, both attacks still lead to a collision. A detection time of 0.5 seconds proved sufficient to prevent all implemented negative attacks from resulting in a collision. The results show clearly, that even if an attack is detected before the vehicles collide, it might not be sufficient to prevent the collision.

Controller	Gradual	Attack	Suspiciousness	PRIME	80Km/h	100Km/h	120Km/h
PATH	False	SPEED	False	False	0,0016	0,0015	0,0014
PATH	False	SPEED	True	False	0,0016	0,0015	0,0014
PATH	False	SPEED	True	True	0,0016	0,0015	0,0014
PATH	True	ACC	False	False	0,0004	0,0010	0,0014
PATH	True	ACC	True	False	0,0028	0,0028	0,0024
PATH	True	ACC	True	True	0,0028	0,0028	0,0024
PATH	True	POS	False	False	0,0082	0,0100	0,0112
PATH	True	SPEED	False	False	0,0124	0,0137	0,0151
CONSENSUS	False	ACC	True	False		0,0095	0,0014
CONSENSUS	False	ACC	True	True		0,0095	0,0014
CONSENSUS	True	ACC	False	False	0,0018	0,0008	0,0026
CONSENSUS	True	POS	False	False	0,0024	0,0047	0,0053
CONSENSUS	True	SPEED	False	False	0,0037	0,0015	0,0009
FLATBED	True	ACC	False	False	0,0002	0,0004	0,0008
FLATBED	True	SPEED	False	False	0,0003	0,0001	0,0001
PLOEG	True	ACC	False	False	0,0051	0,0074	0,0051

Table 4.2: Impact Velocities for positive attacks without timed mitigation at different speeds

It might be that an attacker has no interest in preserving the attacking vehicles safety e.g. if the vehicle is merely compromised by malware. In this case it might also be a feasible attack to provoke the follower to crash into the attacking vehicle. When using positive falsification attacks, i.e. attempting to force the victim to accelerate, the outcome is very different. Table 4.2 shows that far more scenarios lead to collisions. This can easily be explained with the chosen communication structure. Since none of the implemented controllers use a bidirectional communication topology, they simply do not react to changes in their followers beacons. This means that, in contrast to the negative attacks, the vehicle with which the victim collides will make no attempts to evade the collision. An especially interesting scenario in this case can be observed when using the Consensus controller together with suspiciousness based mitigation. While constant acceleration attacks do not lead to collisions for the other controllers they do so for the Consensus controller. This phenomenon can be explained by a finding which will later be discussed for the jamming attacks. Jamming of the Consensus controller leads, depending on the timing of the attack, either to acceleration or deceleration due to its implementation in SUMO. In this case the positive constant acceleration attack will after a short while lead to a surpassing of the suspiciousness threshold as explained in Section 3.2.3.1. However, since the attack is comparably subtle, it will not surpass the misbehaviour threshold in time and will therefore not lead to a fallback to ACC but instead attempt to gradually increase the safe distance while ignoring the falsified messages. This, however, leads to a scenario where the vehicle is technically jammed and since this jamming happens during acceleration the vehicle will eventually crash into its predecessor. The table does not show detection timings, since, other than for negative attacks, even a 1 second detection timeframe proved sufficient to prevent collisions for all controllers. This is mostly due to the comparably long time it takes for a vehicle to accelerate in contrast to the short time it takes to decelerate, i.e brake.

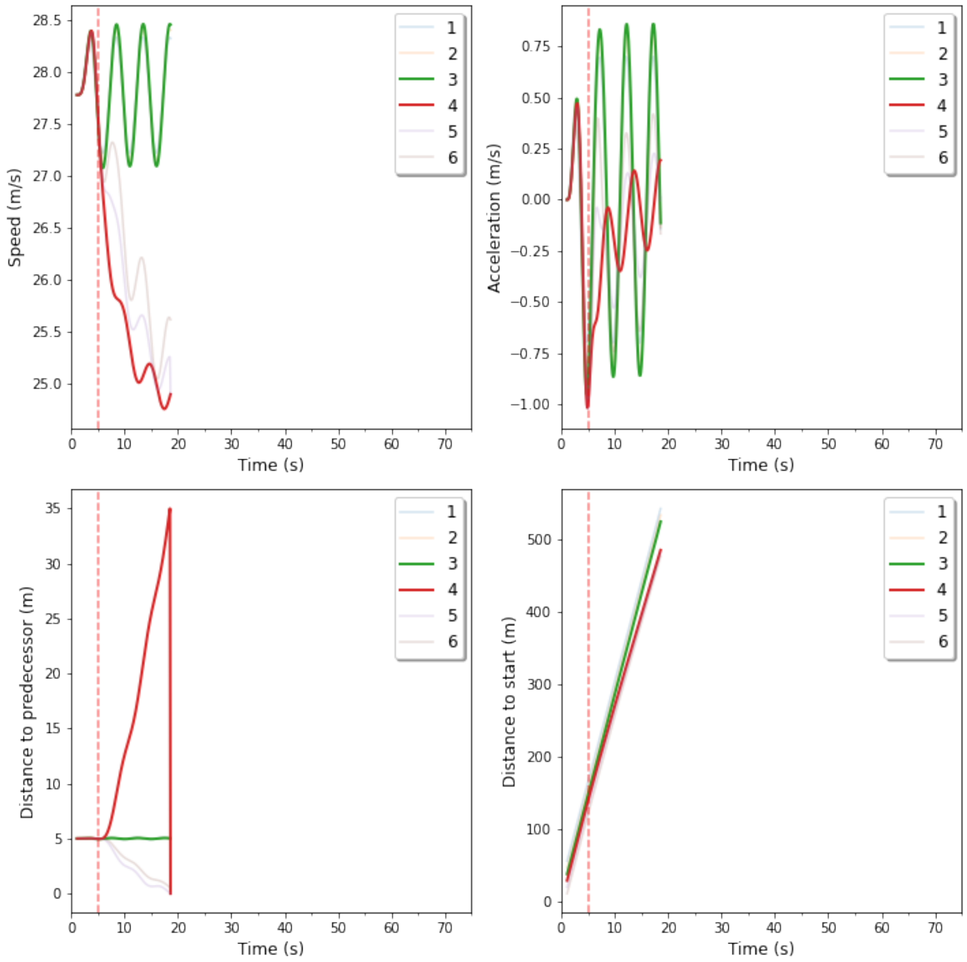


Figure 4.7: PATH controller when jammed after 5 seconds at 100km/h with prediction activated

An interesting find which has been made during the course of this thesis, is in regards to jamming attacks. Figure 4.7 shows how the PATH controller behaves when vehicle 4 is being cut off from the communication via jamming after 5 seconds of simulation time. As visible in the graph, the vehicle collides with its follower just below the 20 second mark. However, this find has been further investigated and could be traced back to an implementation issue within the simulator itself. The implementation of PATH and Flatbed, instead of just directly using the parameters it receives in the beacons, attempts to predict the future speed based on the received acceleration value. This is done, as shown in Listing 4.1, by adding to the received speed values.

```
predecessorSpeed += (currentTime - vars->frontDataReadTime) * vars->frontAcceleration;
leaderSpeed += (currentTime - vars->leaderDataReadTime) * vars->leaderAcceleration;
```

Listing 4.1: Implementation of the speed prediction for PATH and Flatbed in PLEXE

The problem with this is, that this calculation does not only happen when a message is received, but instead each time the controller recomputes its kinematic parameters. While, under attack-free circumstances, this technique allows faster stabilization by using the prediction to anticipate future speeds, it becomes problematic and even dangerous during attacks. The jamming scenario exemplifies how using the predictive capabilities in the absence of actual new information leads to vehement miscalculations, since the last received acceleration of the predecessor and leader will be re-used in every calculation. Since the prediction parameter has considerable negative impact on the controllers behaviour during attacks, it has been deactivated for all attacks except the jamming attack as described here.

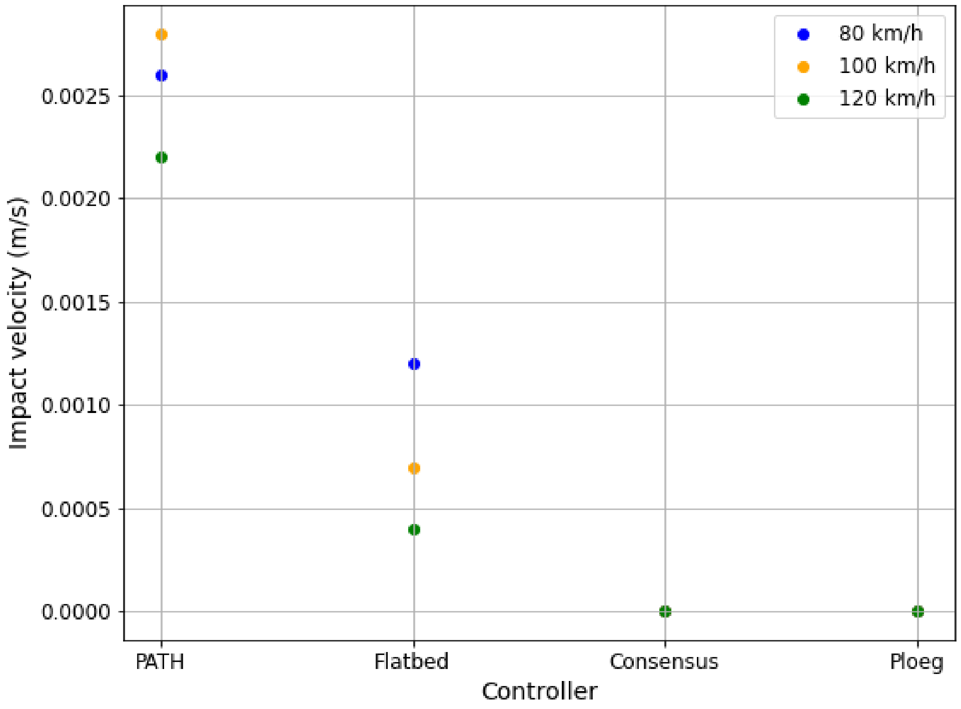


Figure 4.8: Collision severities during jamming attacks with activated prediction

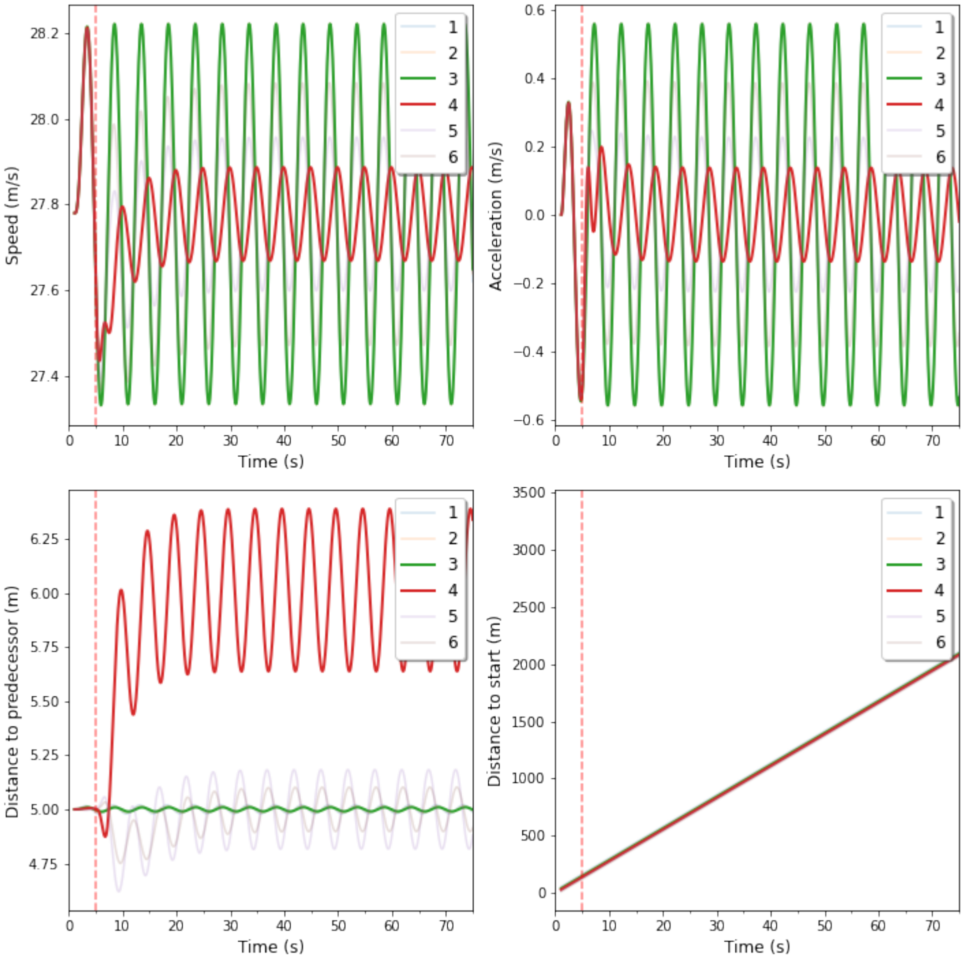
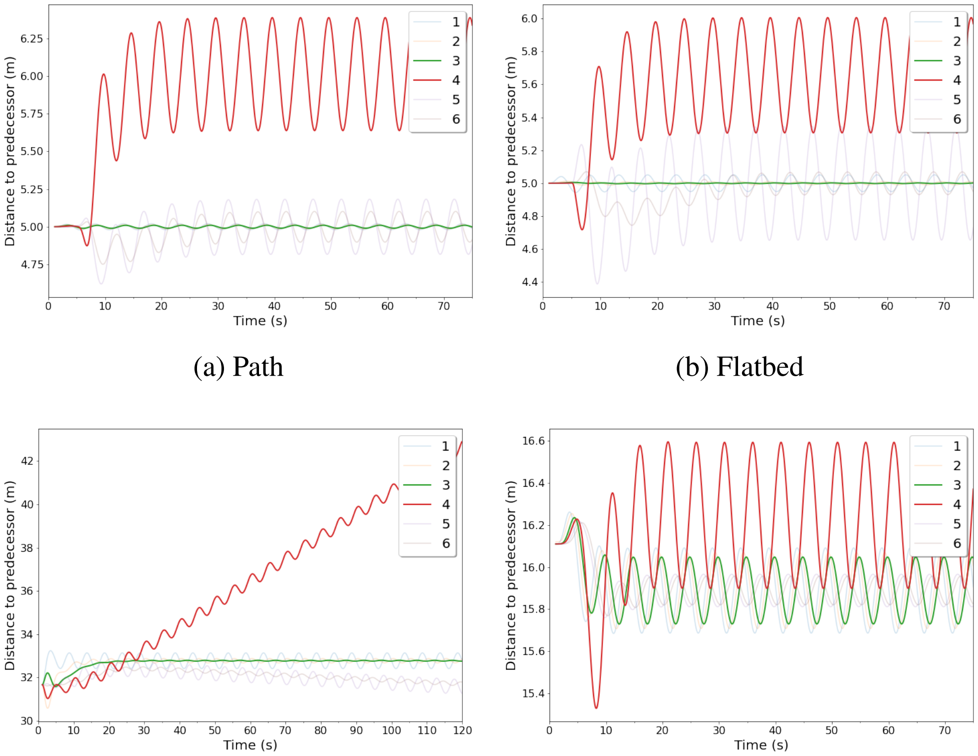


Figure 4.9: PATH controller when jammed at 100km/h after 5 seconds with prediction deactivated

By deactivating the predictive capabilities of the PATH and Flatbed controllers, we can observe that following this, both controllers exhibit a more expected behaviour in regards to the jamming. Figure 4.9 exemplifies this in detail for the PATH controller. Here, the jammed vehicle 4 no longer receives beacons and can no longer match the leader oscillation as a functioning platoon member. Without the prediction, it does not crash into its surrounding vehicles since no incentive for strong acceleration or deceleration is given. The same is true for Flatbed. For both controllers the jamming attack led to collisions when prediction was enabled, the severities of which can be seen in Fig. 4.8. Yet, with prediction disabled, both controllers refrain from accelerating or decelerating significantly.



(a) Path

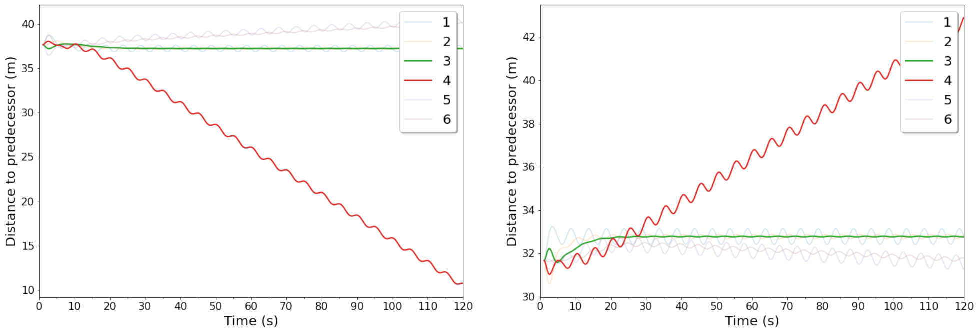
(b) Flatbed

(c) Consensus

(d) Ploeg

Figure 4.10: Intra-platoon distances when jammed at 100km/h without prediction after 5 seconds

The behaviour of all four controllers during jamming can be observed in Fig. 4.10. Ploeg and Consensus both do not have any prediction implemented by default and are therefore unaffected by whether or not the prediction parameter is activated or not. When unable to receive messages from the other platoon members the vehicle will in stop matching the oscillation and resume driving at a constant speed. The fluctuation in distance to the predecessor of the jammed vehicle, observable in the figure, is in this case not caused by the victims oscillation but rather by the absence of it, while the predecessor still oscillates with the rest of the platoon. The complete absence of any oscillation is not realistic driving behaviour for a vehicle cut off from the platoon communication. It should under normal circumstances be expected that the jammed vehicle would oscillate as well, however, at its own frequency. This could potentially lead to stronger fluctuations in distance to the predecessor, has, however, not been further explored withing this thesis.



(a) Jammed during acceleration

(b) Jammed during deceleration

Figure 4.11: Consensus controller when jammed at different times during the oscillation

As already visible in Fig. 4.10, Consensus exhibits a peculiar reaction to the jamming attack which, in the exemplified case, leads it to slowly approach its predecessor. The assumption for why this is happening, is due to the way the Consensus controller is implemented in SUMO. The controller saves a datastructure containing the parameters from the other platoon members, which is being updated each time a message is received. Based on this datastructure it computes its new acceleration and desired speed. However, if jammed this datastructure is not being updated anymore which means that, depending on at which point during the oscillation the vehicle is being jammed, it will get stuck either in acceleration or deceleration. Fig. 4.11 shows the controllers reaction to jamming during either acceleration or during deceleration. With a long enough simulation time jamming during the acceleration process will eventually lead to a collision, while jamming during deceleration will lead to an ever increasing gap withing the platoon. Considering that the obtained results can be attributed to how Consensus is implemented within SUMO, rather than an aspect of the controller itself, these scenarios have not been further analyzed.

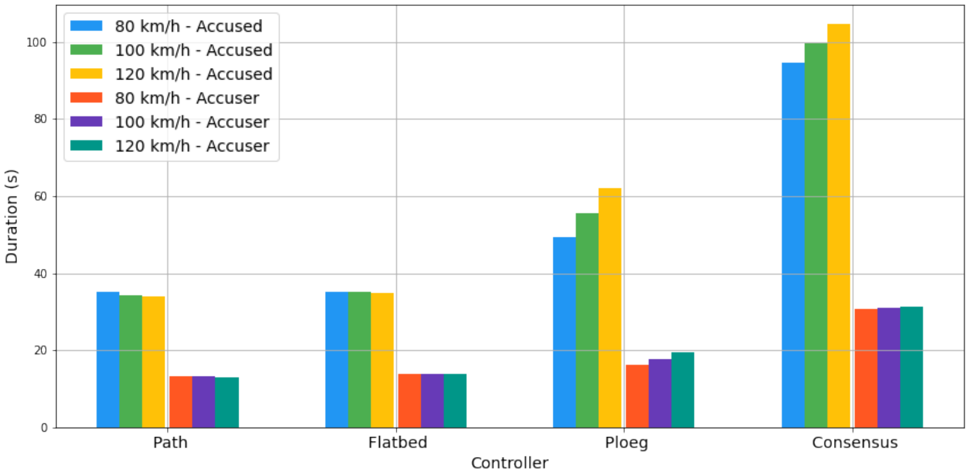


Figure 4.12: Required time to complete protocol execution at different platoon speeds

Finally, the execution times for the different controllers can be seen in Fig. 4.12. The PRIME protocol, under perfect circumstances, meaning under the assumption of a instantly compliant attacker and no obstructions to the leaving and rejoining of the platoon, can finish at a platoon speed of 100km/h in under a minute for all controllers except Consensus in the given simulation scenarios. Considering that the rejoining of the platoon by the attacker can be seen as optional, the time of completion for the accuser vehicle has been plotted separately. The execution time is mainly determined by the time it takes for the excluded vehicles to reach the tail end of the platoon. This means that the farther the vehicle is in relation to the last vehicle in the platoon, the longer it will take to complete a full execution of PRIME. This is influenced by the amount of cars in the platoon, the inter vehicle distances and the positions of the excluded vehicles. The figure shows that, with increasing speed, the execution time rises for Ploeg and Consensus while it remains mostly constant for PATH and Flatbed. This due to PATH and Flatbed being CVS controllers, meaning that they attempt to hold a fixed, predefined distance to their predecessor independent of their speed. Ploeg and Consensus, however, are CTH controllers, which means that with higher speeds the desired distance between vehicles will be larger, since they attempt to hold a time headway to their predecessor. Since larger gaps between vehicles means that the total length of the the platoon increases, it takes longer for the excluded vehicles to reach the tail end of the platoon to rejoin.

4.3 Discussion

The results emphasize the need for resilient mitigation and detection systems within vehicle platoons to ensure the safety of all involved vehicles. While suspiciousness-based mitigation proves adequately effective in preventing collisions among platoon members, it's evident that attacks can still severely disrupt the platoon's functionality by preventing the platoon from regaining its initial stability. The findings suggest that solely relying on suspiciousness-based mitigation could lead to a situation in which the platoon remains permanently destabilized due to the attacker's persistent presence. This situation often results in the platoon resorting to a full fallback to ACC, ultimately breaking up the formation. Depending on the attacker's position within the platoon, this breakdown may cause significant portions of the platoon to dissolve. It's clear that while suspiciousness-based mitigation serves as a viable short-term solution against isolated instances of faulty or malicious messages, an immediate fallback to ACC may not always be the best response upon exceeding the misbehaviour threshold. The results further show that restructuring the platoon, by compelling the attacker to relinquish their position, consistently leads to a return to the initial platoon stability prior to the onset of attacks. We can see in the results, that when PRIME is being activated at a specific time after the start of the attack many attacks still lead to a collision even though they are being detected. The main problem in this case is that the falsified messages are being assumed to be true until misbehaviour is confirmed. This makes a strong cases for the combination of the suspiciousness based mitigation and the PRIME mitigation system. That way suspicious messages would immediately lead to a safe distance increase to the offending vehicle, giving the system more time to evaluate whether or not the misbehaviour justifies an activation of PRIME or, should the attack stop, a return to the desired platooning distance. This combination effectively uses the collision prevention provided by the suspiciousness based mitigation with the stability regaining capabilities of PRIME. However, achieving this favourable outcome with PRIME relies heavily on the attacker's compliance with the directive to vacate their position. Should the attacker fail to comply, trailing vehicles are left with limited options, often making a temporary fallback to ACC necessary. In this case, however, the non-compliance of the attacker serves as a strong proof of misbehaviour which could in turn lead to a quick and effective revocation of the certificate to prevent future attacks by this attacker. One way which has been identified, concerning how this system could be potentially abused, would be to accuse a platoon member of misbehaviour and, by some means, selectively jam them from receiving the exclusion request from the leader. This would make a potentially benign vehicle look like a proven attacker even if no misbehaviour has been conducted. This kind of attack would be very sophisticated and complex, relying on perfect timing and execution, yet, the effectiveness is doubtful, since there would be no records apart from the non-compliance which would support the claim that the accused vehicle is actually an attacker. On caveat of PRIME is that it strongly relies on having the space to fully execute the restructuring of the platoon. Other vehicles travelling next to the platoon can, intentionally or not, prevent vehicles from leaving or rejoining the platoon. In these cases a temporary fallback to ACC would be the likely outcome for the vehicles behind the attacker. A single attacker vehicle cannot abuse the protocol to advance within the platoon, since the accuser as well as the accused are required to relinquish their position. It would require a minimum of two colluding attackers right behind each other and the absence of any validation of the exclusion request to make any gain in this regard. Should an attacker manage to get themselves and their non-misbehaving predecessor excluded, their direct follower would advance to the position the falsely accused vehicle held previous to the attack. For the attackers this would mean advancing one single position inside the platoon formation, at the cost of sacrificing an attacker vehicle within the platoon, since the accuser would need to relinquish their position. Considering that this kind of attack could only be conducted once, and that it requires the leader to accept the false accusation of a benign vehicle, it is unlikely to bring any noticeable benefit to the attackers. In the worst case scenario this would mean that two attackers, who previously held the positions 3 and 4 in the platoon, would achieve that one attacker would advance to position 2, while the other would relinquish their position and potentially rejoin at the back of the platoon. Under the assumption that the benign vehicle which had previously held position 2 rejoins the platoon, this would result in one more vehicle which could be affected by falsification attacks than before the abuse of the maneuver.

Chapter 5

Conclusions

The development and evaluation of the PRIME misbehaviour mitigation protocol for vehicular platoons represent a significant advancement in ensuring the stability and security of automated transportation systems. Through extensive experimentation via the various discussed scenarios, it has been demonstrated that PRIME effectively restores platoons to a stable state after being faced with a plethora of attacks. As is the case with most mitigation systems, PRIME heavily depends on being paired with a robust and effective detection system, which allows a timely reaction before significant damage has been done. When paired with the suspiciousness based mitigation system described by Wolf et al. [64], minor attacks can be mitigated by this system, while only attacks which clearly, or very likely, constitute misbehaviour will lead to an activation of the PRIME protocol, allowing for a measured response to varying threat levels. One of the key findings of this research is the substantial decrease, and in some cases nullification, of the attack potential posed by malicious actors targeting vehicular platoons by isolating them physically at the tail end of the platoon. Even in the case of a continuation of the falsification attempts by the attacker, the new platoon structure makes these efforts pointless, since no vehicle relies on the attackers beacons anymore. Another significant contribution is the prevention of protocol abuse to benefit from the restructuring of the platoon. No single vehicle can gain a stronger attack position, or a position of influence, by using the restructuring capabilities of PRIME. The abuse by colluding vehicles, as described in Section 4.3, can only allow one of the attackers to advance by a single position within the platoon, while sacrificing one of the attacking vehicles, which has to relinquish its position in the process. This abuse case requires both attackers to be right behind each other and it assumes that the false accusation of a benign vehicle is accepted. This is an unlikely yet not unrealistic scenario. However, even in the worst case, where the attackers actually succeed with the abuse of the protocol, it would only marginally increase the actual attack potential, since at most one more vehicle could be affected than before the abuse. It is unlikely that for a rational attacker potential gain outweighs the risk of detection in this case. The completion time of the PRIME protocol is influenced by several factors, including inter-vehicle distances, the attacker's position, and the size of the platoon. Understanding these factors is crucial for optimizing the implementation of PRIME in real-world scenarios, ensuring timely and effective response to potential threats while minimizing disruptions to vehicular operations. Importantly, the uncovering and analysis of simulator-related bugs during the course of this investigation have provided valuable insights into the potential limitations and challenges associated with evaluating mitigation protocols. By addressing these issues and accounting for their impact on the obtained results, this research contributes to the ongoing refinement and improvement of simulation methodologies in the field of vehicular platoon security. In summary, the findings presented in this thesis underscore the significance of the PRIME mitigation protocol in enhancing the stability and security of vehicular platoons. Moving forward, further research and development efforts will be essential for advancing the practical implementation of PRIME and ensuring its effectiveness in real-world scenarios, ultimately paving the way towards safer and more resilient automated vehicular systems.

5.1 Limitations

A major limitation of this work is that the PRIME protocol needs to use another lane for the restructuring of the platoon. Since the excluded vehicles would need to decelerate to reach the tail end of the platoon, this would imply that traffic on this lane could possibly be severely slowed down by the usage of this protocol. It has become clear that PRIME, while effective in regaining long term security, is a costly operation which needs to be justified by an accurate detection system. Another limitation encountered during this study is the limitation by the simulation environment itself. PLEXE and SUMO, as previously shown, will in many cases produce outputs which are strongly influenced by the implementation of the platooning controllers within the simulator.

5.2 Future work

The provided protocol is still merely a rough prototype which needs to be tested more extensively. An attack scenario which has been discussed but not quantitatively analyzed is that of an attacker who will refuse to comply with the protocol. In such a case PRIME needs to be extended with a robust fallback method which splits the platoon. Since this would not only involve a platoon separation but also a new leader election, it has been deemed out of scope for this work since a secure separation protocol would justify a study on its own. Without it, however, PRIME cannot be considered complete. As stated in Sec. 5.1 the simulation environment and its implementations of the controllers can have a strong impact on the outcome of several of the tested scenarios. To ensure that the properties of the mitigation system are truly as stated, it should be evaluated within an alternative simulation environment to further prove the validity of the simulations.

References

[1] A. Balador, A. Bazzi, U. Hernandez-Jayo, I. de la Iglesia, and H. Ahmadvand, “A survey on vehicular communication for cooperative truck platooning application,” *Vehicular Communications*, vol. 35, p. 100460, 2022. doi: https://doi.org/10.1016/j.vehcom.2022.100460. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2214209622000079 [Page 1.]

[2] R. Ghosh, R. Pragathi, S. Ullas, and S. Borra, “Intelligent transportation systems: A survey,” in *2017 International Conference on Circuits, Controls, and Communications (CCUBE)*, 2017. doi: 10.1109/C-CUBE.2017.8394167 pp. 160–165. [Page 1.]

[3] A. Sullivan and M. Hadi, Dec 2016. [Online]. Available: https://rosap.ntl.bts.gov/view/dot/36986/dot_36986_DS1.pdf? [Page 1.]

[4] Jun 2009. [Online]. Available: https://www.etsi.org/deliver/etsi_tr/102600_102699/102638/01.01.01_60/tr_102638v010101p.pdf [Page 1.]

[5] M. Amoozadeh, H. Deng, C.-N. Chuah, H. M. Zhang, and D. Ghosal, “Platoon management with cooperative adaptive cruise control enabled by vanet,” *Vehicular Communications*, vol. 2, no. 2, pp. 110–123, 2015. doi: https://doi.org/10.1016/j.vehcom.2015.03.004. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2214209615000145 [Pages 1, 2, and 11.]

[6] V. Milanés, S. E. Shladover, J. Spring, C. Nowakowski, H. Kawazoe, and M. Nakamura, “Cooperative adaptive cruise control in real traffic situations,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 1, pp. 296–305, 2014. doi: 10.1109/TITS.2013.2278494 [Page 2.]

[7] A. A. Alam, A. Gattami, and K. H. Johansson, “An experimental study on the fuel reduction potential of heavy duty vehicle platooning,” in *13th*

International IEEE Conference on Intelligent Transportation Systems, 2010. doi: 10.1109/ITSC.2010.5625054 pp. 306–311. [Pages 2 and 11.]

[8] A. Festag, “Cooperative intelligent transport systems standards in europe,” *IEEE Communications Magazine*, vol. 52, no. 12, pp. 166–172, 2014. doi: 10.1109/MCOM.2014.6979970 [Pages 2 and 8.]

[9] ——, “Cooperative intelligent transport systems standards in europe,” *IEEE Communications Magazine*, vol. 52, no. 12, pp. 166–172, 2014. doi: 10.1109/MCOM.2014.6979970 [Page 2.]

[10] H. Hasrouny, A. E. Samhat, C. Bassil, and A. Laouiti, “Vanet security challenges and solutions: A survey,” *Vehicular Communications*, vol. 7, pp. 7–20, 2017. doi: https://doi.org/10.1016/j.vehcom.2017.01.002. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2214209616301231 [Page 2.]

[11] R. van der Heijden, T. Lukaseder, and F. Kargl, “Analyzing attacks on cooperative adaptive cruise control (cacc),” in *2017 IEEE Vehicular Networking Conference (VNC)*, 2017. doi: 10.1109/VNC.2017.8275598 pp. 45–52. [Pages 3 and 15.]

[12] S. Dadras, R. Gerdes, and R. Sharma, “Vehicular platooning in an adversarial environment,” in *Proceedings of the 10th ACM Symposium on information, computer and communications security*, ser. ASIA CCS ’15. ACM, 2015. ISBN 9781450332453 pp. 167–178. [Pages 3 and 15.]

[13] K. Kalogiannis, M. Khodaei, W. M. N. M. Bayaa, and P. Papadimitratos, “Attack impact and misbehavior detection in vehicular platoons,” in *Proceedings of the 15th ACM Conference on Security and Privacy in Wireless and Mobile Networks*, ser. WiSec ’22. New York, NY, USA: Association for Computing Machinery, 2022. doi: 10.1145/3507657.3528552. ISBN 9781450392167 p. 45–59. [Online]. Available: https://doi.org/10.1145/3507657.3528552 [Pages 3, 15, and 17.]

[14] E. Preet, “A comparative study of routing protocols in vanet,” 03 2010. [Page 7.]

[15] A. Singh, D. M. Kumar, R. Rishi, and D. k. Madan, “A relative study of manet and vanet: Its applications, broadcasting approaches and challenging issues,” vol. 132, 01 2011. doi: 10.1007/978-3-642-17878-8$_63.ISBN978-3-642-17877-1pp.627--632.[Page\ 7.]$

[16] R. W. Pazzi, K. Abrougui, C. De Rezende, and A. Boukerche, "Service discovery protocols for vanet based emergency preparedness class of applications: A necessity public safety and security," in *Information Systems, Technology and Management*, S. K. Prasad, H. M. Vin, S. Sahni, M. P. Jaiswal, and B. Thipakorn, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 1–7. [Page 7.]

[17] S. Malik, M. Khan, and H. El-Sayed, "Collaborative autonomous driving—a survey of solution approaches and future challenges," *Sensors*, vol. 21, p. 3783, 05 2021. doi: 10.3390/s21113783 [Page 8.]

[18] ETSI, "intelligent transport systems (its); vehicular communications; basic set of applications; definitions," https://www.etsi.org/deliver/etsi_tr/102600_102699/102638/01.01.01_60/tr_102638v010101p.pdf, Tech. Rep., 06 2009. [Page 8.]

[19] "IEEE Standard for Wireless Access in Vehicular Environments (WAVE) - Certificate Management Interfaces for End Entities," *IEEE Std. 1609.2.1*, 2020. [Page 8.]

[20] "Technical Specification Group Services and System Aspects;Security aspect for LTE support of Vehicle-to-Everything (V2X)," *3GPP Std. TS 33.185*, vol. V16.0.0, pp. Rel–16, 2020. [Page 8.]

[21] N. Alexiou, S. Gisdakis, M. Laganà, and P. Papadimitratos, "Towards a secure and privacy-preserving multi-service vehicular architecture," in *2013 IEEE 14th International Symposium on "A World of Wireless, Mobile and Multimedia Networks" (WoWMoM)*, 2013. doi: 10.1109/WoWMoM.2013.6583472 pp. 1–6. [Page 8.]

[22] M. Khodaei, H. Jin, and P. Papadimitratos, "Towards deploying a scalable robust vehicular identity and credential management infrastructure," in *2014 IEEE Vehicular Networking Conference (VNC)*, 2014. doi: 10.1109/VNC.2014.7013306 pp. 33–40. [Page 8.]

[23] M. A. Ferrag, L. Maglaras, and A. Ahmim, "Privacy-preserving schemes for ad hoc social networks: A survey," *IEEE Communications Surveys Tutorials*, vol. 19, no. 4, pp. 3015–3045, 2017. doi: 10.1109/COMST.2017.2718178 [Page 8.]

[24] N. Alexiou, M. Lagana, S. Gisdakis, M. Khodaei, and P. Papadimitratos, "Vespa: Vehicular security and privacy-preserving architecture," *CoRR*, vol.

abs/2001.07534, 2020. [Online]. Available: https://arxiv.org/abs/2001.07534 [Page 8.]

[25] M. Khodaei, A. Messing, and P. Papadimitratos, "Rhythm: A randomized hybrid scheme to hide in the mobile crowd," in *2017 IEEE Vehicular Networking Conference (VNC)*, 2017. doi: 10.1109/VNC.2017.8275642 pp. 155–158. [Page 8.]

[26] H. Jin and P. Papadimitratos, "Resilient privacy protection for location-based services through decentralization," *CoRR*, vol. abs/2001.07583, 2020. [Online]. Available: https://arxiv.org/abs/2001.07583 [Page 8.]

[27] G. Calandriello, P. Papadimitratos, J.-P. Hubaux, and A. Lioy, "On the performance of secure vehicular communication systems," *IEEE Transactions on Dependable and Secure Computing*, vol. 8, no. 6, pp. 898–912, 2011. doi: 10.1109/TDSC.2010.58 [Page 9.]

[28] Y. Wang, Y. Ding, Q. Wu, Y. Wei, b. Qin, and H. Wang, "Privacy-preserving cloud-based road condition monitoring with source authentication in vanets," *IEEE Transactions on Information Forensics and Security*, vol. PP, pp. 1–1, 12 2018. doi: 10.1109/TIFS.2018.2885277 [Page 9.]

[29] H. Jin and P. Papadimitratos, "Dos-resilient cooperative beacon verification for vehicular communication systems," *Ad Hoc Networks*, vol. 90, p. 101775, 2019. doi: https://doi.org/10.1016/j.adhoc.2018.10.003 Recent advances on security and privacy in Intelligent Transportation Systems. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1570870518307108 [Page 9.]

[30] P. Papadimitratos, L. Buttyan, J.-P. Hubaux, F. Kargl, A. Kung, and M. Raya, "Architecture for secure and private vehicular communications," in *2007 7th International Conference on ITS Telecommunications*, 2007. doi: 10.1109/ITST.2007.4295890 pp. 1–6. [Page 9.]

[31] J. R. Douceur, "The sybil attack," in *Peer-to-Peer Systems*, P. Druschel, F. Kaashoek, and A. Rowstron, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2002. ISBN 978-3-540-45748-0 pp. 251–260. [Page 9.]

[32] C. Eryonucu and P. Papadimitratos, "Sybil-based attacks on google maps or how to forge the image of city life," in *Proceedings of the 15th ACM Conference on Security and Privacy in Wireless and Mobile Networks*, ser. WiSec '22. New York, NY, USA: Association for Computing Machinery,

2022. doi: 10.1145/3507657.3528538. ISBN 9781450392167 p. 73–84. [Online]. Available: https://doi.org/10.1145/3507657.3528538 [Page 9.]

[33] M. Khodaei, H. Jin, and P. Papadimitratos, "Secmace: Scalable and robust identity and credential management infrastructure in vehicular communication systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 5, pp. 1430–1444, 2018. doi: 10.1109/TITS.2017.2722688 [Page 9.]

[34] P. Papadimitratos, V. Gligor, and J.-P. Hubaux, "Securing vehicular communications - assumptions, requirements, and principles," *4th Workshop Embedded Security in Cars*, 01 2006. [Page 10.]

[35] P. Papadimitratos, L. Buttyan, T. Holczer, E. Schoch, J. Freudiger, M. Raya, Z. Ma, F. Kargl, A. Kung, and J.-P. Hubaux, "Secure vehicular communication systems: design and architecture," *IEEE Communications Magazine*, vol. 46, no. 11, pp. 100–109, 2008. doi: 10.1109/MCOM.2008.4689252 [Page 10.]

[36] F. Kargl, P. Papadimitratos, L. Buttyan, M. Müter, E. Schoch, B. Wiedersheim, T.-V. Thong, G. Calandriello, A. Held, A. Kung, and J.-P. Hubaux, "Secure vehicular communication systems: implementation, performance, and research challenges," *IEEE Communications Magazine*, vol. 46, no. 11, pp. 110–118, 2008. doi: 10.1109/MCOM.2008.4689253 [Page 10.]

[37] M. Khodaei and P. Papadimitratos, "The key to intelligent transportation: Identity and credential management in vehicular communication systems," *IEEE Vehicular Technology Magazine*, vol. 10, pp. 63–69, 12 2015. doi: 10.1109/MVT.2015.2479367 [Page 10.]

[38] 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)*, 2010. doi: 10.1109/WONS.2010.5437115 pp. 176–183. [Page 10.]

[39] P. Papadimitratos, "Mix-zones in wireless mobile networks," *Encyclopedia of Cryptography, Security and Privacy*, pp. 1–5, 2021. [Page 10.]

[40] C. Vaas, M. Khodaei, P. Papadimitratos, and I. Martinovic, "Nowhere to hide? mix-zones for private pseudonym change using chaff vehicles," in *2018 IEEE Vehicular Networking Conference (VNC)*, 2018. doi: 10.1109/VNC.2018.8628449 pp. 1–8. [Pages 10 and 11.]

[41] M. Khodaei and P. Papadimitratos, "Cooperative location privacy in vehicular networks: Why simple mix zones are not enough," *IEEE Internet of Things Journal*, vol. 8, no. 10, pp. 7985–8004, 2021. doi: 10.1109/JIOT.2020.3043640 [Pages 10 and 11.]

[42] A. P. Mdee, M. T. R. Khan, J. Seo, and D. Kim, "Security compliant and cooperative pseudonyms swapping for location privacy preservation in vanets," *IEEE Transactions on Vehicular Technology*, pp. 1–15, 2023. doi: 10.1109/TVT.2023.3254660 [Page 11.]

[43] R. J. Caudill and W. L. Garrard, "Vehicle-Follower Longitudinal Control for Automated Transit Vehicles," *Journal of Dynamic Systems, Measurement, and Control*, vol. 99, no. 4, pp. 241–248, 12 1977. [Page 11.]

[44] S. Darbha and J. Hedrick, "Constant spacing strategies for platooning in automated highway systems," *Journal of Dynamic Systems Measurement and Control-transactions of The Asme - J DYN SYST MEAS CONTR*, vol. 121, 09 1999. doi: 10.1115/1.2802497 [Page 12.]

[45] Y. Zheng, S. Eben Li, J. Wang, D. Cao, and K. Li, "Stability and scalability of homogeneous vehicular platoon: Study on the influence of information flow topologies," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 1, pp. 14–26, 2016. doi: 10.1109/TITS.2015.2402153 [Page 12.]

[46] B. Ribeiro, F. Gonçalves, A. Santos, M. J. Nicolau, B. Dias, J. Macedo, and A. Costa, "Simulation and testing of a platooning management protocol implementation," in *Wired/Wireless Internet Communications*, Y. Koucheryavy, L. Mamatas, I. Matta, A. Ometov, and P. Papadimitriou, Eds. Cham: Springer International Publishing, 2017. ISBN 978-3-319-61382-6 pp. 174–185. [Page 12.]

[47] R. Hall and C. Chin, "Vehicle sorting for platoon formation: Impacts on highway entry and throughput," *Transportation Research Part C: Emerging Technologies*, vol. 13, no. 5, pp. 405–420, 2005. doi: https://doi.org/10.1016/j.trc.2004.09.001. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0968090X06000040 [Page 13.]

[48] R. Rajamani, *Vehicle Dynamics and Control*, 2nd ed., ser. Mechanical Engineering Series. New York, NY: Springer US, 2012. ISBN 1-283-44402-X [Page 13.]

[49] A. Ali, G. Garcia, and P. Martinet, “The flatbed platoon towing model for safe and dense platooning on highways,” *IEEE Intelligent Transportation Systems Magazine*, vol. 7, no. 1, pp. 58–68, 2015. doi: 10.1109/MITS.2014.2328670 [Page 14.]

[50] J. Ploeg, B. T. M. Scheepers, E. van Nunen, N. van de Wouw, and H. Nijmeijer, “Design and experimental evaluation of cooperative adaptive cruise control,” in *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, 2011. doi: 10.1109/ITSC.2011.6082981 pp. 260–265. [Page 14.]

[51] S. Santini, A. Salvi, A. S. Valente, A. Pescapé, M. Segata, and R. Lo Cigno, “A consensus-based approach for platooning with intervehicular communications and its validation in realistic scenarios,” *IEEE Transactions on Vehicular Technology*, vol. 66, no. 3, pp. 1985–1999, 2017. doi: 10.1109/TVT.2016.2585018 [Page 14.]

[52] M. Amoozadeh, A. Raghuramu, C.-n. Chuah, D. Ghosal, J. Rowe, and K. Levitt, “Security vulnerabilities of connected vehicle streams and their impact on cooperative driving,” *Communications Magazine, IEEE*, vol. 53, pp. 126–132, 06 2015. doi: 10.1109/MCOM.2015.7120028 [Page 15.]

[53] K. Kalogiannis, A. Henriksson, and P. Papadimitratos, “Vulnerability analysis of vehicular coordinated maneuvers,” in *2023 IEEE European Symposium on Security and Privacy Workshops (EuroSPW)*, 2023. doi: 10.1109/EuroSPW59978.2023.00006 pp. 11–20. [Page 15.]

[54] A. Abdo, Z. Qian, Q. Zhu, M. Barth, N. Abu-Ghazaleh, and S. Malek, “Application level attacks on connected vehicle protocols,” 09 2019. [Page 15.]

[55] M. Iorio, F. Risso, R. Sisto, A. Buttiglieri, and M. Reineri, “Detecting injection attacks on cooperative adaptive cruise control,” in *2019 IEEE Vehicular Networking Conference (VNC)*. IEEE, 2019, pp. 1–8. [Page 15.]

[56] K. Kalogiannis, “Investigating attacks on vehicular platooning and cooperative adaptive cruise control,” 2020. [Page 15.]

[57] R. Beyah, B. Chang, Y. Li, and S. Zhu, “Veremi: A dataset for comparable evaluation of misbehavior detection in vanets,” in *Security and Privacy in Communication Networks*. Switzerland: Springer International Publishing AG, 2018, vol. 254. ISBN 9783030017002 [Page 16.]

[58] J. Kamel, M. Wolf, R. W. Van Der Hei, A. Kaiser, P. Urien, and F. Kargl, "Veremi extension: A dataset for comparable evaluation of misbehavior detection in vanets," in *ICC 2020-2020 IEEE International Conference on Communications (ICC)*. IEEE, 2020, pp. 1–6. [Page 16.]

[59] C.-Z. Bai, V. Gupta, and F. Pasqualetti, "On kalman filtering with compromised sensors: Attack stealthiness and performance bounds," *IEEE transactions on automatic control*, vol. 62, no. 12, pp. 6641–6648, 2017. [Page 16.]

[60] E. Khanapuri, V. V. T. K. Chintalapati, R. Sharma, and R. Gerdes, "Learning based longitudinal vehicle platooning threat detection, identification and mitigation," *IEEE transactions on intelligent vehicles*, vol. 8, no. 1, pp. 1–1, 2023. [Page 16.]

[61] N. Bermad, S. Zemmoudj, and M. Omar, "Securing vehicular platooning against vehicle platooning disruption (vpd) attacks," in *2019 8th International Conference on Performance Evaluation and Modeling in Wired and Wireless Networks (PEMWN)*, 2019. doi: 10.23919/PEMWN47208.2019.8986956 pp. 1–6. [Page 16.]

[62] M. Sun, M. Li, and R. Gerdes, "A data trust framework for vanets enabling false data detection and secure vehicle tracking," in *2017 IEEE Conference on Communications and Network Security (CNS)*, 2017. doi: 10.1109/CNS.2017.8228654 pp. 1–9. [Page 16.]

[63] J. Kamel, M. R. Ansari, J. Petit, A. Kaiser, I. B. Jemaa, and P. Urien, "Simulation framework for misbehavior detection in vehicular networks," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 6, pp. 6631–6643, 2020. doi: 10.1109/TVT.2020.2984878 [Pages 17 and 18.]

[64] M. Wolf, A. Willecke, J.-C. Muller, K. Garlichs, T. Griebel, L. Wolf, M. Buchholz, K. Dietmayer, R. W. Van Der Heijden, and F. Kargl, "Securing cacc: Strategies for mitigating data injection attacks," vol. 2020-December, 2020, Conference paper. doi: 10.1109/VNC51378.2020.9318396 [Pages 17, 18, 25, 26, 55, and 69.]

[65] I. Sajjad, D. D. Dunn, R. Sharma, and R. Gerdes, "Attack mitigation in adversarial platooning using detection-based sliding mode control," in *Proceedings of the First ACM Workshop on Cyber-Physical Systems-Security and/or PrivaCy*, ser. CPS-SPC '15. New York, NY, USA: Association for Computing Machinery, 2015. doi: 10.1145/2808705.2808713. ISBN

9781450338271 p. 43–53. [Online]. Available: https://doi.org/10.1145/2808705.2808713 [Page 18.]

[66] E. Mousavinejad, F. Yang, Q.-L. Han, and L. Vlacic, "A novel cyber attack detection method in networked control systems," *IEEE transactions on cybernetics*, vol. 48, no. 11, pp. 3254–3264, 2018. [Page 18.]

[67] T. Moore, M. Raya, J. Clulow, P. Papadimitratos, R. Anderson, and J.-P. Hubaux, "Fast exclusion of errant devices from vehicular networks," in *2008 5th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, 2008. doi: 10.1109/SAHCN.2008.26 pp. 135–143. [Page 19.]

[68] M. Segata, S. Joerer, B. Bloessl, C. Sommer, F. Dressler, and R. Lo Cigno, "Plexe: A platooning extension for veins," vol. 2015, 12 2014. doi: 10.1109/VNC.2014.7013309 [Page 32.]

[69] 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. doi: 10.1109/TMC.2010.133 [Page 32.]

[70] P. A. Lopez, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y.-P. Flötteröd, R. Hilbrich, L. Lücken, J. Rummel, P. Wagner, and E. Wiessner, "Microscopic traffic simulation using sumo," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, 2018. doi: 10.1109/ITSC.2018.8569938 pp. 2575–2582. [Page 32.]

[71] 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*, ser. Simutools '08. Brussels, BEL: ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2008. ISBN 9789639799202 [Page 32.]

Appendix A

Mitigation Implementation

In Section 3.2 the suspiciousness based mitigation system by Wolf et al. [64] has been described. Due to deviations from the original and possible interpretation differences, the implementation chosen for this work will be shown here to provide further clarity. The following code shows how the mitigation system has been implemented in detail.

```
float aMin = -9.0; //Minimum Acceleration - Value taken from Wolf et al.
float aMax = 2.5; //Maximum Acceleration - Value taken from Wolf et al.
float tnoise = 0.1; //Noise treshold
float tmisb = 0.3; //misbehaviour threshold
float alpha = 0.8;

float a0 = (app->getLeaderAcc() - aMin)/(aMax - aMin); //Normalized Leader Acceleration
float a1 = (pb->getAcceleration() - aMin)/(aMax - aMin); //Normalized Predecessor Acceleration
float AttackP = fabs((a0-a1)/a0); //Attack Probability
float s = (1-alpha) * app->getSuspiciousness() + alpha*AttackP; //Suspiciousness
float h = (s-tnoise)/(tmisb-tnoise); //Headway Factor

double speed = 0;
double acceleration = 0;
VEHICLE_DATA data;
plexeTraciVehicle->getVehicleData(&data);
plexeTraciVehicle->getStoredVehicleData(&data, 0);
speed = data.speed;
acceleration = data.acceleration;

float Headway = 0; //initialize new headway

//If the Noise Threshold is exceeded -> set new Headway
if(s > tnoise)
{
    if(plexeTraciVehicle->getActiveController() == 5)
    {
        Headway = fmax(0.8, 2*h); //New headway for CONSENSUS
    }
    else if (plexeTraciVehicle->getActiveController() == 4 )
    {
        Headway = fmax(0.5, 2*h); //New headway for PLOEG
    }
    else // CVS CONTROLLERS
    {
        Headway = fmax(5.0 , (speed*(1000.0/3600.0)*2)*h);
    }

    // SET new Headway values
    plexeTraciVehicle->setCACCConstantSpacing(Headway);
    plexeTraciVehicle->setConsensusHeadwayTime(Headway);
    plexeTraciVehicle->setFlatbedHeadway(Headway);
    plexeTraciVehicle->setPloegCACCParameters(0.2, 0.7, Headway);

    //SAVE the new Suspiciousness for the predecessor
    app->setSuspiciousness(s);
}

// IF the misbehaviour threshold is exceeded
if(s >= tmisb && !app->isInManeuver())
{
    if(app->Prime()) // Check if PRIME MITIGATION is activated
    {
        LeaveManeuverParameters params;
        params.platoonId = positionHelper->getPlatoonId() ;
        params.leaderId = positionHelper->getLeaderId() ;
        params.position = positionHelper->getPosition() ;
        startManeuver(&params); //START the PRIME restructuring Protocol
    }
    else // Fallback to ACC
    {
        plexeTraciVehicle->setActiveController(0);
        plexeTraciVehicle->setACCHeadwayTime(2.0);
    }
}
```

Listing A.1: Simplified implementation of the suspiciousness based mitigation

TRITA-EECS-EX-2024:653

www.kth.se